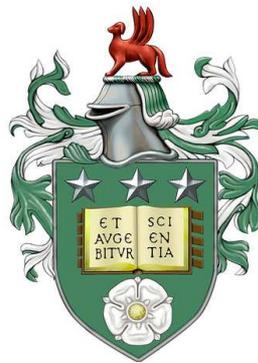


# **Representing and Reasoning about Changing Spatial Extensions of Geographic Features**

**by**

*Claudio Elizio Calazans Campelo*

**Submitted in accordance with the requirements for the degree of  
*Doctor of Philosophy***



**The University of Leeds  
School of Computing**

**July 2013**

The candidate confirms that the work submitted is his/her own, except where work which has formed part of jointly authored publications has been included. The contribution of the candidate and the other authors to this work has been explicitly indicated below. The candidate confirms that appropriate credit has been given within the thesis where reference has been made to the work of others.

Some parts of the work presented in this thesis have been published in the following articles. In each case, details are given on the contributions by the other authors and me, as well as details about the chapters which these articles were used within are mentioned.

**Campelo, C. E. C., Bennett, B.** “Representing and Reasoning about Changing Spatial Extensions of Geographic Features.”, Proceedings of the Conference on Spatial Information Theory (COSIT 2013), edited by T. Tenbrink, J. Stell and Z. Wood, LNCS 8116, pp. 3352, Springer International Publishing, 2013.

**My contributions:** Principal author, wrote all sections. Implemented the system prototype.

**Other author contributions:** Overall wording of the paper; discussion on the development of the formalism.

**Chapters based upon work:** Chapter 4 refers to the logical framework for representing events and processes. Chapter 5 presents the system prototype.

**Campelo, C. E. C., Bennett, B.** “Towards a Comprehensive Framework for Representing and Reasoning about Geographic Phenomena.”, Proceedings of International Workshop: Spatio-temporal Theories and Research for Environmental, Urban and Social Sciences: Where Do We Stand?, held in conjunction with the Conference on Spatial Information Theory (COSIT 2013).

**My contributions:** Principal author, wrote all sections.

**Other author contributions:** Overall wording of the paper.

**Chapters based upon work:** Chapter 4 refers to the logical framework for representing geographic phenomena. Chapter 5 presents the system prototype.

**Campelo, C. E. C., Bennett, B.** “Knowledge Representation of Geographic Phenomena.”, Mehdi Khosrow-Pour (Editor), Encyclopedia of Information Science and Technology, Third Edition, IGI Global Publishing. In Press.

**My contributions:** Principal author, wrote all sections.

**Other author contributions:** Overall checking of the paper.

**Chapters based upon work:** Chapter 2 contains part of the discussion given in this book chapter for background.

**Campelo, C. E. C., Bennett, B. and Dimitrova V.**, “From Polygons and Timestamps to Dynamic Geographic Features: Grounding a Spatio-temporal Geo-ontology.”, *Proceedings of the 6th International Workshop on Semantic and Conceptual Issues in GIS (SeCoGIS 2012), held in conjunction with the 31st International Conference on Conceptual Modelling (ER 2012)*. S. Castano et al.(Eds.): ER Workshops 2012, Lecture Notes in Computer Science, 7518, pp. 251–260, Springer, Florence, Italy, 2012.

**My contributions:** Principal author, wrote all sections.

**Other author contributions:** Overall wording of the paper.

**Chapters based upon work:** Chapter 3 describes in more detail the approach to representing spatio-temporal data and to modelling geographic features.

**Campelo, C. E. C., Bennett, B. and Dimitrova V.**, “Identifying Geographic Events and Processes in Spatio-temporal Data.” *Proceedings of the 6th Starting Artificial Intelligence Research Symposium (STAIRS), held in conjunction with the 20th European Conference on Artificial Intelligence (ECAI)*. Montpellier, France, 2012.

**My contributions:** Principal author, wrote all sections. Implemented the system prototype.

**Other author contributions:** Overall wording of the paper; Discussion on the development of the representational approach.

**Chapters based upon work:** Chapter 4 refers to the approach to representing spatial changes, events and processes. Chapter 5 presents the system prototype.

**Campelo, C. E. C. and Bennett, B.**, “Geographical Processes Representation: Issues and Challenges.”, *Towards Universal Ontology of Geographic Space*, IGI Global Publishing, Tomaz Podobnikar (Editor), 2012.

**My contributions:** Principal author, wrote all sections.

**Other author contributions:** Overall checking of the paper.

**Chapters based upon work:** Chapter 2 contains part of the discussion given in this book chapter for background.

**Campelo, C. E. C. and Bennett, B.**, “Applying Standpoint Semantics to Determine Geographical Processes Instances”, *Proceedings of the Workshop on Identifying Objects, Processes, and Events in Spatio-Temporally Distributed Data (IOPE)*, held at The Conference on Spatial Information Theory (COSIT’11), Belfast, Maine, USA, 2011.

**My contributions:** Principal author, wrote all sections.

**Other author contributions:** Overall wording of the paper; discussion on the development of the approach to handling vagueness; Checking of definitions.

**Chapters based upon work:** The approach presented in this paper to handling temporal

vagueness is a preliminary version of the approach presented in chapter 4.

**Campelo, C. E. C., Bennett, B. and Dimitrova V.**, “Identifying Geographical Processes from Time-stamped Data.”, *GeoSpatial Semantics: Proceedings of the 4th International Conference (GeoS 2011)*, edited by C. Claramunt, S. Levashkin and M. Bertolotto, Lecture Notes in Computer Science, 6631 , pp 70-87, Springer, Brest, France, 2011.

**My contributions:** Principal author, wrote most sections.

**Other author contributions:** Overall wording of the paper; discussion on the development of the approach to representing spatio-temporal data; Checking of definitions; Section 5.2 (with some input from myself).

**Chapters based upon work:** This paper presents a preliminary version of the approach presented in chapter 3 to representing spatio-temporal data.

This copy has been supplied on the understanding that it is copyright material and that no quotation from the thesis may be published without proper acknowledgement.

©2013 The University of Leeds and Claudio Elizio Calazans Campelo.

The right of Claudio Elizio Calazans Campelo to be identified as Author of this work has been asserted by him in accordance with the Copyright, Designs and Patents Act 1988.

# Abstract

This thesis presents a novel approach to representing and reasoning about geographic phenomena which can be interpreted based on changes affecting spatial extensions of geographic features. Of particular interest in this work are geographic features whose extensions can be described as 2-dimensional regions corresponding to portions of the earth surface under a specified projection, such as deserts, forests and oceans.

The work resulted in the development of a logical framework for representing geographic events and processes. In developing such a framework, issues have been addressed regarding the relationship between these concepts and also between them and geographic features. Other crucial issues are how to define the relation between event and process types and their particular instances, and how to handle different kinds of vagueness to associate specific spatial and temporal boundaries with those instances.

Of particular interest in this work is the development of a method of explicitly linking the formalism to spatio-temporal data. This requires work at multiple levels, both in consideration of how the data can be represented and in regards of how primitive elements of the logical framework can be defined.

Although data can be regarded as a faithful reproduction of physical elements of the world, some conceptual elements are not always explicitly represented within data. For that reason, a logic-based approach to representing spatio-temporal geographic data was also developed and is presented in this thesis. Representing the data in a logical fashion allows implicit data to be derived by means of logical inferences, and provides a natural way of explicitly connecting the data to a semantic-based formalism. Derived data may include spatial extensions of geographic features at different times, based on existing data describing, for example, portions of the earth's surface associated with different observable properties.

Furthermore, a system has been implemented to evaluate the applicability of the proposed theory. The system takes time-stamped topographic data as an input and allows logical queries to be formulated about the data, returning textual and graphical information on geographic events, processes, and features which participate in them.

# Acknowledgements

I would like to thank my supervisors, Dr. Brandon Bennett and Dr. Vania Dimitrova, for all their support and guidance throughout the course of this PhD.

I would like to thank CAPES for providing financial funding for this PhD.

I would like to thank my viva examiners, Dr. Antony Galton and Dr. Anthony Cohn, for all their valuable comments on my work.

I would like to thank my transfer examiner, Dr. John Stell, for his important feedback on my work.

I would like to thank the members of the KRR group for all their comments on my work.

I would like to thank other students and staffs of the School of Computing who gave any kind of feedback on my work.

I would like to thank the anonymous reviewers of the papers I have submitted for publication, for their comments and suggestions.

I would like to thank the School's administrative staffs for all the assistance provided. Special thanks go to Judi, George, Charlotte, Teresa, Graham and Ian who helped me sort things out in many different situations.

I would like to thank my MSc supervisor, Dr. Claudio Baptista, who persuaded me to study abroad and provided important guidance during this PhD.

I would like to thank my parents for being so supportive throughout my academic career, encouraging me to continue with my studies and to achieve all that I could.

I would like to thank my family and friends for their continuous support throughout my time in Leeds.

I would like to thank my son, Pedro Campelo, for his love and for being the greatest motivation for completing this PhD.

Finally, very special thanks are due to my wife, Carolina Campelo, for all that she has done for me that made possible this PhD dream come true.

# Acronyms

AI	Artificial Intelligence
COBRA	Context Broker Architecture
DBMS	Database Management Systems
DETER	Near Real Time Deforestation Detection in the Amazon Region
DR	Derivation Rule
EPL	Event Pattern Language
GIS	Geographic Information Systems
GIScience	Geographic Information Science
ISO	International Organization for Standardization
KB	Knowledge Base
ODBC	Open Database Connectivity
QSR	Qualitative Spatial Reasoning
QSTR	Qualitative Spatio-Temporal Reasoning
QTR	Qualitative Temporal Reasoning
RCC	Region Connection Calculus
RDBMS	Relational Database Management Systems
RGB	Region-Based Geometry
SQL	Structured Query Language
STAR	Spatio-temporal Attributed Region
XML	Extensible Markup Language
XSD	XML Schema Definition Language

# Contents

<b>1</b>	<b>Introduction</b>	<b>1</b>
1.1	Introduction . . . . .	1
1.2	Research Questions . . . . .	3
1.3	Achievements . . . . .	4
1.4	Thesis Overview . . . . .	6
<b>2</b>	<b>Background</b>	<b>8</b>
2.1	Introduction . . . . .	8
2.2	Qualitative Spatial Reasoning . . . . .	8
2.3	Qualitative Temporal and Spatio-Temporal Reasoning . . . . .	10
2.4	The Frame Problem . . . . .	12
2.5	Material Objects and Fields . . . . .	12
2.6	Semantic Models for Events and Processes . . . . .	15
2.7	Representing Events and Processes in the Geographic Domain . . . . .	18
2.8	Approaches to Vague Reasoning . . . . .	21
2.8.1	Fuzzy Logic in Geography . . . . .	22
2.8.2	Supervaluation Theory in Geography . . . . .	22
2.8.3	Handling Vagueness in Geography . . . . .	23
2.9	Dealing with Information Granularity . . . . .	25
2.10	Spatial Aggregates . . . . .	26
2.11	Ontology Grounding . . . . .	27
2.12	Summary . . . . .	28
<b>3</b>	<b>Representing Data and Geo-Features</b>	<b>29</b>
3.1	Introduction . . . . .	29
3.2	Overview of the Problem and Motivation . . . . .	30
3.3	Introducing the Logical Framework . . . . .	32
3.3.1	Space . . . . .	32

3.3.2	Spatial Region Coverages . . . . .	33
3.3.3	Time . . . . .	34
3.3.4	Geographic Features . . . . .	34
3.4	Spatio-Temporal Attributed Regions . . . . .	35
3.4.1	Types of Attributes . . . . .	36
3.4.2	Formal Description . . . . .	37
3.5	Axiomatisation . . . . .	38
3.5.1	Integrity Constraints . . . . .	39
3.5.2	Derivation Rules . . . . .	46
	Derivation Rule (DR1) . . . . .	46
	Derivation Rule (DR2) . . . . .	48
	Derivation Rule (DR3) . . . . .	52
	Derivation Rule (DR4) . . . . .	56
	Derivation Rule (DR5) . . . . .	58
3.5.3	Aggregated STARS . . . . .	61
	Derivation Rule (DR6) . . . . .	62
3.6	Modelling Geographic Features . . . . .	64
3.7	Summary . . . . .	70
<b>4</b>	<b>Logical Framework</b>	<b>72</b>
4.1	Introduction . . . . .	72
4.2	Overview of the Problem . . . . .	73
4.3	Events and Processes . . . . .	74
4.4	Syntax and Semantics . . . . .	75
4.4.1	Syntax . . . . .	76
4.4.1.1	Functional Terms . . . . .	76
4.4.1.2	Auxiliary Functions . . . . .	77
4.4.1.3	Atomic Propositions . . . . .	78
4.4.1.4	Propositional Constructs . . . . .	78
4.4.2	Semantics . . . . .	79
4.5	Representing Events . . . . .	80
4.6	Representing Processes . . . . .	82
4.7	Relating Events and Processes . . . . .	83
4.8	Process Activeness and Event Occurrences . . . . .	83
4.9	Determining Process Boundaries . . . . .	86
4.9.1	Deriving Events from Processes . . . . .	89

4.10	Identifying Processes in Spatio-temporal Data . . . . .	90
4.10.1	Spatial Changes . . . . .	90
4.10.2	Defining Process Activeness . . . . .	91
4.11	Summary . . . . .	91
<b>5</b>	<b>System Implementation</b>	<b>93</b>
5.1	Introduction . . . . .	93
5.2	The System Prototype . . . . .	93
5.2.1	The System Architecture . . . . .	94
5.2.2	Logical Language Choice . . . . .	96
5.2.3	Query Handling . . . . .	96
5.2.3.1	Argument Passing and Control . . . . .	98
5.2.3.2	Searching for Solutions . . . . .	100
5.2.3.3	Progress' Command-line Terminal . . . . .	100
5.2.4	Result Visualisation . . . . .	102
5.3	Building a KB upon a Relational Database . . . . .	103
5.4	Implementing the Spatial Reasoner . . . . .	104
5.5	Implementing the Temporal Reasoner . . . . .	105
5.6	Implementing the STAR Model . . . . .	105
5.6.1	Implementing Axioms for Integrity Checking . . . . .	106
5.6.2	Implementing Derivation Rules . . . . .	107
5.6.3	Explicitly Asserting Derived Facts . . . . .	107
5.6.4	Implementing the Principle of Inertia . . . . .	109
5.6.5	Calculating Maximal Extensions of Regions . . . . .	110
5.6.6	Calculating Spatial Aggregates and Concave Hulls . . . . .	110
5.7	Identifying Geographic Features . . . . .	112
5.8	Implementing Predicates for Events and Processes . . . . .	113
5.9	Summary . . . . .	115
<b>6</b>	<b>Results of Using the System</b>	<b>116</b>
6.1	Introduction . . . . .	116
6.2	Evaluation Methodology . . . . .	117
6.2.1	Auxiliary Tools . . . . .	118
6.3	Case Study and Dataset Description . . . . .	121
6.4	Data Collection and Preprocessing . . . . .	122
6.5	Evaluating the STAR Model . . . . .	124
6.5.1	Deriving <i>Stars</i> with Different Attributes . . . . .	124

6.5.2	Identifying Spatial Aggregates . . . . .	130
6.5.2.1	Distance Measurements . . . . .	130
6.5.2.2	Calculating Spatial Aggregates . . . . .	131
6.5.3	Execution Order of Derivation Rules . . . . .	132
6.5.4	Geographic Feature Generation . . . . .	133
6.6	Evaluating the Logical Framework . . . . .	135
6.6.1	Logical Queries Used . . . . .	135
6.6.2	Queries about Processes . . . . .	136
6.6.2.1	Queries Containing No Free Variables . . . . .	136
6.6.2.2	Queries Containing Free Variables . . . . .	144
6.6.3	Queries about Events . . . . .	148
6.6.3.1	Queries Containing No Free Variables . . . . .	148
6.6.3.2	Queries Containing Free Variables . . . . .	149
6.7	Overall Analysis and Conclusion . . . . .	157
6.8	Summary . . . . .	158
<b>7</b>	<b>Conclusions and Future Work</b>	<b>159</b>
7.1	Thesis Overview and Contributions . . . . .	159
7.2	Discussion and Future Work . . . . .	161
7.3	Conclusion . . . . .	164
	<b>Bibliography</b>	<b>166</b>

# List of Figures

2.1	RCC-8 Relations . . . . .	10
2.2	Allen’s Interval Algebra: 13 Base Relations . . . . .	11
2.3	Possible forest demarcations for a given tree distribution . . . . .	22
3.1	Possible part-hood relations between <i>Stars</i> . . . . .	43
3.2	Examples of <i>Stars</i> derived by DR1 . . . . .	47
3.3	Logical inferences which follow DR2 . . . . .	51
3.4	Applying DR2 followed by DR1 . . . . .	52
3.5	Drawing inferences according to DR3 . . . . .	54
3.6	Logical inferences that follow DR3 . . . . .	55
3.7	Inferences drawn in the example of Figure 3.6 . . . . .	56
3.8	Deriving geographic features from spatial regions and their coverages. . .	57
3.9	Deriving spatial regions and their coverages are from geographic features	59
3.10	Combining derivation rules to integrate data . . . . .	60
3.11	Aggregating disconnected spatial regions . . . . .	61
3.12	Different <i>Stars</i> representing regions with distinct coverages . . . . .	68
3.13	Spatial extensions of a particular geographic feature . . . . .	68
3.14	Feature as a spatio-temporal volume. . . . .	69
3.15	Layers of abstraction for connecting the logical and data levels. . . . .	70
4.1	Possible primitive event tokens for a given event type . . . . .	86
4.2	Example of a processes that is said to proceed . . . . .	88
4.3	Examples of processes that are said to proceed on distinct intervals . . . .	89
4.4	Examples of spatial changes . . . . .	90
5.1	System prototype’s main screen. . . . .	94
5.2	Architecture of the system prototype. . . . .	95
5.3	Visualising Results in a Graphical Manner. . . . .	103
5.4	Unsuitable concave hulls generated for changing aggregates. . . . .	112

6.1	uDig's main screen. . . . .	119
6.2	Visualisation tool main screen. . . . .	121
6.3	<i>Star</i> facts explicitly asserted in the synthetic KB. . . . .	125
6.4	<i>Star</i> facts in the initial KB and other produced by DR3. . . . .	126
6.5	<i>Star</i> facts representing spatial extensions of geographic features. . . . .	127
6.6	System output for Query 1 . . . . .	128
6.7	System output for Query 2 . . . . .	129
6.8	Checking whether a process is active on any interval. . . . .	139
6.9	Checking whether a process is active at a particular instant. . . . .	141
6.10	Obtaining features and intervals for conducting experiments. . . . .	143
6.11	Identifying features which participate in processes active at a given instant. . . . .	145
6.12	Identifying features and intervals on which processes are active. . . . .	147
6.13	Spatial extensions of features returned for Query 8 at 2 different instants . . . . .	148
6.14	System's answer for Query 10 (Geographic features which participate in events). . . . .	151
6.15	First 18 results returned by Progress to Query 11b. . . . .	153
6.16	Extension of feature 131 at 6 different time instants . . . . .	154
6.17	First 18 results returned by Progress to Query 11c . . . . .	156

# List of Tables

- 6.1 Results of generating spatial aggregates with 3 distinct aggregation factors 131
- 6.2 Results of executing derivation rules in different order . . . . . 133
- 6.3 Results of generating geographic features (statistics on MLPs) . . . . . 134
- 6.4 Results of generating geographic features (execution time) . . . . . 135

# Chapter 1

## Introduction

---

### 1.1 Introduction

There has been an increasing awareness of the importance of semantic models as an effective resource for representing knowledge in the geographical domain, providing a precise conceptualisation of the entities present in geographic space and the relationships between them [28, 55, 89]. Moreover, the Geographical Information Science (GIScience) community has increasingly realised the role that semantic-based approaches play in developing modern Geographic Information Systems (GIS). Coupled with this, substantial efforts have been made by many scientists to include time and the representation of *geographic phenomena* as fundamental constituents of GIScience [46, 49, 67, 94].

In GIS, significant contributions have been made towards the modelling of different kinds of geographic phenomena, and an assorted terminology has been applied (e.g. geoprocesses, geo-phenomena, dynamic GIS, spatio-temporal GIS). In the field of Knowledge Representation (KR), spatio-temporal reasoning [18, 32, 90] and reasoning about spatio-temporal changes [51, 84] have been investigated.

Representing geographic phenomena in terms of *events* and *processes* has been suggested by many authors [20, 27, 38, 91, 92], and such concepts appear to be significant in the way humans reason about changes affecting geographic space. Establishing a suitable representation for geographic events and processes requires dealing with issues regarding the relationship between these concepts, between their types and particular instances, and

also between them and geographic features. Other important issues are how to define the relation between events and process types and their particular instances, and how to associate specific spatial and temporal boundaries with those instances.

Geographic information can be affected by different kinds of vagueness, leading to considerable representational difficulties [9]. Such a representation task becomes particularly challenging when the temporal dimension is considered. Thus, associating specific spatial and temporal boundaries with instances of events and processes requires an appropriate method of handling spatio-temporal vagueness. Although many approaches have been proposed for dealing with vagueness in geography, it seems that methods of handling spatio-temporal vagueness for representing and reasoning about geographic events and processes have not yet been sufficiently investigated.

Formal theories for modelling spatial changes, events and processes have been proposed. Nonetheless, most approaches are not particularly related to the geographic domain and their applicability to geographic space would require further developments. In addition, although some works provide important directions, most of them are not yet implemented, and therefore their suitability for processing real-world data is not often discussed. Implementing a system to evaluate a logical framework with real data requires establishing a method of grounding the symbols upon elements of data. This requires work at multiple levels, both to select the appropriate set of predicates to be grounded and to formulate a suitable representation for the data. Methods of grounding geographic ontologies upon the data have been already proposed [13, 74]; however, approaches to developing a semantic-based formalism grounded upon spatio-temporal data have not been sufficiently discussed in the literature, and therefore further investigations are still required.

Geographers have been making notable contributions to the field of geostatistics, and have been producing an increasing amount of useful data related to a variety of fields (e.g., hydrology, geology, climate, urban planning, logistics, and epidemiology). Furthermore, the advance in remote sensing techniques, geo-sensor networks and other areas of computer science has contributed to the production of effective mechanisms for generating spatio-temporal geographic data. Nevertheless, such data can be fully exploited only when we provide intelligent ways to enrich data semantically.

Although data can be regarded as a faithful reproduction of physical elements of the world, descriptions of certain conceptual elements are not often explicitly represented within data. For example, a dataset containing values of average precipitation at different times could be useful for analysing changes affecting the spatial extension of deserts, even though such extensions are not explicitly given in the data. Hence, a system that intends

to link a spatio-temporal formalism to real world spatio-temporal data, should therefore include methods of deriving coherent descriptions of conceptual entities of geographic space.

This thesis examines how geographic phenomena can be interpreted within a logical framework for representing and reasoning about events and processes, with focus on geographic phenomena which can be described in terms of *changes affecting the spatial extension of geographic features*. Examples of geographic phenomena are ‘urbanisation’, ‘desertification’ and ‘deforestation’, which can be described in terms of spatial changes affecting ‘urbanised areas’, ‘deserts’ and ‘forests’ respectively. Of particular interest in developing this framework is the phenomenon of vagueness and how it affects the modelling of geographic events and processes.

Moreover, this work investigates how a logic-based approach can be developed to representing spatio-temporal geographic data. Representing the data in a logical fashion allows implicit data to be derived by means of logical inferences, and should provide a natural way of explicitly connecting the data to a semantic-based formalism. Derived data may include spatial extensions of geographic features at different times, based on other data describing, for example, values of observable properties associated with particular portions of the earth’s surface.

This work also pays special attention to the representation of *changing geographic features*, as an appropriate modelling of these entities can provide ways of defining other conceptual elements of dynamic geographic space (e.g., events and processes) in a high level of abstraction. That is, the approach to representing geographic features aims to enable the grounding of the logical framework upon the spatio-temporal data, by establishing an abstraction layer between them. In that approach, the concept of geographic feature is defined in terms of primitive data elements (e.g., geometries, timestamps), so that other conceptual elements (e.g., events, processes) can be defined only in terms of geographic features, and therefore without any concerns about data structure. Finally, the applicability of the proposed theory is evaluated by using a system prototype which implements the formalism and can operate on different spatio-temporal datasets.

## 1.2 Research Questions

The main goal of this research is to develop a knowledge representation approach to identifying geographic events and processes in time-stamped topographic data. In this work, focus is placed on events and processes that can be described in terms of spatial transformations affecting spatial extensions of geographic features.

To achieve this goal, a formalism for representing events and processes has been developed, as well as methods of representing spatio-temporal data and to explicitly link the formal framework to the data. Moreover, a system has been implemented to evaluate the applicability of the proposed theory. This system takes temporal series of topographic data as an input and is able to answer different logical queries about the data.

The following principal research questions are addressed in this thesis:

- What is the most appropriate approach to representing events and processes in the geographic domain? How are events and processes related to each other, and how are they related to geographic features? How can events and processes be defined in terms of changes affecting spatial extensions of geographic features? What kinds of geographic features can be considered? What changes can affect their spatial extensions? Is there a single solution that fits all types of features and changes?
- How flexible must a logical framework be in order to represent events and processes relating to different geographic phenomena? What is the most suitable logical language to use within such a logical framework? Can approaches to handling spatial and temporal vagueness help represent a larger number of situations? What approaches to vague reasoning could be incorporated to the logical framework? What requirements should be considered in order to choose the most suitable approach?
- How can temporal topographic data be represented effectively? Can spatial extensions of geographic features be defined in terms of observable properties of the earth's surface? Can features' extensions be derived from temporal topographic data where such extensions are not explicitly given? What methods of representation can be used, and what information will this allow us to extract? How can the spatial relations between parts of the data be modelled, and can this be done efficiently?
- How can a logical framework be explicitly linked to the data? What work is required at both levels to ensure the levels can be properly connected? How can this framework be implemented to work with the data? How can vague reasoning approaches be incorporated into the representation of the data? What information can be extracted from the data using this approach?

### **1.3 Achievements**

The principal achievements of this work can be summarised as follows:

- *Development of a novel logical framework for representing and reasoning about geographic events, processes and their participant geographic features:*

This framework, named REGEP (REasoning about Geographic Events and Processes), comprises formal descriptions of space, time, geographic features, events and processes and some relationships which hold between them. It provides precise definitions for predicates to represent event occurrences, as well as to represent processes which are said to proceed on certain periods and to be active/inactive at different times. Moreover, the framework includes a method of handling vagueness which allows different instances of event or process types to be determined based on individual viewpoints.

- *Development of an approach to representing temporal topographic data in a logical fashion:*

This modelling approach has been named STAR (Spatio-Temporal Attributed Regions). In this approach, portions of the earth's surface are represented as spatial regions associated with different attributes (e.g., arid, cold). Moreover, spatial relationships that can hold between regions associated with distinct attributes are explicitly determined (e.g. 'a region associated with attribute A1 can be part of a region associated with attribute A2'). Then, based on these relationships, a number of different inferences can be performed to derive additional implicit data. Implicit data include spatial extensions of geographic features at different times, derived from other data describing distinct attributed regions. This approach also includes a method of handling spatial vagueness which allows the spatial extension of a geographic feature to be represented at different levels of granularities.

- *Development of a method of explicitly linking the logical framework to spatio-temporal data:*

In this method, a geographic feature is represented as an entity which endures over time and that is said to have a life, during which its spatial extension can be affected by different types of transformations. Thus this grounding method consists of representing geographic features and their lives based on primitive elements of data, allowing higher level concepts (e.g., events, processes) to be defined only in terms of changing features, and therefore without concerns about data structure.

- *Implementation of a system to evaluate the practical application of the proposed formalism using real spatio-temporal data:*

Beyond the contributions of the work to theory, it also resulted in the development of a system named Progress. The system takes time-stamped topographic data as an

input and allows logical queries to be formulated about the data, returning textual and graphical information on events, processes, and the geographic features which participate in them.

## 1.4 Thesis Overview

This thesis is organised as follows:

*Background:* Chapter 2 discusses the background to the thesis, giving an overview of the research that was required to form this thesis. This include a general consideration of semantic models for modelling geographic phenomena; a review of logic-based approaches to representing events and processes; the problem of grounding a geographic ontology upon the data; and a discussion of approaches to handling vagueness. The chapter also overviews some other topics related to this work, such as qualitative spatial and temporal reasoning; approaches to dealing with information granularity; methods of determining spatial aggregates and defining suitable contours enclosing all elements of a particular aggregate.

*Representing Spatio-temporal Data and Modelling Changing Geographic Features:* Chapter 3 presents a logic-based approach to representing geographic spatio-temporal data that allows implicit data to be derived. The chapter also presents an approach to representing geographic features whose spatial extensions are subject to changes over time. Moreover, the chapter discusses how this approach to representing features can help establish a explicit link between data and a logical framework for representing geographic events and processes.

*Logical Framework:* Chapter 4 presents the *logical framework* for representing and reasoning about geographic events and processes. This framework, named REGEP (REasoning about Geographic Events and Processes), comprises formal descriptions of space, time, geographic features, events and processes and some relationships which hold between them. In addition, the chapter describes how this framework can be used to identify events and processes within the data, by employing the method of linking these levels (presented in Chapter 3).

*System Implementation:* Chapter 5 presents Progress, a system prototype implemented to evaluate the applicability of the theory proposed in this thesis. A description of the system's architecture and an overview on its main components are given. Then specific discussions are conducted on the approaches to implementing the spatial and temporal reasoners; the logic-based approach to representing spatio-temporal data and geographic features (presented in Chapter 3); and the logical framework presented in Chapter 4.

*Results of Using the System with Spatio-temporal Data:* Chapter 6 describes the experiments carried out using the system prototype (presented in Chapter 5) with temporal series of topographic data, and examines the results obtained from these experiments.

*Conclusions and Future Work:* The final chapter summarises the most important aspects of this research, highlighting the main contributions presented. The chapter also discusses the strengths and limitations of the work, and points to future work that might be conducted in the field of research.

# Chapter 2

## Background

---

### 2.1 Introduction

This chapter presents an overview of the background to the various aspects of this work, which later chapters will expand upon. First, an overview is given on the fields of qualitative spatial and temporal reasoning. Then Section 2.5 discusses the representation of objects, fields, and their relation to the representation of geographic features. A discussion of semantic models for events and processes is given in Section 2.6. The representation of events and process in the geographic domain is discussed in Section 2.7. Then Section 2.8 overviews approaches to handling vagueness. Section 2.9 and 2.10 introduce the notion of granularity and aggregates, respectively. Finally, Section 2.11 overviews the problem of ontology grounding.

### 2.2 Qualitative Spatial Reasoning

Of particular interest in this thesis are qualitative approaches to reasoning about space, and the mereological and topological relationships involving spatial elements of geographic space. Qualitative Spatial Reasoning (QSR) allows reasoning about relations that hold between spatial objects. This is done without the need for precise quantitative information.

Whitehead et al. [87] presented a theory of extensive abstraction based on the two-place predicate, ‘x is extensionally connected with y’. Later on, Clarke [22] presented a calculus of individuals based on the Whitehead’s primitive predicate. Following Whitehead, Clarke interprets the individual variables as ranging over spatio-temporal regions and the two-place primitive predicate, ‘x is connected with y’, as a rendering of ‘x and y share a common point’. The author used most of Whitehead’s mereological definitions and proposed an axiomatisation for Whitehead’s theory. The work of Clarke [22] has led to the development of the well-cited Region Connection Calculus (RCC) [25, 69]. Other theories have also been developed based on the Clarke’s theory, such as the theory of common-sense geometry [86].

Perhaps the most popular approaches to qualitative spatial reasoning nowadays are the RCC [25, 69] and Egenhofer and Franzosa’s 9-Intersection Calculus [29, 30]. The former has its roots on logic (i.e. QSR), whilst the latter is based directly on point-set topology. RCC proposes a set of mereotopological relations which may hold between a pair of regions  $(r_1, r_2)$ . These relations are derived from a primitive relation  $C(x, y)$ , which means that the regions  $r_1$  and  $r_2$  are connected. This initial relation holds when the topological closures of regions  $r_1$  and  $r_2$  share a common point [69]. RCC-8 consists of the eight relations listed below (and illustrated in Figure 2.1). Depending upon the requirements, this set can be restricted or expanded.

- $DC(r_1, r_2)$  (Disconnected);
- $EC(r_1, r_2)$  (Externally Connected);
- $PO(r_1, r_2)$  (Partially Overlapping);
- $EQ(r_1, r_2)$  (Equals);
- $TPP(r_1, r_2)$  (Tangential Proper Part);
- $TPPi(r_1, r_2)$  (Tangential Proper Part Inverse);
- $NTPP(r_1, r_2)$  (Non-Tangential Proper Part); and
- $NTPPi(r_1, r_2)$ , (Non-Tangential Proper Part Inverse).

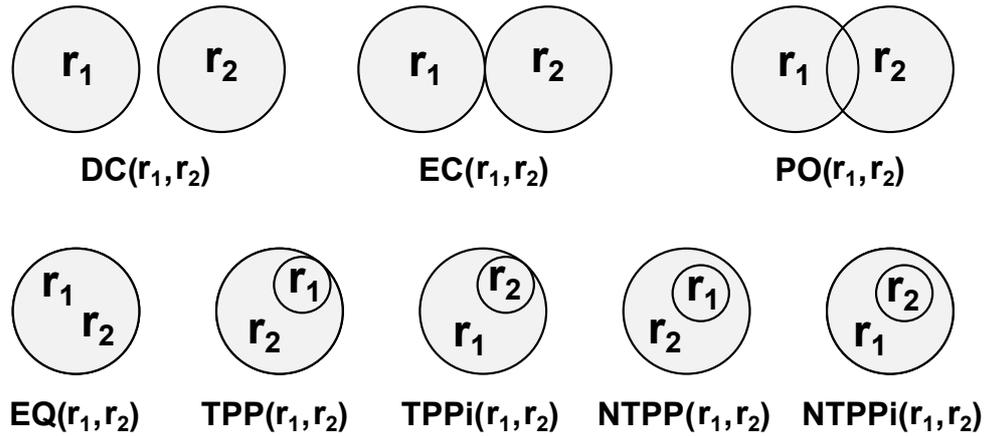


Figure 2.1: RCC-8 Relations.

## 2.3 Qualitative Temporal and Spatio-Temporal Reasoning

According to Peuquet [67], ‘The passage of time is normally understood via changes we perceive occurring to objects in space – their transformations over time and their movements in relation to one another’ (p. 01). In this context, considerable efforts have been made by many researchers to incorporate time as an essential component of GI-Science, aiming to develop a suitable representation of geographic phenomena. Galton [37] maintains that ‘our most basic notions of temporality are essentially qualitative: the idea that one event preceded another is conceptually more fundamental than the idea that the temporal separation of the events is a certain number of hours’ (p. 172). Qualitative approaches to representing time are of particular interest in this work.

Qualitative Temporal Reasoning (QTR) allows inferences to be made about the relations that hold between temporal elements (i.e., instants and intervals) without requiring quantitative measurements. The most popular approach is known as Allen’s Interval Algebra [1, 2]. This is a calculus for temporal reasoning which defines possible relations that hold between two time intervals and provides a composition table that can be used as a basis for reasoning about temporal descriptions of events. The Algebra consists of the following 13 relations (illustrated in Figure 2.2): Before( $i_1, i_2$ ), After( $i_1, i_2$ ), Meets( $i_1, i_2$ ), Met-By( $i_1, i_2$ ), Overlaps( $i_1, i_2$ ), Overlapped-By( $i_1, i_2$ ), Starts( $i_1, i_2$ ), Started-By( $i_1, i_2$ ), Finishes( $i_1, i_2$ ), Finished-By( $i_1, i_2$ ), During( $i_1, i_2$ ), Contains( $i_1, i_2$ ), Equals( $i_1, i_2$ ).

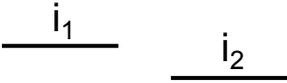
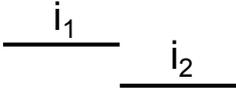
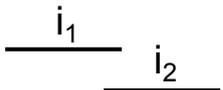
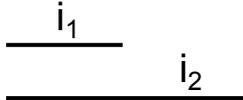
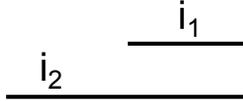
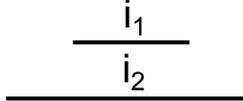
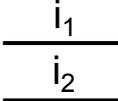
Before( $i_1, i_2$ )		After( $i_2, i_1$ )
Meets( $i_1, i_2$ )		Met-By( $i_2, i_1$ )
Overlaps( $i_1, i_2$ )		Overlapped-By( $i_2, i_1$ )
Starts( $i_1, i_2$ )		Started-By( $i_2, i_1$ )
Finishes( $i_1, i_2$ )		Finished-By( $i_2, i_1$ )
During( $i_1, i_2$ )		Contains( $i_2, i_1$ )
Equals( $i_1, i_2$ )		

Figure 2.2: Allen's Interval Algebra: 13 base relations that may hold between two intervals.

Qualitative Spatio-Temporal Reasoning (QSTR) encompasses both QSR and QTR techniques. Numerous models have been proposed in the literature in the field of QSTR. Perhaps one of the most influential is the mereotopological theory presented by Muller et al. [63], in which primitive entities are spatio-temporal regions, on which temporal spatio-temporal relations are defined. In addition, the concept of temporal slice introduced by Muller et al. [63] and Muller [62], where  $TS(x, y)$  means that 'x is a temporal slice of y' (i.e., the maximal component part corresponding to a certain time extent) is important for modelling spatio-temporal interactions and to define relations changing through time. Another important contribution of Muller's mereotopological theory of space-time was an explicit qualitative definition of continuity.

## 2.4 The Frame Problem

The frame problem was first introduced in 1969 by McCarthy and Hayes [59]. In the field of Artificial Intelligence (AI), the problem corresponds to the challenge of representing the effects of an action without having to represent explicitly its many accompanying obvious non-effects. In its technical form (i.e., mathematical logic), the frame problem deals with the question: how is it possible to specify formulae that describe the effects of actions without having to specify many other formulae that describe their mundane, intuitive non-effects of those actions? In philosophy, the frame problem is described as the problem of limiting the beliefs that have to be updated in response to actions. In the logical context, actions are traditionally specified by what they change, with the implicit assumption that everything else (the frame) remains unchanged [79].

Within classical AI, a number of workable solutions to the frame problem have been proposed, and this is no longer regarded as an obstacle (even for those working in a strictly logic-based paradigm [60, 78]).

The frame problem is solved in various formalisms, including:

- The event calculus solution [77];
- The default logic solution [4];
- The fluent calculus [85];
- The successor state axioms [71];
- The fluent occlusion solution and its similar solution known as predicate completion [73];
- Answer set programming [44];

Solutions to the logical frame problem developed by AI researchers normally appeal to the common sense law of inertia. According to this law, properties of a situation by default are assumed to remain unchanged as the result of a certain action. In the technical point of view, the frame problem can be regarded as the task of formalising this law [78, 79].

## 2.5 Material Objects and Fields

In this work, geographic features are regarded as a particular kind of enduring entity. Here geographic features share some of the characteristics described in the literature for

*material objects*, such as: they are discrete individuals; they have well-defined spatial extensions<sup>1</sup>; they are able to undergo change; and they can change some of their parts while keeping their identity. Therefore, this section discusses some key concepts related to the representation of objects and their relation to the concept of *fields*.

Galton [34] argues that ‘an *object* is defined by some abstract notion of identity which (a) defines (to an appropriate degree of precision) its spatial extent at any one time, and (b) enables it to persist through changes in spatial location and other attributes. Thus considered, an object has spatial parts but not temporal parts, rather existing as an entirety at each moment of its history’ (p. 02). Spatial objects are normally referred to as *continuants*, from the philosophical tradition. Galton [39] identifies four categories of objects: ‘(i) mobile objects (e.g., people, cars), (ii) fixed objects distinct from their environment (e.g., trees, buildings), (iii) fixed objects that form part of the environment (e.g., roads, rivers, mountains), and (iv) conventional (fiat) objects which only exist by virtue of human convention (e.g., administrative units, land registry parcels)’ (p. 08).

Jacquez et al. [52] state that ‘fields typically represent measurements on a variable whose value varies through geographic space. The altitude of the surface of the Earth is an example of a spatial field’ (p. 222). As described by Galton [34], ‘like objects, *fields* change over time, but unlike with objects there do not seem to arise any issues of identity’ (p. 2). Fields are generally represented by decomposing geographic space in minimal regions (which are not further divided into sub-regions) and then assigning to each region a value from some range. There is no restriction on the type of information which can be associated with a particular location. It can be a number (e.g., real, integer), a string or any other type of information. The nature of the information that these values represent vary according to the problem in hand. A few examples are temperature, level of precipitation, humidity, pressure, water salinity, elevation, vegetation type, tree density and a variety of population statistics (e.g., number of deaths from tuberculosis per month, number of births per 1000 girls aged 15 to 19).

Traditionally, objects and fields have been presented in the literature as two non-overlapping views of spatio-temporal data, and authors used to defend one of them as the most appropriate way to represent geographic space. Presently, the need for both views to do justice to the numerous ways of representing reality in geographic domain is generally accepted. Assuming that any spatial data are semantically weak and that one

---

<sup>1</sup>Actually, determining the spatial extension of a geographic feature is a matter of vagueness, as discussed in Section 2.8.3. However, it is assumed here that every feature in the model can be associated with a precise spatial extension. Methods of handling vagueness could be applied to assign precise spatial extensions to vague features. An example of such an approach to dealing with this issue based on standpoint semantics [10] is presented by Bennett et al. [12, 13].

of the roles of geospatial semantics is to provide solutions for semantic enrichment of spatial data, it can be realised that the development of appropriate methods of deriving useful information from existing data has become essential.

Peuquet et al. [66] support the idea that intermediate forms of representation between the pure views of objects and fields should be provided. An example is the ‘object-field’ representation [33]. Galton [34] identifies several interconnections between object-based view and field-based view. These interconnections are described below, together with additional cases in which fields can be created from fields:

- *Objects can be created from fields.* There are many cases in which we can single out certain features from a field data and designate them as objects. For example, if the field values express the average annual precipitation, then maximal connected regions within which the precipitation rate is less than 250 millimetres per year might be designated as a deserts.
- *Fields can be produced from objects.* For instance, from individual trees, which are naturally conceived as objects (to each of which a fixed location is assigned) we can produce different fields describing types of vegetation.
- *Objects can be generated from other objects.* It can be done either by aggregating/disaggregating them [81], or by detaching them, by considering the idea of parts (objects as component of other objects, such as wheels, tyres and windscreen as parts of a car).
- *Fields can be created from fields.* This consists of converting from a continuous to a discrete value field by banding. This conversion process is discussed by Galton [33], who names discrete value fields as ‘categorical *coverage* fields’. For example, values of field data comprising a single variable describing the average annual precipitation can be redefined in value bands (e.g., 0-50 mm, 50-250 mm, and 250-500 mm of annual rainfall) and then a new field could be generated, describing climate types (e.g., extremely arid, arid, semiarid), commonly applied to distinguish different types of deserts. Furthermore, one can still combine two or more field data covering the same spatial region to produce a new field. For instance, values of precipitation might be combined with other values, such as of temperature, humidity, evaporation, distance to the ocean, soil type, and elevation, to provide a new field containing a more accurate climate/geological classification.

Finally, the operations discussed above might be combined. For instance, from a field created from other fields (e.g., precipitation, temperature, humidity, elevation), one can

single out desert objects of more accurate types, such as montane deserts, rain shadow deserts, cold deserts, coastal deserts, monsoon deserts. It should be noticed that, in this case, objects would correspond to maximal connected overlaps of certain fields.

Reasoning about certain kinds of geographic phenomena requires tracking the changes affecting the spatial extension of geographic features. For example, the phenomenon of deforestation is associated with changes affecting the extension of a forest. However, frequently, the extensions of geographic features are not explicitly represented in the data. In these cases, feature extensions might be derived from field data. Consequently, different forms of mapping field values to objects lead to different interpretations of geographic phenomena being examined. The study of numerous geographic phenomena may be based on the examination of objects generated from fields. Examples are pollution, drought, and spread of a disease.

In agreement with Galton [34], the discussion given in this section leads to the conclusion that field-based and object-based are complementary perspectives and not poles of a sharp dichotomy. The the relationships presented here are just a small portion of a complex network of interrelations involving these views. Many questions remain open, both in relation to fundamental problems and in regard to aspects of implementation. Therefore a more appropriate understanding of these interconnections is critical to conceive comprehensive representational approaches to geographic phenomena.

## 2.6 Semantic Models for Events and Processes

A wide variety of terminology has been applied to describe events and processes. Particular disagreements can be observed about what the terms ‘event’ and ‘process’ refer to and what are the distinctions and relations between them. Of particular interest in this work are some remarks made by Galton [34, 36]. concerning some desirable elements for developing a comprehensive formalisation for events, processes and a spatio-temporal geo-ontology. Another important contribution in the field has been made by Grenon and Smith [47], who outlined a system comprising two kinds of ontologies, called SNAP (for continuants, e.g., individuals and objects) and SPAN (for occurrents, e.g., processes, events and changes), and a meta-ontological framework which deals with the relations between them.

A large number of formalisms have been proposed to deal with events. Among the most influential are the *Event Calculus* [54, 77] and the *Situation Calculus* [59]. The Event Calculus is a logical language for representing and reasoning about *actions* and their effects. The basic components of this language are *fluents*, which describe the state

of the world, and actions that can be performed in the world. In the Event Calculus, it is possible to specify the value of fluent at given time points, the actions that happened at given time points, and the effects of actions. The predicate  $\text{Holds-At}(f, t)$  specifies that the fluent holds at time point  $t$ .  $\text{Happens}(a, t)$  means that the action takes place at time  $t$ . The predicates  $\text{Initiates}$  and  $\text{Terminates}$  determine the effects of actions. For instance,  $\text{Initiates}(a, f, t)$  means that, if the action  $a$  happens at time  $t$ , then the fluent  $f$  will be true after  $t$ . In the Situation Calculus, other basic elements are added to the elements present in the Event Calculus: objects and situations. The former is any element upon which actions can be performed and the latter represents a history of action occurrences.

Bennett and Galton [6] proposed a formal semantics for representing temporal relationships and events. In this approach, a highly expressive language is proposed, which is called *Versatile Event Logic (VEL)*, encompassing the essential insights of many other approaches. To express the versatility of this language, it is illustrated how Situation Calculus and Event Calculus can be represented within VEL. Bennett and Galton [6] suggest a branching tree structure to determine relationships between different times, which comprehend all possible histories of the world. A world state  $h(t)$  determines all properties of the world at time  $t$  in history  $h$ . Event-types are distinguished from event-tokens, where the latter are occurrences of the former. It also provides semantics to model the identity of individuals which inhabit the time structure.

Bennett [8] presents an extremely expressive logical language for describing physical situations and processes, which brings together a variety of previously developed theories. This formalism is an ontologically well-founded background theory and therefore contains several desiderata for a comprehensive ontology of geographic phenomena. It incorporates the theory of Region-Based Geometry [7, 11], that allows the modelling of spatially-extended physical objects naturally. RGB can represent arbitrary geometrical properties and configurations of regions and so subsumes less expressive theories such as RCC [25, 69].

In the formalism proposed by Bennett [8], time is modelled as a set of histories which correspond to possible alternatives to the actual history of the world. This rich structure enables the employment of modal operators to distinguish between analytic and contingent propositions. In this structure, the truth of a proposition is evaluated relative to an index  $\langle h, t \rangle$  specifying a history and a time point. A history is modelled by a function from time points to physical states. This branching tree structure is also employed in the Versatile Event Logic (VEL) language proposed by Bennett and Galton [6] for representing temporal relationships and events. Bennett and Galton [6] discuss this model in more detail and point out some useful references for further study.

Bennett [8] models the world in terms of a vocabulary of basic matter types, each of which having a spatial extension that may change (continuously) over time. A state is modelled as a function from matter types to spatial regions. Indeed, at a given time, each matter type  $m$  denotes a region  $r$  which can be referred to using the relation  $\text{Ext}(m, r)$ . In this terminology, temporarily persistent objects are regarded as individuals (e.g., a cat, a person, a forest). Individuals do not exist at a single time point in a single history. They are modelled by employing a function which maps histories and time points to the spatial extension of the individual. This formalism also incorporates the Gupta's theory of count nouns [48]. In Gupta's theory, the count noun is evaluated at some possible world to give a set of individuals which are instances of the count noun. Each individual in turn is evaluated relative to a possible world to yield an entity existing at that world.

It is generally accepted amongst philosophers and logicians that, as with objects, events and processes are subject to the distinction between *types* and *tokens*, which separates abstract entities (types) and their particular instances (tokens). However, particular distinctions can be observed in the way authors specify types and relate them to their correspondent tokens. Some authors describe a type as an abstract entity which is not related to any specific participant, and particular instances of this type may involve distinct participants. For example, the event type 'knocking on a door' may have a particular instance 'John knocked on Mary's door today at 1 pm'. On the other hand, other authors argue that a type should incorporate the participants. For instance, John can knock on Mary's door at different times, and such occurrences can be regarded as distinct instances of an event of type 'John knocks on Mary's door'. This is the view considered in the work of this thesis.

Galton [36] surveys a number of approaches to describing events and processes and states that 'there has been an unfortunate terminological variation, in particular disagreements about how process is related to events' (p. 324). The author proposes a classification encompassing a variety of previously proposed terminology (e.g., activities, accomplishments, achievements, procedures, transition, and so on). In that classification, processes are classified regarding:

- their *homogeneity*, that is, some processes are most nearly homogeneous (e.g., falling, cooling, or sliding) and others are more 'granular' (e.g., walking or playing the piano);
- their *open-endedness*, i.e., while openended processes can in principle be continued indefinitely (e.g., walking or vibrating), closed process are generally associated with an definite result (e.g., making a cup of tea or refuelling a motor-car).

A set of operations to interrelate events and processes is proposed by Galton [36], such as ‘composition’, ‘specification’, ‘chunking’, ‘transitions’, ‘constituency’, and ‘repetition’. In agreement with Galton [36], the work presented in this thesis considers the following relationships between events:

- A process can be regarded as *constituted* of events (e.g., a process of type ‘walking’ can be regarded as a sequence of events of type ‘taking a step’);
- An event can be regarded as *made of* (or as a *chunking of*) processes (e.g., the event ‘Paul swam to the island’ can be regarded as a bounded instantiation of a process ‘Paul swimming’);

Given what has been discussed in this section, it could be argued that a semantic-based formalism to representing geographic phenomena should provide appropriate logical relations to represent the interrelations amongst events and processes. Such a complex network of interrelations might include those described by Galton [36], and also other not covered by the author, such as possible typological inheritance involving types of events and processes. For instance, processes of types ‘swimming’, ‘walking’, and ‘running’ may be regarded as a specialisation of a process of type ‘moving’. Furthermore, a flexible method of describing composite processes and events, e.g., Event Pattern Language (EPL) [42, 43], appears as another crucial component of such a formalism.

## 2.7 Representing Events and Processes in the Geographic Domain

There are many dynamic phenomena that may occur in geographic space, from ‘the falling of a rain drop’ to ‘the flow of cars along a road’. Some modelling approaches conceive events and processes based on series of spatial changes over time (e.g., changes of locations, change of land coverage) involving geographic entities (e.g., features, objects). These changes may characterise, for example, movement of an entity or shrinkage of a geographic feature. Other approaches take into account physical processes and chemical reactions involved in a geographic phenomenon. For instance, when investigating bush-fires, one might aim to model some physical processes (and sub-processes) that cause its ignition, such as lightning strikes or spontaneous combustion. The latter, for example, also involves other sub-processes, such as oxidation or fermentation of substance with a relatively low ignition temperature.

The representation of geographic phenomena in terms of events and processes has been investigated by researchers from several areas, such as GIScience, Knowledge Representation, and Spatio-Temporal Databases. Approaches to modelling such events and processes have been presented based on different methods (e.g., logic-based, object oriented, cellular automata, or agent-based), with different purposes (e.g., semantic integration, reasoning about process properties, simulation, or prediction), and sometimes developed to meet the requirements of a particular application area (e.g., urban spreading, wildfires, meteorology, human population studies, ecology, geomorphology).

Claramunt and Thériault [19, 20] have made significant contributions to the problem of representing of geographic events and processes. In [19], the authors define a typology of spatio-temporal processes comprising three main categories:

- *Evolution of a single entity*: basic changes (appearance, disappearance, stability); transformations (expansion, contraction, deformation); movements (displacement, rotation);
- *Functional relationships between entities*: replacement (succession, permutation); diffusion (production, reproduction, transmission);
- *Evolution of spatial structures involving several entities*: restructuring (split, union, re-allocation).

Additionally, Claramunt and Thériault [20] propose an event-oriented model to describe the evolution of spatial entities. This model is based on a relational formalism, extends the *versioning* concept (in which object ‘versions’ correspond to successive states of the represented entity) and distinguishes between thematic, temporal, and spatial domains. A semantic formalism is presented for describing and modelling spatio-temporal processes within the geographic domain. The approach is based on Event Pattern Language (EPL) [42, 43].

More recently, Devaraju and Kuhn [27] presented a process-centric ontology to represent relations between geographic processes and observed properties originating from Geo-Sensor Networks (GSNs), by defining a controlled terminology which allows an explicit representation of a process and its participants. ‘A Geo-Sensor Network (GSN) is a network of sensors that monitors the properties of natural environments in different locations at different times’ [27, p. 01]. Devaraju and Kuhn [27] argue that ‘facts about geo-processes can be inferred from sensor data. However, this is not easy to achieve due to the nature of sensors, which return huge amounts of data in different formats and semantics’ (p. 01). Thus the main goal of Devaraju’s and Kuhn’s research is to define

controlled terminologies that can be used to present a unified view over heterogeneous sensor data.

The work described by Devaraju and Kuhn [27] is focused on the development of an ontology to represent concepts of surface hydrology, based on DOLCE Ontology Library [41, 58]. The authors align the concepts and relationships of two processes in the hydrological domain (precipitation and evapotranspiration) to the general categories defined in DOLCE, such as *endurant*, *perdurant*, and *quality*. This initiative represents an important step to overcoming the challenge of filling the gap between low-level measurements and high-level conceptualisations in the geographic domain. However, developing an ontology comprising a fully characterisation of geographic processes, their participating endurants and related temporal and physical properties is still the subject of exhaustive research.

Galton and Worboys [40] maintain that it is generally accepted that object, event and process are key concepts for modelling dynamic geographic phenomena; besides, of course, an appropriate representation for space and time. This chapter now discusses important issues and points out several desiderata for the conceptualisation of space, time, object, event, and process for the development of an ontology for dynamic geographic phenomena.

In the modular ontology of dynamic features of reality presented by Grenon and Smith [47], continuants are described as the class of entities which ‘exist in full in any instant of time at which they exist at all and they preserve their identity over time through a variety of different sort of changes’ (p. 140), such as a person, the planet Earth, a rock and Leeds. This is the concept normally attributed to spatial objects which inhabits a spatio-temporal model. On the other hand, the authors state that, while ‘continuants are themselves subject to constant change, occurrents depend on continuant objects as their bearers’ (p. 140). According to the authors, occurrents are all bound in time and this classification comprises events and processes. Continuants are commonly referred to as endurants, and occurrents are usually referred to as perdurants.

Galton [36] agrees with Grenon and Smith (2004) in the view that continuants are time-dependent entities and occurrents are time-independent. This is to say that, whilst the properties of a continuant can be different at different times, the properties of an occurrent are possessed timelessly. Galton (2007) summarises the distinction between continuants and occurrents by saying that continuants are entities which ‘(a) can undergo change, (b) have spatial parts but not temporal parts, and (c) are wholly present at each moment of its existence’ (p. 329). In contrast, occurrents are entities which ‘(a) cannot undergo change, (b) has temporal parts, and (c) are not wholly present at any time short

of its entire duration' (p. 330).

Nonetheless, Galton [36] disagrees with Grenon and Smith [47] in the point that processes are regarded as occurrents by the latter. Galton [36] gives several compelling arguments and examples to support the view that a process should be regarded as an entity which undergoes change. Thus, the author suggests to set aside the distinction between continuants and occurrents, and proposes a new dichotomy encompassing two ontological views, called EXP (experience) and HIST (history), one populated by time-dependent entities (objects and processes) and the other by time-independent entities (events), respectively.

Following Galton [36], in the work of this thesis, a process is regarded as an entity which is subject to change over time and to which we can ascribe certain properties (for example, a process may be described as being slowing down, or accelerating). Therefore, a comprehensive ontology of geographic processes should provide the appropriate conceptualisation for these distinct ontological views (i.e., one comprising time-dependent entities and the other comprising time-independent entities), and the appropriate mechanism to do justice with the complex interrelations involving these views.

## 2.8 Approaches to Vague Reasoning

'Vagueness arises from lack of definite criteria for the applicability of certain linguistic terms' [9, p. 02]. The most evident type of vague expression is adjectives, such as 'large', 'small', 'tall', 'short'. Bennett [9] states that the vagueness of count nouns (e.g., table, car, lake, mountain) differs from that of adjectives as to the number of parameters of variation that are usually involved. For example, whilst the tallness depends on the height, the definition of table depends on many factors, such as its constituent material, flatness of its surface, existence of legs, and various other constraints.

The existence of vagueness in the geographic domain leads to an indeterminacy of spatial extension of some geographic feature types [9]. The problem of individuation is discussed by Bennett [9], who defines the problem as 'the determination of the class of entities to which the predicate might be applied'. Bennett [9] illustrates possible forest demarcations for a given tree distribution, based on different choices for a threshold on the tree density. This example is reproduced in Figure 2.3, where inner and outer contours are based on higher and lower thresholds, respectively.

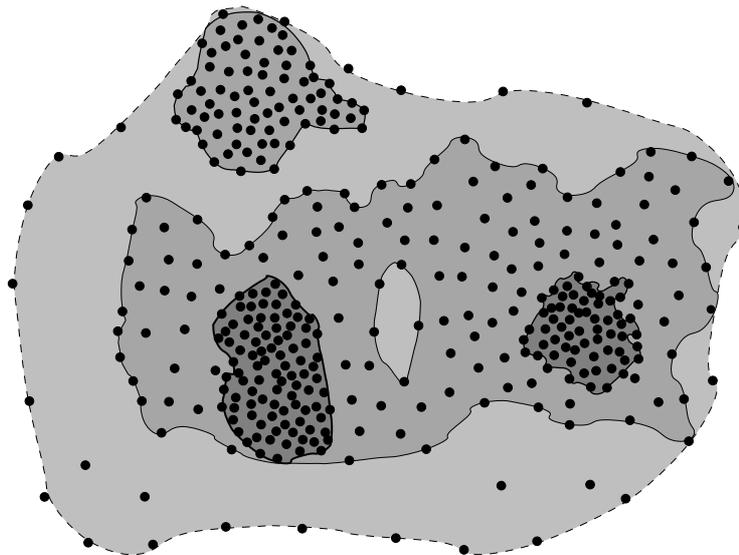


Figure 2.3: Possible forest demarcations for a given tree distribution [9].

### 2.8.1 Fuzzy Logic in Geography

Several approaches have been proposed to handle vagueness about time and space. An influential approach to modelling spatial vagueness is that of fuzzy logic and the theory of fuzzy sets, originated with the works by Zadeh [95, 96]. In approaches to spatial vagueness based on fuzzy logic, the degree of membership of a point in a vague region corresponds to the degree of membership in the fuzzy set. An important approach has been proposed by Schockaert et al. [76], which consists of a fuzzy version of the Region Connection Calculus. In this approach, the degree of connection is modelled by taking the primitive relation  $C$  as a fuzzy set and replacing all other relations using fuzzy logic operators.

### 2.8.2 Supervaluation Theory in Geography

Other important approaches have been developed based on the supervaluationist account of vagueness, in which each completely precise interpretation of a vague predicate is defined as a precisification. The standpoint semantics [10] is an extension of supervaluation semantics. The main idea of standpoint semantics is to define a finite number of parameters (standpoints) related to observable properties in order to describe each possible precisification of a vague predicate, and then assign different threshold values to these parameters. In standpoint semantics, the syntax for defining a predicate allows additional arguments to be attached to it corresponding to semantic variation parameters.

Specifically, where a vague  $n$ -ary predicate  $V$  depends on  $m$  parameters we write it in the form:

$$V[p_1, \dots, p_m](x_1, \dots, x_n).$$

The following example illustrates the use of this syntax:

$$\text{Tall}[tall\_thresh](x) \equiv_{def} height(x) > tall\_thresh$$

Bennett et al. [12, 13] proposed to apply standpoint semantics to ground vague terms of a geographic ontology upon a spatial dataset, which involved performing geometric analyses and data segmentation. For instance, by defining standpoints for the predicates elongation and expansiveness, it was possible to make qualitative distinctions between water features, such as rivers (elongated) and lakes (expansive).

### 2.8.3 Handling Vagueness in Geography

Section 2.5 discussed the importance of providing ways of representing the numerous interconnections existing between field-based and object-based views. However, several issues may arise when manipulating these relationships if granularity and the phenomena of vagueness are not disregarded. First, some count nouns (e.g., a ‘cat’, a ‘table’) have reasonably precise boundaries and consequently can be depicted and assigned as a detached object straightforwardly.

On the other hand, the delimitation of some other count nouns (e.g., a ‘mountain’, a ‘river’) rely on human judgements and therefore requires an appropriate method of handling vagueness to provide the most suitable object representation according to the problem in hand. For example, a mountain can be characterised based on its height relative to a given point taken as reference (i.e., field values), such as the sea level or the nearest flat surface (in this case flatness is another vague concept and therefore should also be associated to a range of precisification values).

Rivers are similarly problematic. As illustrated by Bennett [9], the same section of an extended water body can be interpreted either as a river section that is rather irregular in width and includes a number of bulges; or as a water body consisting of three lakes connected by short rivers. We have also seen that adjectives (e.g., short, tall) are classical examples of a vague terminology. Thus, if we are to develop a model of geographic phenomena in which adjectival classifications change over time (e.g., a mountain can become

higher or lower over the years), methods of handling vagueness should be considered as significant component of such model.

Second, issues related to vagueness are also raised when performing operations between fields and objects. The basic example is that precisification values for vague features (e.g., mountains) can be originated from fields (e.g., a field containing values of elevation). A similar problem is observed when creating objects from objects. For instance, different forests may be characterised according to the distinct tree densities, as illustrated in Figure 2.3.

Moreover, if it is considered that a forest may be composed by *detached forested regions* (which may be separated by rivers, for example), then a related problem is to specify precisification values to define the regions which belong to the same forest (i.e., regions which should be considered as elements of an aggregate). Note that these examples involving forests can also be regarded as a matter of spatial granularity. Furthermore, if we take into account the temporal dimension (in which trees may be cut down, roads can be built crossing forests, land coverage may change, amongst numerous other transformation which can be observed in geographic space), even more interpretations may arise regarding the identification of the set of objects participating in a certain process instance or the exact spatial extent in which a process goes on.

Important issues regarding the conceptualisation of geographic events and processes are how to define the relation between their types and particular instances, and how to associate specific spatial and temporal boundaries with such instances. Vagueness, granularity and aggregation are intrinsically interconnected and defining an appropriate model of handling them is crucial to provide an appropriate interpretation of dynamic geographic phenomena. To illustrate how these concepts affect the delimitation of the spatial and temporal boundaries of a process, let us suppose a region  $R$  which undergoes urbanisation during a certain period of time  $I$ . Nonetheless, this supposition raises a number of issues: is this equivalent to saying that such a process is going on in every non urbanised sub-region of  $R$  at every instant inside  $I$ ? How long can this process be inactive for to maintain that this is the same process, and not an aggregate of several processes? How close should each urbanised sub-region of  $R$  be in order to justify that  $R$  is a homogeneous region, instead of an aggregate of several urbanised regions? And how close should these urbanised sub-regions be in order to say that there is only one process proceeding, and not several processes going on in parallel?

A characteristic of some geographic processes is their spatial and/or temporal vague boundaries. When dealing with a process such as a car crossing a bridge, it is easy to envisage its end, that is, when the car reaches the other side of the bridge, which is a

precise spatial boundary of the process. On the other hand, it is not always clear to define when or where an urbanisation process ceases. Additionally, once it is ceased, it may restart later. Moreover, the fact that a process may cease and then restart is controversial, and leads to the discussion of spatio-temporal continuity in geographic space, which has been investigated by Cohn and Hazarika [23]. In addition, there are theories in which a process is an entity which cannot contain spatial or spatio-temporal gaps [47], and others which allow a process to be regarded as active or inactive during distinct periods of time Galton [36].

From the discussion given in this section, it can be realised that an appropriate method of handling vagueness should be incorporated to any approach to representing dynamic geographic phenomena, in order to do justice to the countless ways in which reality may be sampled and observed. Standpoint semantics appears as a simple and flexible solution to deal with vagueness in the geographic domain.

## 2.9 Dealing with Information Granularity

In the geographic domain, vagueness is often related to the level of granularity at which spatial and temporal information are dealt with. Information granularity refers to the extent to which a piece of information can be broken down into smaller parts. In other words, it refers to the level of detail the information reflects. A higher level of granularity (or fine-grained information) means more detail, whilst a lower level of granularity (coarse-grained information) means less detail.

Section 2.8.3 discussed whether the same section of an extended water body should be interpreted either as a river section that is rather irregular in width and includes a number of bulges; or as a water body consisting of three lakes connected by short rivers. However, it should be noted that the same person could change their viewpoint if the spatial resolution of the geometrical representation is changed, for instance. Granularity may also affect the temporal perspective. For instance, a temporal series of spatial information may be produced by satellite images generated daily or monthly. Similarly, geo-sensors may provide values of measurements made for different regions at different times (and, for some problems, these pieces of information should be considered in conjunction). Vagueness in spatio-temporal information is directly associated with the granularity of the information considered to define the objective facts about a particular domain. That is, by changing the way information is provided to an interpreter (i.e., a person or a machine), their standpoints may also change for a particular situation.

Spatial granularity has been considered in GIScience and in the field of spatial data

mining, respectively, to improve the capability of the system to work appropriately with different map scales and to develop clustering algorithms which group spatial regions according to a set of characteristics (which include spatial and aspatial characteristics). Examples can be found in [3, 5, 93]. In GIS, granularity is usually treated as a matter of resolution, that is, the finer is the granularity, the larger the map scale is. However, the meaning of granularity within a semantic model also concerns the level of detail at which reality is sampled and observed [45]. Granularity affects not only space [75], but also affects time [31] and both simultaneously [17].

Three important concepts associated with granularity are grain, extent and frequency. Extent is concerned with the temporal duration over which a phenomenon operates and with the spatial size of a phenomena [57]. For instance, ‘continental glaciers operate over a much larger extent, both spatial and temporal, than thunderstorms’ [72, p. 14]. Grain refers to the level of detail at which data is recorded. ‘For example, the spatial grain of a remotely sensed image is the size of each pixel in its relation to the patch of the Earth it represents’ [72, p. 14]. ‘Frequency is traditionally defined as the number of cycles a phenomenon completes within a specified time interval. For example, the movement of a glacier occurs at a much lower frequency than the ephemeral cusp formation at a beach’ [72, p. 15]. In this context, low and high frequencies are referred as slow and fast behaviours, respectively.

In geography, vagueness is intimately related to the granularity of spatial and temporal information, so that different interpretations may arise depending upon the level of granularity at which dynamic entities of geographic space (e.g., events, processes) are observed. The work of this thesis aims to interrelate these concepts within the representational approach to geographic events and processes, in order to provide a logical framework which can accommodate multiple interpretations for a given geographic phenomena.

## 2.10 Spatial Aggregates

Aggregation or agglomeration [81] consists of bringing together a group of individual objects so that they can be considered as a single object. The criteria for collecting these objects can be based on many factors, such as spatial proximity, causal interaction, coherent motion, shared history or administrative fiat [34]. Galton [34] underlines that ‘some of these involve time and therefore would not be available in a purely static model’ (p. 03).

Spatial aggregates are considered in this work as a form of representing a group of spatial objects which are disconnected in space, but that should be regarded as a single element within the interpretation of a geographic phenomenon. An example of the use of

aggregates for analysing geographic phenomena is described by Steenbergen et al. [83], who examine the evolution of mussel beds between spring and winter. They argue that a mussel bed consists of a collection of smaller patches, and that a method of determining the group of patches that should be regarded as a particular bed is essential for the interpretation of the phenomena. There are a number of other geographic features which could be represented as a set of disconnected regions. For instance, a forest might be crossed by rivers and highways and still be regarded as the same forest.

## 2.11 Ontology Grounding

Grounding an ontology means establishing an explicit link between ontology and data. A characteristic of ontologies of geographic domain is that they are likely to contain concepts which can be effectively grounded upon data. For example, as discussed by Scheider et al. [74], the abstract concept of ‘depth’ in the geographic sub-domain of ‘water networks’ could be defined in terms of water level measurements provided by a gauge. Then other concepts such as ‘shallow’ and ‘deep’ might be defined in terms of ‘depth’. These terms, in turn, might be used in the definition of rivers, lakes, amongst others.

Furthermore, as noted by Jakulin and Mladenic [53], ontology grounding ‘is especially suitable for problem domains where extensive data is available, and where it would be time consuming to manually convert unstructured data into structured meta-data’ (p. 01), which is therefore applicable to the geographic domain. However, as pointed out by Bennett et al. [12], ‘the process of grounding an ontology upon data requires work at multiple levels, both in consideration of what predicates need to be grounded and how the data can be represented’ (p. 06).

‘Grounding gives meaning to ontological primitives by relating them to qualities outside the symbol system, and thus stopping infinite regress’ [74, p. 01]. Approaches to grounding geographic ontologies have been already proposed. For instance, Bennett et al. [13] presented an approach to grounding vague geographic terms (e.g., river, lake) based on geometric characteristics of water bodies (e.g., linearity, expansiveness). Scheider et al. [74] suggested to ground symbols for qualities (e.g., depth of a lake) by defining them from perceptual/observable primitives (e.g., ‘length of a vertically aligned path from the water surface to the bed of a particular water body’ [74, p.02]).

In this work, ontology grounding is considered not only as a method of linking primitive symbols to elements of data. The grounding layer should also allow high level concepts to be defined in terms of the primitive ones, that is, without concerns about the data structure. For instance, within an arbitrary geographic ontology, a primitive symbol

for ‘proximity’ could be grounded upon data elements consisting of geographic points (pairs of coordinates), so that higher level concepts, such as ‘neighbourhood’, could be defined without any reference to geographic coordinates. Similarly, in the logical framework developed in this thesis, a number of primitive symbols are determined, so that high level concepts representing dynamic elements of geographic space can be defined without concerns about the data structure.

## **2.12 Summary**

This chapter has summarised the background required for this thesis. The main approaches to qualitative spatial and temporal reasoning were presented. The representation of objects and fields was considered. Semantic models for events and processes were discussed, both in the general sense and within the geographic domain. The problems of handling vagueness and granularity were considered. The notion of spatial aggregates was introduced. Finally, an overview on the problem of ontology grounding was given.

# Chapter 3

## Representing Spatio-temporal Data and Modelling Changing Geographic Features

---

### 3.1 Introduction

This chapter presents a logic-based approach to representing geographic spatio-temporal data and to representing geographic features whose spatial extensions are subject to changes over time. Section 3.2 gives an overview of the problem. Then Section 3.3 introduces some elements of the logical framework which are necessary for understanding the approach described in this chapter<sup>1</sup>. Following this, Section 3.4 introduces the approach to representing spatio-temporal data of geographic domain. Then Section 3.5 describes the axiomatisation specified to determine integrity constraints and to derive implicit data. Then the method of modelling changing geographic features by establishing an explicit link between data and ontology layers is presented in Section 3.6. Finally, Section 3.7 summarises the most important points discussed in this chapter.

This chapter presents the main fundamentals and conceptual aspects of the proposed approach. Logical descriptions of the developed formalism are presented. However, relevant algorithms will be discussed in Chapter 5, where the system implementation is de-

---

<sup>1</sup>A complete description of the the logical framework will be given in Chapter 4.

scribed.

## 3.2 Overview of the Problem and Motivation

Of particular interest in this work is the modelling of geographic phenomena which can be described in terms of *changes affecting the spatial extensions of geographic features*, such as forests, deserts, rivers and oceans. Hence, special attention is drawn to the representation of geographic features, since an appropriate modelling of these entities can provide a natural way of defining other dynamic elements of geographic space, such as events and processes. Geographic features have some of the characteristics described in the literature for *material objects* (e.g., [38, 58]). In particular, geographic features are discrete individuals with well-defined spatial-temporal extensions and can change some of their parts while keeping their identity. This work aims to produce a formalism whose concepts are grounded upon the data, and a number of representational difficulties are encountered in modelling such a kind of dynamic entity by establishing an explicit link between the data and logical levels.

Many spatio-temporal models have been proposed to represent dynamic geographic space, and these models often include entities whose semantics corresponds to the concept of spatial objects conceived here (e.g., [16, 18, 51, 84]). Nevertheless, most existing spatio-temporal models do not usually address issues relating to the representation of these conceptual entities at the data level. Thus, in order to design a system that implements such a formalism, it is often assumed that the objects which inhabit the model are spatially well defined in data (such as a desert represented as a precise polygon). However, as discussed in Section 2.5, geographic data can be provided in a variety of other forms, such as *fields*. In this case, objects could be inferred from fields, as suggested by Galton [33]. For instance, the boundaries of objects representing deserts can be determined from field data consisting of average precipitation rate measurements distributed on a grid-based spatial dataset.

Furthermore, there are a number of spatial and spatio-temporal datasets that consist of a set of precise demarcations of portions of the earth's surface (e.g., using polygons or multi-polygons), but whose portions do not necessarily represent spatial extensions of particular objects. For instance, remote sensing techniques provide means of demarcating boundaries of different land use or land cover units by processing satellite imagery containing multiple bands of the electromagnetic spectrum, and an increased number of datasets have been made available as an output of studies in this field. Although this type of dataset does not often describe spatial objects explicitly, they contain meaningful

data that can be used to infer many different objects (e.g., to infer ‘farms’ from data on cultivated and pasture areas, or to infer rivers from data on water bodies). Nonetheless, such kinds of inference mechanisms are not yet fully developed and further investigations are still required. Thus the contributions of this work include the provision of a method of inferring the spatial extension of distinct objects (i.e., geographic features) at different times so that the spatial changes they undergo over time can be formally modelled.

Particular focus is placed on geographic features whose spatial extension at a given time corresponds to a portion of the earth’s surface. Additionally, it is considered that these portions of the earth’s surface can be described in terms of certain semantic characteristics. There are many different ways in which the earth’s surface can be described, which depends on the objectives pursued in obtaining such a description. These descriptions may vary, for instance, in terms of the range of elements represented. For example, the same portion of the earth’s surface can be described either in terms of land use or in terms of average precipitation rates. In GIS, these distinct conceptual classes of elements are usually represented using separate map layers, which may spatially overlap each other. Descriptions of geographic space may also vary in terms of the level of details adopted. For instance, whilst one might describe a certain region as ‘forested’, the other might need to describe the same region in a greater level of detail, by specifying sub-regions covered by different types of vegetation.

Moreover, numerous topological and mereological relationships may hold between the types of coverages which constitute geographic space, and a variety of constraints could be specified to restrict the set of relations which can or cannot hold between distinct types of coverage. For instance, it could be defined that a ‘forested’ region can be composed by any region containing types of vegetation  $v_1$  or  $v_2$ , but may never be composed by a region containing a type of vegetation  $v_3$ . Furthermore, it could be said that an ‘urbanised’ region can never overlap a ‘forested’ region. The approach described here includes a method of defining these relationships and constraints, which constitutes the basis of a mechanism for inferring geographic features and tracking their changes over time. In addition, this approach intends to provide representational flexibility, so that a wide range of geographic elements can be identified by inferences performed upon a simple and uniform storage structure. Furthermore, this approach aims to facilitate the linking between the conceptual and data layers, by defining a set of primitive conceptual elements in terms of the data context, so that higher level concepts can be defined in terms of these primitive ones, without any concern about data structure.

Spatial and spatio-temporal datasets are usually limited to a particular area and to a particular range of elements of interest. Hence, by combining information from multiple

datasets (e.g., one produced by processing satellite imagery and another by processing data from other geo-sensors, such as temperature and salinity), a wider range of objects can be inferred and a higher accuracy can potentially be achieved. However, when the temporal dimension is taken into account, that is, when the spatial changes performed by the objects which inhabits the model are relevant, a number of issues are raised. Distinct datasets may contain distinct information about different portions of geographic space, in distinct periods of time and at different temporal granularities. For example, while one dataset may consist of data about rivers within a portion of space  $r_1$ , collected weekly between the calendar years 2005 and 2012, another dataset might contain data about lakes within a portion of space  $r_2$  (which partially overlaps  $r_1$ ), collected monthly between 2001 and 2008. Besides the other characteristics already described, the approach presented in this chapter can also be used as a mechanism for facilitating the integration of spatio-temporal data originated from distinct sources and based on different temporal granularities, allowing the interpretation of high level concepts to be supported efficiently thorough continued updates in the database.

### 3.3 Introducing the Logical Framework

Formal descriptions are used in this thesis to present a number of characteristics of the proposed theory. Most of these formalisms are presented in this chapter and in Chapter 4. The formalism presented here is described in terms of definitions and axioms in first-order logic, where free variables are implicitly universally quantified with maximal scope. These definitions and axioms are indexed by D and A, respectively.

A complete description of the *logical framework* developed to represent and reason about geographic events and processes will be given in Chapter 4. However, this chapter introduces some elements of the logical language  $\mathfrak{R}$  used within that framework. These elements are particularly relevant to the understanding of the proposed approach to modelling geographic features, and to the comprehension of the motivations behind the proposed method of representing spatio-temporal data. Namely, the elements introduced in this chapter are time *instants* and *intervals*; *spatial regions* and their *coverages*; and *geographic features* and their *types*.

#### 3.3.1 Space

The Region Connection Calculus (RCC) [69] is employed as the theory of *space*. An overview on the RCC relations mentioned in this thesis is given in Section 2.2. Spatial

regions are used here to represent portions of the earth's surface under some specified projection, and explicit variables  $r_i$  are used to denote spatial regions. These variables can be quantified over in the usual way (e.g.,  $\forall r[\phi(r)]$  or  $\exists r[\phi(r)]$ ).

The logical language  $\mathfrak{R}$  also includes a number of *functions* to exchange information between variables and to perform spatial operations between regions. The complete set of functions will be presented in Chapter 4. However, the following *auxiliary functions* are used in definitions presented in this chapter and therefore are introduced now.

- union :  $(\mathcal{V}_r \times \mathcal{V}_r) \rightarrow \mathcal{V}_r$  which returns a spatial region that corresponds to the spatial union of a pair of spatial regions.
- distance :  $(\mathcal{V}_r \times \mathcal{V}_r) \rightarrow \mathbb{R}$ , which returns a non-negative number representing the 2-dimensional Cartesian minimum distance between two regions in projected units.
- concave-hull :  $(\mathcal{V}_r \times \mathcal{V}_r) \rightarrow \mathcal{V}_r$ , which returns a concave region that encloses the two specified regions. The concave hull of a set of geometries represents a possibly concave geometry that encloses all geometries within the set. One can think of the concave hull as the geometry obtained by 'vacuum sealing' a set of geometries. Many different algorithms for calculating concave hulls are currently available, and they normally work based on the value of a parameter. Roughly, this parameter corresponds to the target percent of area of convex hull the algorithm solution will try to approach before giving up or exiting. Different algorithms often compute different results (even when equal values are assigned to corresponding parameters). The concave hull algorithm used to implement the system prototype is described in Chapter 5.

### 3.3.2 Spatial Region Coverages

A spatial region can be described in terms of characteristics of the portion of the earth's surface it represents. The logical language  $\mathfrak{R}$  includes a special type of element to denote a certain semantic description which can be associated with spatial regions. These descriptions are called here 'spatial region *coverages*', and are denoted by explicit variables  $c_i$ . The meaning of 'coverage' employed here is not restricted to land coverages. It can also denote qualities which can be measured (by sensors or by human observation), such as 'urbanised', 'arid', 'temperature > 10 °C', 'water covered', or 'heavily populated'. The way coverages can be associated with spatial regions will be further clarified throughout this chapter.

### 3.3.3 Time

It is assumed a total linear reflexive ordering on *time*, and explicit variables  $t_i$  and  $i_i$  are used to denote *time instants* and *proper intervals*, respectively. A time interval  $i$  is considered a *proper* interval if the time instant which represents its beginning precedes the time instant denoting its end (i.e.,  $b(i) \prec e(i)$ ). These temporal variables can also be quantified over in the usual way.

The following *functions* are used to exchange information between these temporal variables:

- $b(i)$ , which returns an instant  $t$  corresponding to the beginning of an interval  $i$ ;
- $e(i)$ , which returns an instant  $t$  corresponding to the end of an interval  $i$ .

Time instant variables can be compared by equality ( $t_1 = t_2$ ) and by ordering ( $t_1 \prec t_2$  and  $t_1 \preceq t_2$ ) operators. Allen temporal relations [1, 2] are employed between time intervals. These relations are described in Section 2.3. The relations  $\text{In}(i_1, i_2)$  and  $\text{In}(t_1, i_2)$  are also defined, meaning that a time interval  $i_1$  (or time instant  $t_1$ ) is inside a proper time interval  $i_2$ . These relations are defined as shown below, in Definitions D3.1 and D3.2.

$$\text{D 3.1} \quad \text{In}(i_1, i_2) \equiv_{\text{def}} (\text{Starts}(i_1, i_2) \vee \text{During}(i_1, i_2) \vee \text{Finishes}(i_1, i_2) \vee \text{Equals}(i_1, i_2))$$

$$\text{D 3.2} \quad \text{In}(t, i) \equiv_{\text{def}} b(i) \preceq t \preceq e(i)$$

The logical language  $\mathfrak{R}$  also includes other *functions* to exchange information between variables and to perform spatial operations between temporal variables. These functions will be presented in Chapter 4.

### 3.3.4 Geographic Features

*Geographic features* will be regarded as a particular kind of *endurant entity*, and therefore they are able to *undergo change* over time. Special attention is paid to changes affecting their spatial extensions. Of particular interest are geographic features whose spatial extension at a given time instant  $t$  can be modelled as the maximal well-connected region of some particular coverage at  $t$ . The expression ‘well-connected region’ is used here in agreement with the discussion and definitions given in Cohn et al. [24]. Examples of

geographic features are forests (which can be regarded as the maximal extension of a certain type of vegetation), deserts (which can be defined based on the level of precipitation) and sea (represented as the maximal extension of water body over a specified level of salinity). Section 3.6 further discusses the fundamentals underlying the representation of geographic features.

In the logical language  $\mathfrak{R}$ , variables  $f_i$  and  $u_i$  are used to denote, respectively, individual geographic features (e.g., Amazon rainforest, Atlantic ocean) and feature types (e.g., sea, forest). This language also includes *functions* to exchange information between features and other types of variables. These functions will be presented in Chapter 4. However, the following function is introduced now as it is mentioned in logical definitions presented in this chapter.

- $\text{ext}(f, t)$ , which returns the spatial region corresponding to the spatial extension of a feature  $f$  at time instant denoted by  $t$ .

Relevant predicates relating to the representation of geographic features will be presented in Section 3.6.

### 3.4 Spatio-Temporal Attributed Regions

This section presents a logic-based approach to modelling spatio-temporal data. This approach, named *STAR* (Spatio-temporal Attributed Regions), provides a way of representing spatial regions in association with their respective coverages at different times, and a mechanism for performing inferences with respect to data, based on given semantic relationships between regions. A system which implements this approach becomes capable of inferring the spatial extension of geographic features at different times from data describing arbitrary spatial regions and their coverages.

In the *STAR* model, the spatio-temporal data are stored as triples of the form  $\langle a, g, s \rangle$ , which corresponds to the fact that attribute  $a$  holds for geometry  $g$  at time instant denoted by timestamp  $s$ . Currently, geometries are restricted to 2-dimensional simple polygons (also called Jordan polygons), which are those polygons whose boundary does not cross itself. Here the term ‘polygon’ refers to a plane figure that is bounded by a closed path, composed by a finite sequence of straight line segments (i.e., by a closed polygonal chain). Therefore, this contrasts with the view held by some mathematicians that a polygon corresponds to a shape made up by those straight line segments (which does not include the enclosed region).

A wide range of attributes can be associated with geometries. They can be used to describe either types of region coverage (e.g., ‘forested’, ‘arid’, ‘water covered’) or types of geographic features (e.g., ‘ocean’, ‘desert’, ‘forest’). Polygons may denote either spatial regions or spatial extensions of geographic features, depending on the type of attribute they are associated with. Those triples are represented at the logical level as facts of the Knowledge Base (KB) by using the predicate *Spatio-temporal Attributed Region*  $\text{Star}(a, g, s)$ . For simplicity, a fact represented by the predicate  $\text{Star}(a, g, s)$  is referred to in this thesis as ‘a *Star* fact’ or just ‘a *Star*’.

A *Star* can either be asserted explicitly or resulting from inferences performed by the system. If a given spatio-temporal dataset consists of meaningful data which can accurately and consistently describe spatial extensions of geographic features at different time instants, facts of the form  $\text{Star}(a, g, s)$  are therefore regarded as explicitly asserted in the KB in association with feature attributes. However, as discussed in Section 3.2, this kind of dataset is not frequently produced and made available.

On other hand, facts representing spatial extensions of geographic features (i.e., a *Star* associated with feature attributes) can be inferred from facts denoting spatial regions (i.e., a *Star* associated with coverage attributes), as will be described in detail in Section 3.5.2.

### 3.4.1 Types of Attributes

There are many different ways in which an attribute can be used to describe a spatial region with respect to a time instant. Since an attribute  $a$  is treated as a special kind of entity, sortal predicates can be used to classify attributes and first-order formulae to axiomatise semantic characteristics and inter-dependencies of attributes.

The STAR model currently supports a geographic KB in which the following kinds of attributes are recorded.

- $\text{CAtt-Hom}(a)$  — *homogeneous coverage attributes* — are applied to denote spatial regions which are regarded as covered by a single type of coverage (e.g., ‘water covered’, ‘forested’, ‘paved’).
- $\text{CAtt-Het}(a)$  — *heterogeneous coverage attributes* — are employed to denote spatial regions which may contain multiple types of coverages (e.g., ‘urbanised’, ‘agricultural’).
- $\text{FAtt-Sim}(a)$  — *simple feature attributes* — are applied to denote geographic features, where every region which is part of it must have the same coverage (e.g., ‘ocean’, ‘road’, ‘desert’).

- $\text{FAtt-Com}(a)$  — *compound feature attributes* — are applied to denote geographic features which normally contain regions with different coverages (e.g., ‘city’, ‘park’, ‘beach’).

These attributes are *mutually exclusive*. The actual denotation of these distinct types of attributes depends on the intended application. For example, an attribute named ‘forested’ can be employed to denote either a homogeneous or a heterogeneous type of coverage. The former might be applied when different types of vegetations are not relevant to the problem at hand, whilst the latter might be employed in association with several homogeneous coverage attributes denoting types of vegetation. The spatial extension of a geographic feature at a certain time instant can be asserted explicitly or can be inferred as a maximal well-connected region of some particular coverage. For example, a forest can be inferred as a maximal well-connected region whose coverage is regarded as ‘forested’.

More general predicates are also used to describe types of attributes. The predicate *coverage attribute*  $\text{CAtt}(a)$  denotes any type of coverage attribute, either homogeneous or heterogeneous. The predicate *feature attribute*  $\text{FAtt}(a)$  denotes any type of geographic feature attribute, either simple or compound. Finally, the most general predicate *attribute*  $\text{Att}(a)$  denotes any type of attribute. Formal descriptions of these predicates are given in Definitions D3.3, D3.4 and D3.5, respectively.

$$\text{D 3.3} \quad \text{CAtt}(a) \equiv_{def} \text{CAtt-Hom}(a) \vee \text{CAtt-Het}(a)$$

$$\text{D 3.4} \quad \text{FAtt}(a) \equiv_{def} \text{FAtt-Sim}(a) \vee \text{FAtt-Com}(a)$$

$$\text{D 3.5} \quad \text{Att}(a) \equiv_{def} \text{CAtt}(a) \vee \text{FAtt}(a)$$

### 3.4.2 Formal Description

In the STAR model introduced above, the spatio-temporal data is structured as follows:

$\mathcal{D} \subseteq A \times G \times S$ , where:

- $A$  is the set of attributes;
- $G$  is the set of geometries;
- $S$  is the set of timestamps;

- A *datum* is a tuple assuming the form  $\langle a, g, s \rangle$ , where  $g$  is a polygon,  $a$  is an attribute and  $s$  is a timestamp. These data elements are stored as asserted facts using the predicate *Spatio-temporal Attributed Region*  $\text{Star}(a, g, s)$ <sup>2</sup>.

A formal model  $\mathcal{G}$  of a geographic dataset can be specified as follows.

$\mathcal{G} = \langle \mathbb{R}^2, \langle T, \leq \rangle, A, \mathcal{D} \rangle$ , where:

- $\mathbb{R}^2$  is the real plane, which represents a portion of the earth's surface under some specified projection<sup>3</sup>.
- $T$  is the set of all time instants over the time sequence  $\langle T, \leq \rangle$ , where  $\leq$  is a total linear order over  $T$ .
- $A$  is a set of geographic attributes.
- $\mathcal{D} \subseteq A \times \text{Poly}(\mathbb{R}^2) \times T$  represents the geographic attributed data as a subset of all possible triples of the form  $\langle a, g, s \rangle$ , where  $\text{Poly}(\mathbb{R}^2)$  is the set of 2-dimensional simple polygons over  $\mathbb{R}^2$ .

### 3.5 Axiomatisation

The axiomatisation specified in the STAR model comprises two main groups of axioms. The first determines a number of *integrity constraints*, which constrain the *Star* facts that can be asserted in the KB. The other set of axioms specifies a variety of *derivation rules* which can be applied on facts stored in the KB to derive new facts by means of logical inferencing. Since the first set of axioms defines how different facts can co-exist in the knowledge base, it also constrains the inference mechanism, that is, their axioms restrict the ways in which facts can be implicitly generated.

As discussed in Section 3.3, the logical framework developed in this work includes variables to represent time instants, time intervals, spatial regions, region coverages, geographic features and feature types, which are mapped to elements of the domain by appropriate assignment functions (presented in Chapter 4). Some of these variables can

<sup>2</sup>In the implementation of this model, the data is stored in a database within a spatial DBMS (Database Management System), and the predicate  $\text{Star}(a, g, s)$  is implemented so that each solution for this predicate represents either a record stored in the database or an implicit fact derived by the system. Further details of the system implementation is presented in Chapter 5.

<sup>3</sup>Clearly, one might want to use a different coordinate system or a 2.5D surface model. For simplicity we just assume that the space is modelled by  $\mathbb{R}^2$ ; however, this could easily be changed without modification to the rest of the semantics.

be directly linked to data elements that are associated through the *Star* predicate: region coverages and feature types correspond to coverages and feature attributes, respectively; whereas spatial regions correspond to geometries (polygons) and time instants correspond to timestamps. Hence, to help establish the explicit link between the data and logical levels, the framework also includes predicates of the form  $\text{Star}(c, r, t)$  and  $\text{Star}(u, r, t)$ , that are mapped to appropriate facts of the form  $\text{Star}(a, g, s)$  within the STAR model.

Given this direct mapping between data elements and primitive concepts of the logical framework, some derivation rules are described in this section in terms of elements of the logical framework, although definitions and axioms actually specify relations between facts of the form  $\text{Star}(a, g, s)$ . Moreover, some definitions and axioms presented in this section ascribe to timestamps and geometries relations which actually hold between times and spatial regions, respectively (e.g., Allen's and RCC relations). Furthermore, spatial and temporal relationships between *Stars* (e.g., '*Star a* is part of *Star b*' or '*Star a* is before *Star b*') are mentioned in the text referring to the relationships that hold between their geometries and timestamps, respectively.

Furthermore, the predicate  $\text{A-Star}(a, g, s)$  is used to indicate that the fact  $\text{Star}(a, g, s)$  is *explicitly asserted* in the knowledge base, whereas the truth of  $\text{Star}(a, g, s)$  is determined by the semantics of attribute  $a$  and the geographic characteristics of the geo-referenced polygon  $g$ , whether or not it is actually asserted in the knowledge base. Consequently, the Axiom A3.1 is specified to assure that  $\text{Star}(a, g, s)$  is true if the corresponding fact (explicitly asserted) is true.

$$\text{A 3.1} \quad \text{A-Star}(a, g, s) \rightarrow \text{Star}(a, g, s)$$

### 3.5.1 Integrity Constraints

The first axiom presented here is specified to ensure that any fact of the form  $\text{Star}(a, g, s)$  actually relates the correct types elements: attributes, geometries and timestamps. This is shown in Axiom A3.2, where  $\text{Att}(a)$  ensures the attribute  $a$  has been previously asserted (explicitly) in the KB; the predicate  $\text{Polygon}(g)$  is employed to assure that  $g$  is a two-dimensional simple polygon; and  $\text{Timestamp}(s)$  assures  $s$  represents a timestamp in ISO 8601 format (e.g., '2011-03-30 02:15:00')<sup>4</sup>.

$$\text{A 3.2} \quad \text{Star}(a, g, s) \rightarrow \text{Att}(a) \wedge \text{Polygon}(g) \wedge \text{Timestamp}(s)$$

---

<sup>4</sup>For convenience, the ISO 8601 format has been chosen for implementing this model; however, this could easily be changed without modification to the rest of the semantics. Further implementation details are discussed in Chapter 5.

Beyond the facts which can be explicitly asserted using the predicate  $\text{Star}(a, g, s)$ , other facts can also be *explicitly asserted* using logical relations between pairs of attribute types. By specifying how two types of attributes are related to each other, it is possible to define derivation rules which determine how two *Stars* associated with these attributes can co-exist in space and time.

The *asymmetric* logical relations *Can Contain*  $\text{CC}(a_1, a_2)$  and *Must Contain*  $\text{MC}(a_1, a_2)$  are used to specify, respectively, the cases where part-hood relations *can* or *must* hold between *Stars* associated with attributes  $a_1$  and  $a_2$ .

The semantics of CC and MC can be described as follows:

- A fact of the form  $\text{CC}(a_1, a_2)$  is meant to be understood as saying that there may exist instances in which a region associated with  $a_2$  is part of a region associated with  $a_1$ . For example,  $\text{CC}(\textit{urbanised}, \textit{paved})$  means that there may exist instances in which urbanised regions have paved regions as their part. In addition, if  $\text{CC}(\textit{urbanised}, \textit{paved})$  does *not* hold, means that there may exist *no* instance in which a paved region is part of an urbanised region. It should be noticed that  $\text{CC}(\textit{urbanised}, \textit{paved})$  *does not mean* that there exists at least one instance in which a paved region is part of an urbanised region.
- A fact of the form  $\text{MC}(a_1, a_2)$  is meant to be understood as saying that every region associated with  $a_1$  must contain a region associated with  $a_2$ . For example, the fact  $\text{MC}(\textit{urbanised}, \textit{built-up})$  means that every urbanised region has a built-up part. It should be noticed that  $\text{MC}(\textit{urbanised}, \textit{built-up})$  *does not mean* that every built-up region is part of some urbanised region; and *neither* that there exists at least one instance in which a built-up region is part of an urbanised region.

Hence, since the relations CC and MC establish *possibility* and *obligatoriness*, respectively, it can be said that if  $\text{MC}(a_x, a_y)$  holds, then  $\text{CC}(a_x, a_y)$  also holds. This is therefore specified in Axiom A3.3. These relations will be explained in more detail throughout this section.

$$\text{A 3.3} \quad \text{MC}(a_1, a_2) \rightarrow \text{CC}(a_1, a_2)$$

These relations are key instruments for deriving implicit facts, such as *Stars* inferred from explicit facts representing other *Stars* which contain (or are part of) them. A number of rules which enable the system to derive implicit data are described in detail in Section 3.5.2. On the other hand, this section focusses on the axioms specified to assure the

integrity and consistency of facts explicitly asserted in the knowledge base, in order to prevent the inference mechanism from performing anomalous inferences and from generating contradictory facts.

Facts using  $CC(a_1, a_2)$  and  $MC(a_1, a_2)$  relations can be explicitly asserted when attributes  $a_1$  and  $a_2$  are (respectively) of types:

- heterogeneous and homogeneous coverage attributes; or
- simple feature attributes and homogeneous coverage attributes; or
- compound feature attributes and heterogeneous coverage attributes.

The first case above ensures that a heterogeneous region whose coverage type is denoted by an attribute  $a_1$  can contain a homogeneous region whose coverage type is denoted by an attribute  $a_2$ ; Similarly, the second case ensures a simple feature denoted by an attribute  $a_1$  can contain a homogeneous region whose coverage type is denoted by an attribute  $a_2$ . The last case ensures that a compound feature whose type is denoted by an attribute  $a_1$  can contain a heterogeneous region whose coverage is denoted by an attribute  $a_2$ .

Thus Axiom A3.4 is specified to restrict the types of attributes which can be related using these relations. In addition, Axiom A3.4 specifies that these relations can be regarded as *reflexive* if they are used to relate a region coverage attribute to itself. However, facts self-relating coverage attributes do not need to be asserted explicitly, as the relation between them is already specified in Axiom A3.5. This reflexivity allows the system to consider a given region  $r_1$  as having the same coverage of a region  $r_2$ , if  $r_1$  is a sub-region of  $r_2$ , as described in detail in Section 3.5.2.

$$\begin{aligned}
 \text{A 3.4 } CC(a_1, a_2) \rightarrow & ((a_1 = a_2) \wedge \text{CAtt}(a_2)) \vee \\
 & (\text{CAtt-Het}(a_1) \wedge \text{CAtt-Hom}(a_2)) \vee \\
 & (\text{FAtt-Sim}(a_1) \wedge \text{CAtt-Hom}(a_2)) \vee \\
 & (\text{FAtt-Com}(a_1) \wedge \text{CAtt-Het}(a_2))
 \end{aligned}$$

$$\text{A 3.5 } \text{CAtt}(a) \rightarrow \text{MC}(a, a)$$

For convenience, the fact of the form  $CC(a_x, a_y)$  is described here as ‘attribute  $a_x$  can contain attribute  $a_y$ ’, referring to the part-hood relation which can hold between spatial regions associated with these attributes.

A fact of the form  $CC(a_x, a_y)$  is asserted to determine that a part-hood relation *can* hold (but does not necessarily hold) between *Stars* associated with attributes  $a_x$  and  $a_y$ . In other words, a fact of the form  $CC(a_x, a_y)$  does *not* mean that a *Star*  $x$ , associated with attribute  $a_x$ , only exists if it spatially contains a *Star* associated with attribute  $a_y$ . For example, a *forested* region might be modelled as composed by any combination of vegetations of types  $v1$ ,  $v2$ ,  $v3$ , and  $v4$ . In this case, 4 facts  $CC(\textit{forested}, v1)$ , ...,  $CC(\textit{forested}, v4)$  should be explicitly asserted. However, a *forested* region could exist without any region covered by vegetation  $v3$ , for example.

On the other hand, a given *Star*  $x$  (associated with attribute  $a_x$ ) only exists if, for any *Star*  $y$  (associated with attribute  $a_y$ , where  $a_y \neq a_x$ ) which is part of  $x$ , a fact of the form  $CC(a_x, a_y)$  is explicitly asserted. This integrity constraint is specified in Axiom A3.6. In the example above, this is to say that a fact specifying that a certain *forested* region exists in a certain spatio-temporal location and contains regions covered by vegetations  $v1$  and  $v2$  will only be true if facts using the relation  $CC$  are asserted relating attributes which describe these vegetation types and a forested region (e.g.,  $CC(\textit{forested}, v1)$ ,  $CC(\textit{forested}, v2)$ ).

$$\text{A 3.6} \quad (a_1 \neq a_2) \wedge \exists g_1 g_2 s [\text{Star}(a_1, g_1, s) \wedge \text{Star}(a_2, g_2, s) \wedge P(g_2, g_1)] \rightarrow CC(a_1, a_2)$$

The  $MC$  relation distinguishes from  $CC$  by the fact that if  $MC(a_x, a_y)$  holds, a *Star*  $x$  only exists if it spatially contains at least one *Star* associated with attribute  $y$ . This is specified in Axiom A3.7. For example, it might be coherent to assert that a forest must contain at least one area covered by vegetation  $v1$ , and that the remaining forested area might be covered by any combination of vegetations of types  $v1$ ,  $v2$ ,  $v3$ , and  $v4$ . On the other hand, the integrity constraint specified in Axiom A3.6 is applied to both  $CC$  and  $MC$  relations, without the need to specify other axioms. That is, given the existence of Axiom A3.3, the Axiom A3.6 will be satisfied if attributes  $a_1$  and  $a_2$  are related by facts using either  $CC$  or  $MC$ .

$$\text{A 3.7} \quad \exists a_1 [\text{Star}(a_1, g_1, s) \wedge MC(a_1, a_2) \wedge (a_1 \neq a_2)] \\ \rightarrow \exists g_2 [\text{Star}(a_2, g_2, s) \wedge P(g_2, g_1)]$$

Axiom A3.6 ensures that a *Star* can contain (or can be equals to) another *Star* associated with a different attribute only if these attributes are related by  $CC$ . Nevertheless, this axiom does not prevent a *Star* associated with a feature attribute from being proper part of

a *Star* associated with the same feature attribute. Since geographic features are regarded here as the maximal well-connected extension of a certain type of coverage, this situation should be prevented. For this reason, Axiom A3.8 specifies that two *Stars* associated with the same attribute can be a proper part of each other only in case this attribute is a region coverage attribute.

$$A\ 3.8\ \text{Star}(a, g_1, s) \wedge \text{Star}(a, g_2, s) \wedge \text{PP}(g_1, g_2) \rightarrow \text{CAtt}(a)$$

Figure 3.1 shows a simplified Entity-Relationship Diagram (ERD) where all edges denote *part-of* relations which may hold between instances of *Stars* associated with the attributes specified in the boxes. Therefore, the subset of possible instances of these relationships for a particular domain is determined by facts asserted using either CC or MC relations.

In this ERD, *Stars* associated with different attributes are represented as different types of entities. Numbers represent the cardinality of the relationship, which indicates the number of distinct instances of one entity which can be associated with an instance of the related entity. This cardinality represent the number of CC or MC facts which can be explicitly asserted to relate pairs of distinct attributes.

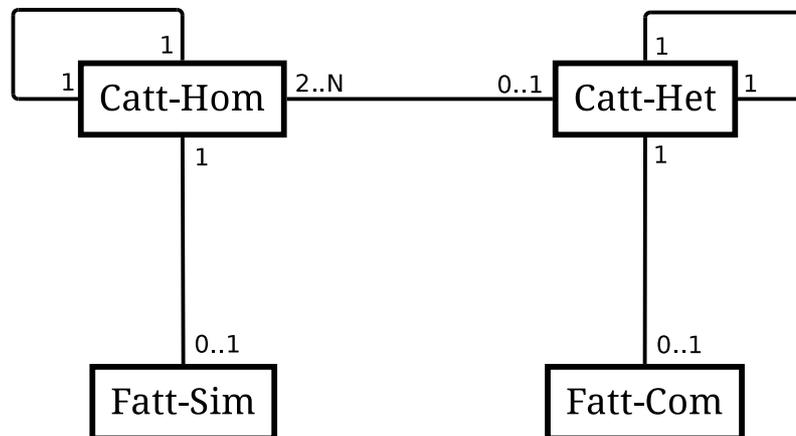


Figure 3.1: Cases where part-hood relations may hold between *Stars* associated with distinct types of attributes. Boxes represent an entity *Star* associated with the specified type of attribute. Numbers represent the cardinality of the relationship, which indicates the number of times an instance of one entity can be associated with instances of the related entity.

According to the ERD exhibited in Figure 3.1, each attribute denoting a simple ge-

ographic feature must be related to one (and only one) attribute representing a homogeneous type of region coverage. Similarly, attributes denoting compound geographic features must be related to one (and only one) attribute representing a heterogeneous type of region coverage. On the other hand, region coverage attributes can be related to only one geographic feature attribute; however, instances of this relationship may not exist. These constraints are specified in Axioms A3.9 to A3.11, as follows.

$$\text{A 3.9} \quad \text{FAtt}(a_1) \rightarrow \exists a_2[\text{CAtt}(a_2) \wedge \text{CC}(a_1, a_2)]$$

$$\text{A 3.10} \quad \text{FAtt}(a) \wedge \text{CC}(a, a_1) \wedge \text{CC}(a, a_2) \rightarrow a_1 = a_2$$

$$\text{A 3.11} \quad \text{FAtt}(a_1) \wedge \text{CC}(a_1, a) \wedge \text{CC}(a_2, a) \rightarrow a_1 = a_2$$

The ER diagram (Figure 3.1) shows that compound feature attributes cannot be directly related to homogeneous coverage attributes using CC or MC relations. This means that such features must be specified by relating homogeneous coverage attributes to a heterogeneous coverage attribute and then relating the latter to a compound feature attribute. Although compound features must be specified in terms of heterogeneous coverage attributes, a homogeneous region might still be regarded as *part of* a compound geographic feature. In this case, the attribute that represents the homogeneous coverage must be associated (by CC relation) with a heterogeneous attribute which, in turn, is associated with the attribute that represents the compound feature type.

Furthermore, the diagram shows that geographic features cannot be composed by other features, as a feature is regarded here as the maximal well-connected extension of a certain type of coverage. However, it can be observed that, if a given homogeneous coverage attribute is related to both a heterogeneous coverage attribute and a simple feature attribute, it would allow a compound feature to be composed by simple features. This situation is prevented by specifying Axiom A3.12.

$$\text{A 3.12} \quad \text{CAtt-Hom}(a) \wedge \text{CC}(a_1, a) \wedge \text{CC}(a_2, a) \rightarrow a_1 = a_2$$

Moreover, it can be seen in the diagram that attributes denoting homogeneous and heterogeneous region coverages are always related to themselves, meaning that regions associated with these attributes are composed by regions of the same type (see Axiom 3.5). However, as these relationships must hold for all coverage attributes, a subset of

possible instances of these self-relationships does not need to be determined by explicitly asserting facts using CC or MC relations. The importance of these self-relationships will be further discussed within the description of the derivation rules in Section 3.5.2.

The relationship between homogeneous and heterogeneous coverage attributes shown in the ERD (Figure 3.1) means that a heterogeneous coverage attribute must be related to at least two homogeneous types of coverages (Axiom A3.13) whilst there may be homogeneous coverage attributes which are related to no heterogeneous coverages.

$$\begin{aligned} \text{A 3.13 } \text{CAtt-Het}(a) \leftrightarrow \exists a_1 a_2 [ & \text{CAtt-Hom}(a_1) \wedge \text{CC}(a, a_1) \wedge \\ & \text{CAtt-Hom}(a_2) \wedge \text{CC}(a, a_2) \wedge \\ & a \neq a_1 \wedge a \neq a_2 \wedge a_1 \neq a_2 ] \end{aligned}$$

Although heterogeneous attributes must be related by CC or MC to at least two homogeneous attributes, there is no axiom restricting that a particular instance of a heterogeneous region, at a particular time instant, is covered homogeneously. For example, one might define a heterogeneous coverage attribute ‘agricultural’ to represent regions composed by the aggregation of regions with different cultivations, each of which represented by a different homogeneous coverage attribute. In this case, it would be admissible that one might wish to consider a region as ‘agricultural’ even if it contains only one type of cultivation over a certain period of time.

This assumption could be dropped by adding an axiom to specify a more strict sense of heterogeneity, in which at any one time instant a heterogeneous region should contain at least two sub-regions associated with distinct homogeneous attributes. However, this would impact the semantics of the axioms 3.4 and 3.5 as well as some derivation rules described in Section 3.5.2.

For convenience, the logical relation *Cannot Overlap*  $\text{CO}(a_1, a_2)$  is also defined. It relates two distinct attributes  $a_1$  and  $a_2$ , meaning that regions whose coverages are denoted by these attributes cannot overlap (spatially). For short, this is often described here as ‘attributes  $a_1$  and  $a_2$  cannot overlap’. Facts using this relation are not intended to be explicitly asserted in the KB. This is in designed to be used in Axioms presented in Section 3.5.2. This relation is defined in terms of CC relation between these attributes, as follows.

$$\text{D 3.6 } \text{CO}(a_1, a_2) \equiv_{def} \neg \text{CC}(a_1, a_2) \wedge \neg \text{CC}(a_2, a_1)$$

### 3.5.2 Derivation Rules

This section presents a number of rules which determine the way the system derives implicit *Star* facts. These derivation rules are specified in the form of axioms and, in conjunction with the axioms presented in Section 3.5.1, allow implicit facts representing spatio-temporal attributed regions to be derived from other facts explicitly asserted in the KB.

Derived facts can be related to the originating ones in space, time or in both. For short, in the cases where only spatial relations between *Stars* are relevant to derive new facts, their temporal aspects are ignored. That is, they are not mentioned in the description of the rules or in the examples given. Thus, in these cases, it should be assumed that all facts cited refer to the same instant of time (i.e., *Stars* are associated with the same timestamp).

Some facts inferred by the system are then explicitly asserted in the knowledge base at preprocessing time, so that these facts can be quickly evaluated at query time. A description of the preprocessing mechanism is given in Chapter 5, along with a discussion on the appropriate facts to be considered within this mechanism.

Some of the properties denoted by the association of attributes with spatial regions can be regarded as downwards- or upwards-inheritable. Downwards inheritance refers to the transfer of properties from wholes to their parts. For example, if  $x$  is (completely) made of mud, then its parts are also (completely) made of mud. On the other hand, upwards inheritance refers to the transfer of properties from parts to wholes. For instance, if a part of  $x$  touches the ground, then  $x$  touches the ground.

Hence, this model includes axioms which specify explicitly the cases where property inheritance is applicable amongst spatio-temporal attributed regions, so that implicit *Stars* can be inferred by inheriting the properties of the originating *Star(s)*. The downwards inheritance is described in derivation rule DR1, described below.

**Derivation Rule (DR1)** : if a spatial region has the coverage  $c$ , then every sub-region of this region also has the coverage  $c$ .

That is to say that, if  $\text{Star}(a, g, s)$  holds (where  $a$  is a coverage attribute) then any  $\text{Star}(a, g', s)$  (where  $g'$  is part of  $g$ ) will be evaluated as true. This derivation rule is specified in Axiom A3.14, as follows.

$$\text{A 3.14 } \text{Star}(a, g, s) \wedge \text{CAtt}(a) \rightarrow \forall g' [\text{P}(g', g) \rightarrow \text{Star}(a, g', s)]$$

It is important highlighting that DR1 cannot be applied to *Stars* representing geographic features, since it is assumed here that a feature cannot contain another feature. Moreover, it can be seen that Axiom A3.14 is specified using the predicate  $CAtt(a)$ , meaning that both homogeneous and heterogeneous regions are subject to downwards inheritance. Notice that this derivation rule takes into account the assumption that a sub-region of a region described by a heterogeneous coverage attribute can be described using the same attribute, even if such a sub-region presents a homogeneous coverage over a given period of time, as discussed in Section 3.5.1.

Figure 3.2 shows examples of *Stars* inferred by DR1. In the scenario of this figure, two spatio-temporal attributed regions with the same coverage are explicitly asserted in the KB. The spatial extension of these *Stars* are demarcated using solid outlines, whilst their coverages are illustrated using fills of the same shade of grey. Additionally, the figure shows two other *Stars* whose spatial extensions are demarcated by dotted outlines. Thus, it can be said that these two (last mentioned) *Stars* have been inferred by DR1, since each of them is located completely inside one of those original *Stars*. On the other hand, it is not possible to say that a *Star* whose extension corresponds to the region demarcated by the dashed line has been inferred by DR1, even though it can be seen in the picture that any one part of it is also part of the original stars. This occurs because a *Star* can only be inferred by DR1 if it is part of a particular *Star*.

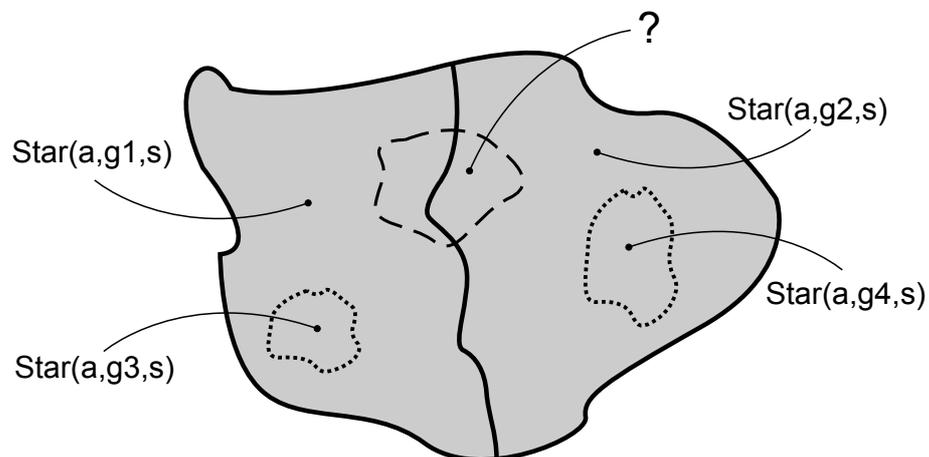


Figure 3.2: Two connected *Stars* are illustrated using solid outlines. Both *Stars* have the same coverage (represented by using fills of the same shade of grey). Moreover, two *Stars* inferred by DR1 are shown using dotted outlines. However, from DR1, it is not possible to infer a *Star* demarcated with a dashed outline, since this region is not completely inside a particular existing *Star*.

DR1 determines how downwards inheritance is established within the STAR model. The following derivation rule (DR2) describes how implicit *Star* facts can be derived by upwards inheritance. This is as follows.

**Derivation Rule (DR2)** : If at a certain time instant there exists a coverage  $c$  which can contain the coverages of two spatially connected regions  $r_1$  and  $r_2$ , then there exists a spatial region  $r_3$  whose coverage at that time instant is  $c$  and whose spatial extension corresponds to the spatial union of the extensions of  $r_1$  and  $r_2$  (subject to the condition that, for all  $c'$  that  $c$  must contain, there exists a subregion of  $r_3$  covered by  $c'$ ).

In other words, if  $\text{Star}(a_1, g_1, s)$  and  $\text{Star}(a_2, g_2, s)$  hold, where  $g_1$  and  $g_2$  are spatially connected (i.e.,  $C(g_1, g_2)$  holds); and there exists an attribute  $a_3$  where  $\text{CC}(a_3, a_1)$  and  $\text{CC}(a_3, a_2)$  hold; then the fact  $\text{Star}(a_3, g_3, s)$  can be inferred (where geometry  $g_3$  corresponds to the spatial union between  $g_1$  and  $g_2$ ) if for all attribute  $a$  that  $a_3$  must contain,  $\text{Star}(a, g, s) \wedge C(g_3, g)$  holds. This derivation rule is specified in Axioms A3.15 and A3.16, shown below.

This derivation is completed by performing two repeated steps. *First*, Axiom 3.15 derives a *Star* based on the coverage of two connected *Stars* which are part of the former. However, since a *Star* associated with attribute  $a_3$  will only exist if it contains at least one subregion covered by each attribute  $a$  such that  $\text{MC}(a_3, a_2)$ , the derived *Star* is associated with an indefinite coverage attribute  $a$  so that  $\text{CC}_{\text{cond}}(a_3, a)$  holds. The relation  $\text{CC}_{\text{cond}}$  is employed to mean that the CC relation between  $a_3$  and  $a$  is subject to a certain condition. That is, if  $\text{CC}_{\text{cond}}(a_3, a)$  holds but  $\text{CC}(a_3, a)$  does not hold, the system will not consider such a containment relationship to perform other inferences. *Second*, Axiom 3.16 checks whether the derived *Star* contains all required subregions (i.e., whose coverages are related to the coverage of the derived *Star* by MC). If so, Axiom 3.16 derives a *Star* that is associated with attribute  $a_3$  and has the same geometry of the *Star* generated in the previous step.

It can be observed that, if the relation MC did not exist, then DR2 could be performed by applying only Axiom 3.15 (where the head of the formulae should be replaced by  $\exists g_3[\text{Star}(a_3, g_3, s) \wedge g_3 = \text{union}(g_1, g_2)]$ ). Moreover, in Axiom 3.15, each occurrence of the CC relation in the body of the formula should be read as  $\text{CC}(x, y) \vee \text{CC}_{\text{conf}}(x, y)$  (this was omitted for readability). Similarly, in Axiom 3.6, the head of the formula (i.e.,  $\text{CC}_{\text{conf}}(a_1, a_2)$ ) should be read as  $\text{CC}(a_1, a_2) \vee \text{CC}_{\text{conf}}(a_1, a_2)$ . This was omitted for improved understanding of the Axiom when it was presented.

$$\begin{aligned}
\text{A 3.15 } & \exists a_1 a_2 [\text{Star}(a_1, g_1, s) \wedge \text{Star}(a_2, g_2, s) \wedge \\
& \text{CAtt}(a_1) \wedge \text{CAtt}(a_2) \wedge \text{CAtt}(a_3) \wedge \\
& \text{CC}(a_3, a_1) \wedge \text{CC}(a_3, a_2) \wedge \text{C}(g_1, g_2)] \\
& \rightarrow \exists ag [\text{CAtt}(a) \wedge \text{CC}_{\text{cond}}(a_3, a) \wedge \\
& \text{Star}(a, g, s) \wedge g = \text{union}(g_1, g_2)]
\end{aligned}$$

$$\begin{aligned}
\text{A 3.16 } & \text{CC}_{\text{cond}}(a_3, a) \wedge \text{Star}(a, g, s) \wedge \\
& \forall a' [\text{MC}(a_3, a') \rightarrow \exists g' [\text{Star}(a', g', s) \wedge \text{C}(g, g')]] \\
& \rightarrow \text{Star}(a_3, g, s)
\end{aligned}$$

In Axiom A3.15, the auxiliary function *union* is used. This function takes two geometries as an input and returns a geometry which corresponds to the spatial union between the other two. This axiom is specified in such a way that it can be applied to the cases where the originating *Stars* are associated with the same or with different coverage attributes (denoted by  $a_1$  and  $a_2$ ). This is possible because this axiom uses the general predicate for coverage attributes *CAtt*. In this axiom, attributes  $a_1$  and  $a_2$  may correspond to:

1. the same homogeneous coverage;
2. the same heterogeneous coverage;
3. different homogeneous coverages;
4. one a homogeneous and the other a heterogeneous coverage.

For the cases 1 and 2 above, where  $a_1$  and  $a_2$  correspond to the same attribute (either homogeneous or heterogeneous), the derivation of Axiom A3.15 can be performed even if no  $\text{CC}(a_1, a_2)$  or  $\text{MC}(a_1, a_2)$  facts are explicitly asserted relating them. This is possible because Axioms A3.3 and A3.5 ensure that  $\text{CC}(a, a)$  always holds (i.e., where both attributes are the same), and therefore  $a_1 = a_2 = a_3$  will hold.

Regarding the cases 3 and 4 above, where  $a_1$  and  $a_2$  correspond to different attributes, it should be noted that Axiom A3.4 ensures that any  $\text{CC}(a, a)$  or  $\text{MC}(a, a)$  fact asserted between different coverage attributes will necessarily relate a homogeneous to a heterogeneous coverage attribute (where the former is the part and the latter is the whole). Thus, if two connected *Stars* fall in one of these cases, so that they are combined to derive a new *Star* by Axiom A3.15, this means that variable  $a_3$  in this axiom corresponds (necessarily) to a heterogeneous coverage attribute. For variables  $a_1$  or  $a_2$  one of the following options will hold:

- case 3:  $a_1$  and  $a_2$  are both homogeneous coverages and the heterogeneous coverage and  $a_3$  can contain both of them;
- case 4: either  $a_1$  or  $a_2$  is a homogeneous coverage (and the heterogeneous coverage  $a_3$  can contain it); and the remaining variable (i.e., either  $a_1$  or  $a_2$ ) is the same heterogeneous coverage assigned to  $a_3$  (therefore  $a_3$  can contain it too).

Figure 3.3 illustrates 3 different situations in which an implicit *Star* can be derived as determined by derivation rule DR2. In each of these illustrations, thicker solid outlines are used to identify spatial boundaries of the 3 *Stars* involved in the logical inference specified by DR2. Therefore 2 of them are regarded as already present in the KB (body of formula in Axiom A3.15) and the other one corresponds to the inferred fact (head of formula in Axiom A3.15).

On the other hand, thinner solid outlines determine spatial boundaries of other *Stars* which already exist in the KB but are not involved in the logical inference demonstrated in the examples. These *Stars* could have been either explicitly asserted or resulting from any other logical inference. It is assumed that there are 3 distinct homogeneous coverage attributes which can be associated with *Stars*. These attributes are illustrated by using different shades of grey. Moreover, just one heterogeneous coverage attribute is considered in these examples, which can contain any of those homogeneous ones.

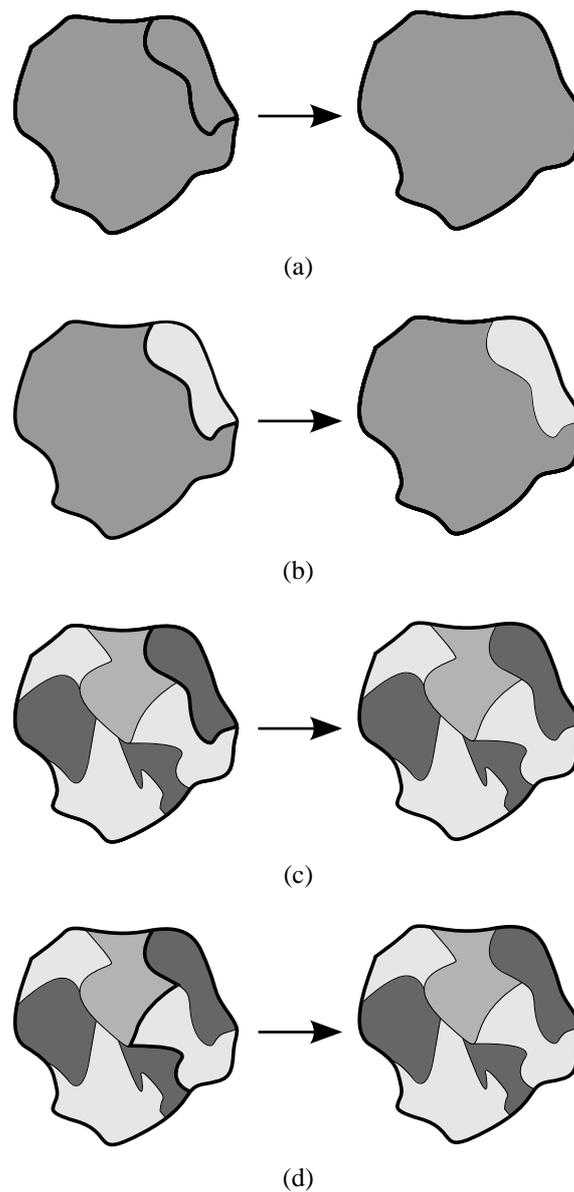


Figure 3.3: Examples of logical inferences which follow DR2. In 3.3a, the combination of two regions with the same homogeneous coverage results in a new homogeneous region. In 3.3b, two regions with different homogeneous coverages are combined to derive a new region covered heterogeneously. In 3.3c, a homogeneous and a heterogeneous region are combined to originate a new heterogeneous region. In 3.3d, two heterogeneous regions coalesce into a new heterogeneous region.

In Figure 3.3a, two homogeneous *Stars* associated with the same homogeneous coverage attribute are combined to derive a new fact representing a *Star* associated with the same homogeneous coverage attribute as the originating facts. In 3.3b, two *Stars* associated with different homogeneous coverages are combined to produce a new *Star*

associated with a heterogeneous coverage attribute. In Figure 3.3c, a heterogeneous and a homogeneous *Star* are combined to derive a new fact representing a heterogeneous *Star*, whose coverage is the same as the originating heterogeneous *Star*. Finally, Figure 3.3d shows the combination of two heterogeneous *Stars* to derive another one with the same coverage.

In the example of Figure 3.2, it has been discussed that the region demarcated using a dashed line cannot be derived (though DR1) from the two homogeneous *Stars* explicitly asserted in the KB. However, this inference is possible by applying DR2 followed by DR1, as illustrated in Figure 3.4. In this figure, the fact  $\text{Star}(a, g_3, s)$  is derived (though DR2) from two other facts  $\text{Star}(a, g_1, s)$  and  $\text{Star}(a, g_2, s)$ , given that  $C(g_1, g_2)$  holds. Then, from this derived fact, it was possible to derive (though DR1) the fact  $\text{Star}(a, g_4, s)$ , where  $g_4$  is the geometry illustrated by the dashed outline (assuming that  $P(g_4, g_3)$  holds).

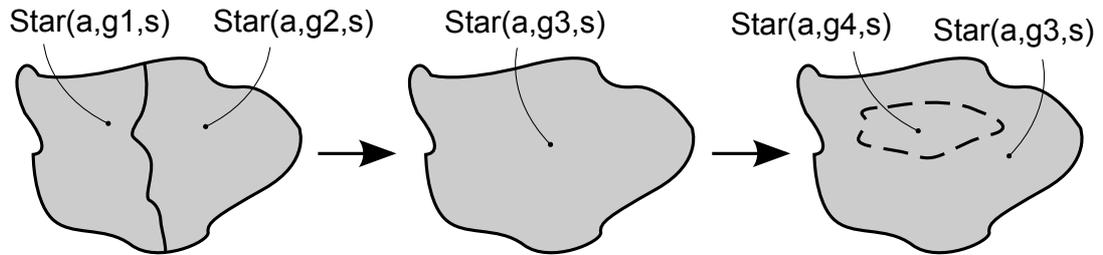


Figure 3.4: Example of a logical inference where a *Star* fact is derived by applying DR2 followed by DR1.

The derivation rules presented so far (DR1 and DR2) are not concerned with the temporal relations between premise facts and derived facts, that is, all facts are said to hold at the same instant of time. Differently, the following rule (DR3) is proposed to determine how facts which hold at a given time instant can be used to derive facts which hold at other time instants. This rule is based on the *common sense law of inertia* (everything remains the same until explicitly changed) and on the *closed world assumption* (what is not currently known to be true is false). This is as follows.

**Derivation Rule (DR3)** : if it is true that a spatial region  $r$  is covered by  $c$  at a given time instant  $t$ , then it is true that this region is covered by  $c$  at any time instant  $t'$  after  $t$  if no fact asserting that  $r$  does not have coverage  $c$  in the meantime (between  $t$  and  $t'$ ) can be derived.

That is to say that, if  $\text{Star}(a, g, s)$  holds, and no fact of the form  $\text{Star}(a', g', s')$  holds

(where  $s'$  is between  $s$  and  $s''$ ,  $g'$  is part of  $g$ , and  $a$  cannot overlap  $a'$ ), then  $\text{Star}(a, g, s'')$  also holds. This is specified in axiom A3.17 and definition D3.7, as follows.

$$\text{A 3.17} \quad \exists s[\text{Star}(a, g, s) \wedge \neg \text{Clipped}(a, g, s, s')] \rightarrow \text{Star}(a, g, s')$$

$$\text{D 3.7} \quad \text{Clipped}(a, g, s_1, s_2) \equiv_{def} \exists a' g' s' [\text{Star}(a, g, s_1) \wedge \text{Star}(a', g', s') \wedge (s_1 \prec s' \prec s_2) \wedge \text{CO}(a, a') \wedge \text{P}(g', g)]$$

This derivation rule (DR3) resembles the method proposed in the Event Calculus [54, 77], to solve the frame problem<sup>5</sup> (which in turn is similar to the successor state axioms of the Situation Calculus [71]): a fluent is true at a certain time instant  $t$  if it has been made true in the past and has not been made false in the meantime. Otherwise, the fluent is false.

An example of logical inferences drawn according to DR3 is exhibited in Figure 3.5. *Stars* are illustrated in this figure by using solid outlines and different coverage attributes are denoted by distinct shades of grey fills. Dashed outlines are shown just to improve the visualisation, and therefore do not represent *Star* facts. In this example, two *Stars* are asserted explicitly (whose predicate is described in the figure using A- prefix), and coverage attributes  $a$  and  $a'$  cannot overlap. The first fact (A-Star( $a, g, s$ )) asserts that the region denoted by geometry  $g$  has the coverage denoted by attribute  $a$  at time instant denoted by timestamp  $s$ . Thus, according to DR3, *Star* facts associating geometry  $g$  with attribute  $a$  also hold at all time instants after that and until the instant denoted by timestamp  $s'$ , as the other explicit fact shown in the figure associates a different attribute  $a'$  with the same geometry  $g$  at that time instant (A-Star( $a', g, s'$ )).

<sup>5</sup>The frame problem is introduced in Section 2.4.

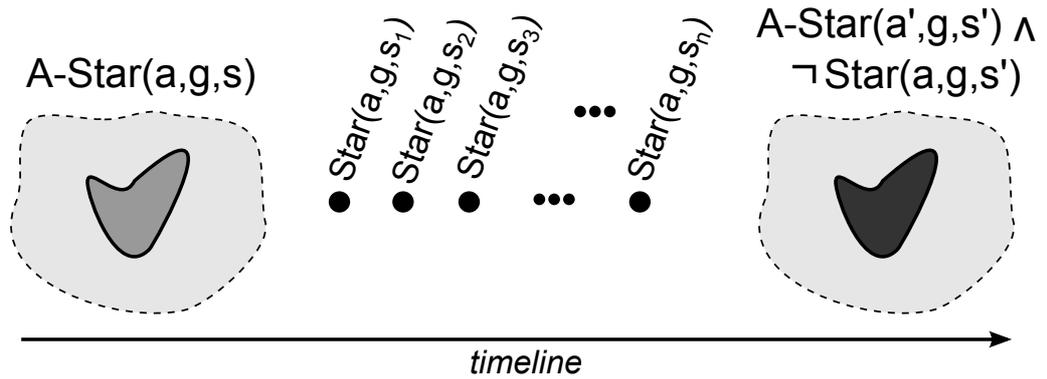


Figure 3.5: Example of logical inferences drawn according to DR3. A *Star* fact explicitly associates geometry  $g$ , attribute  $a$  and timestamp  $s$ . Thus any proposition using the predicate *Star* to associate  $g$ ,  $a$ , and any timestamp between  $s$  and  $s'$  will be evaluated as true.

A more elaborated example is shown in Figure 3.6. In this figure, lines and fills have the same meaning as in Figure 3.5. This example is similar to the example exhibited in Figure 3.5, however, in this last example, the second *Star* fact explicitly asserted associates timestamp  $s'$  and attribute  $a'$  with a geometry which is different from  $g$  (i.e.,  $A\text{-Star}(a',g',s')$ ). Since  $g'$  partially overlaps  $g$ , the fact  $A\text{-Star}(a,g,s')$  is false, meaning that the region denoted by  $g$  does not have the coverage denoted by  $a$  at time instant denoted by  $s'$ , even though it can be said this region has this coverage at any time instant between those denoted by  $s$  and  $s'$  (DR3). Nonetheless, the example shows that  $A\text{-Star}(a,g'',s')$  is still regarded as true, where  $g''$  represents the sub-region of the region denoted by  $g$  which does not overlap  $g'$ . Finally, the figure shows that the regions denoted by  $g$  and  $g''$  have the same coverage at the instants denoted by  $s'$  and  $s''$  (and at all instants between them), since there is no fact in the meantime which causes this to be false.

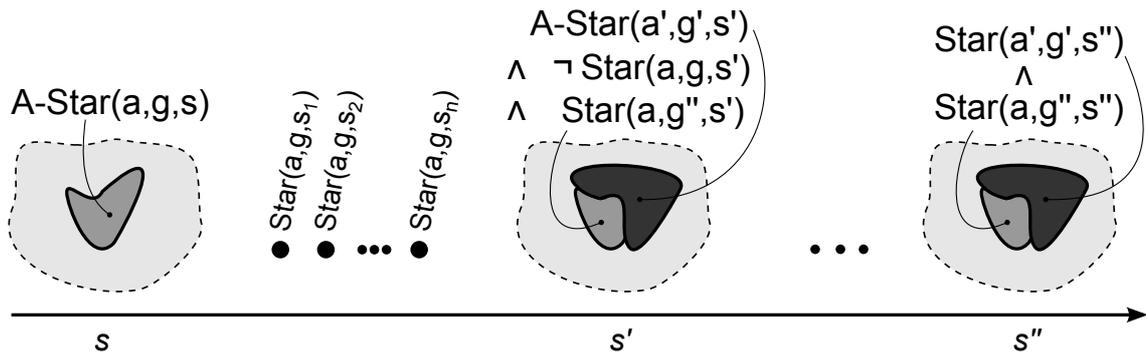


Figure 3.6: Example of logical inferences drawn according to DR3. *Stars* that associate attribute  $a$  and geometry  $g$  with any timestamp between  $s$  and  $s'$  are true. The geometry  $g''$ , which denotes a region that does not overlap  $g'$  can also be associated with  $a$  and  $s'$ . Finally, the same *Stars* evaluated as true for  $s'$  are also true for  $s''$ .

Figure 3.7 illustrates in more detail the inferences drawn in the example of Figure 3.6. On the right of Figure 3.7 can be seen the final scenario described in the example of Figure 3.6, where 2 regions with different coverages are spatially connected. On the top of Figure 3.7, it is shown that a *Star* associated with timestamp  $s$  is first derived through DR1 from another *Star* which contains it, and then it is used to infer other facts where the same geometry and attribute are associated with other timestamps representing time instants after the instant denoted by  $s$  (DR3). On the other hand, at the bottom of the figure, it can be seen that the *Star* derived by DR1 causes the fact  $\text{Star}(a, g''', s')$  not to be inferred by DR3 (and be regarded as false, since  $\text{CO}(a, a')$  holds). Other examples that illustrate inferences involving multiple derivation rules are described later in this Section.

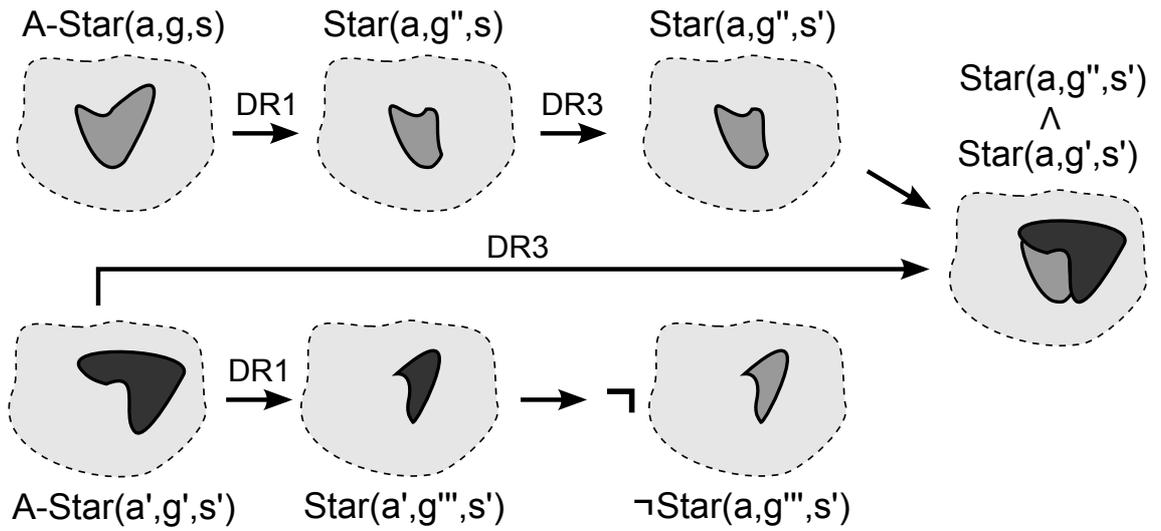


Figure 3.7: Inferences drawn in the example of Figure 3.6. Labels on arrows indicate the rule followed to draw certain inferences.

Rules DR1-3 are applied to derive *Stars* where both known and derived facts represent spatial regions. On the other hand, the following rules are applied to derive *Stars* representing geographic features from others representing spatial regions, and vice-versa.

**Derivation Rule (DR4)** : if there exists a spatial region  $r$  covered by  $c$  at time instant  $t$ ; if there exists no other region which  $r$  is part of this and whose coverage can be composed by  $c$ ; and if there exists a type of feature  $u$  which can be described in terms of regions covered by  $c$ ; then there exists a geographic feature of type  $u$  whose extension at time instant  $t$  corresponds to the extension of  $r$  (i.e., the geographic feature is regarded as the maximal well-connected region whose coverage is denoted by  $c$ ).

That is to say that, if there exists a *Star* associated with a coverage attribute  $a$ , geometry  $g$  and timestamp  $s$  (i.e.,  $\text{Star}(a, g, s)$  holds); there exists *no* *Star* associated with timestamp  $s$ , attribute  $a'$ , and geometry  $g'$ , where  $a'$  can contain  $a$  and  $g$  is proper part of  $g'$  (i.e.,  $\text{CC}(a', a)$  and  $\text{PP}(g, g')$  hold); and there exists a feature attribute  $a_f$  which can contain  $a$  (i.e.,  $\text{CC}(a_f, a)$  holds); then  $g$  represents the spatial extension of a geographic feature denoted by attribute  $a_f$  at time instant denoted by timestamp  $s$  (i.e.,  $\text{Star}(a_f, g, s)$  holds). This is specified in Axiom A3.18 (right-to-left implication), as follows.

$$\begin{aligned}
 \text{A 3.18 } \text{Star}(a, g, s) \wedge \text{FAtt}(a) &\leftrightarrow \exists a' [\text{CAtt}(a') \wedge \text{CC}(a, a') \wedge \text{Star}(a', g, s)] \wedge \\
 &\quad \neg \exists g' a' [\text{Star}(a', g', s) \wedge \text{CAtt}(a') \wedge \\
 &\quad \quad \text{CC}(a, a') \wedge \text{PP}(g, g')]
 \end{aligned}$$

Different examples of the application of DR4 are exhibited in Figure 3.8 (where different lines and fills are used as described for Figure 3.5). In the example of Figure 3.8a, ‘forested’ is regarded as a *homogeneous* coverage. In this example, an implicit fact concerning a homogeneous region is inferred by applying DR2 (two times), which in turn is used to derive a *simple* geographic feature by applying DR4. In the example of Figure 3.8b, different interpretations are given to ‘forested’ and ‘forest’. The former is now regarded as a *heterogeneous* coverage type which can be composed by homogeneous coverages of type ‘veg-a’, ‘veg-b’, and ‘veg-c’ (denoting different types of vegetation), whereas the latter denotes a *compound* type of geographic feature. However, the same derivation rules are applied to derive the compound feature. That is, first DR2 is applied (two times) to derive the ‘forested’ region, then DR4 is applied to derive the *compound* geographic feature ‘forest’.

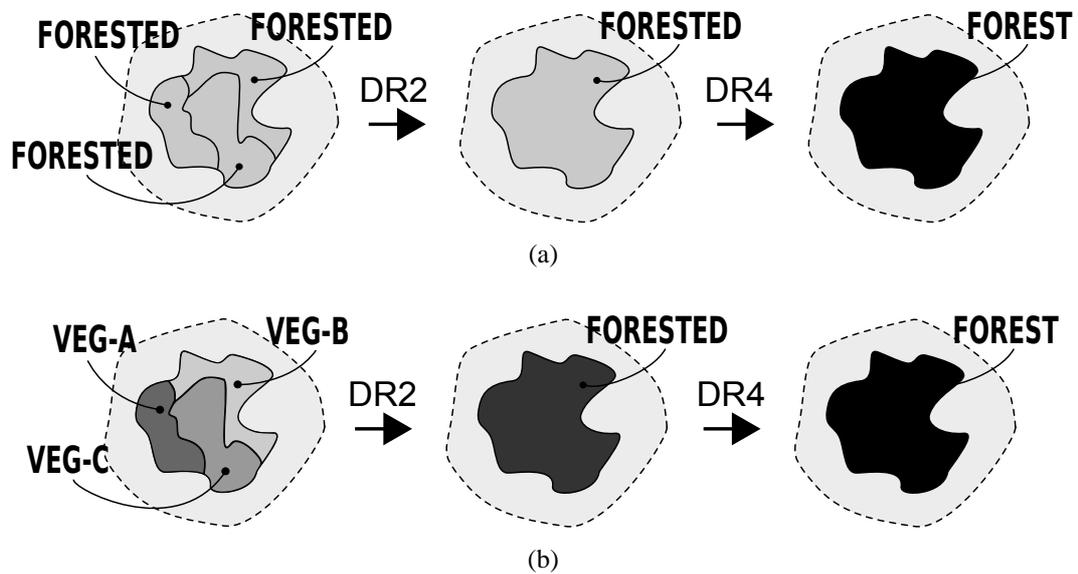


Figure 3.8: Examples of logical inferences where geographic features are derived from spatial regions. In (a), a simple feature ‘forest’ is derived from homogeneous regions (‘forested’) by applying DR2 and DR4. In (b), ‘forested’ is a heterogeneous region derived (through DR2) from homogeneous regions representing different types of vegetations (‘veg-a’, ‘veg-b’, and ‘veg-c’). This heterogeneous region is then used to derive a compound feature ‘forest’ through DR4.

The following derivation rule can be considered as the reverse of DR4. This is applied to derive spatial regions from geographic features.

**Derivation Rule (DR5)** : if there exists a geographic feature at a certain time instant, then there exists a spatial region  $r$  whose spatial extension corresponds to the extension of this feature and whose coverage  $c$  is the one which determines the type of this feature. Moreover, if such a geographic feature exists, there exists no other region which  $r$  is part of and whose coverage can contain  $c$ .

In other words, if  $\text{Star}(a_f, g, s)$  holds (where  $a_f$  is a feature attribute), then  $\text{Star}(a_c, g, s)$  also holds (where  $a_c$  is a coverage attribute related to  $a_f$  by  $\text{CX}(a_f, a_c)$ ) and there is no  $\text{Star}(a'_c, g', s)$  (where  $a'_c$  is a coverage attribute and  $\text{CC}(a'_c, a_c)$  and  $\text{PP}(g, g')$ ). The biconditional used in Axiom A3.18 allows inferences to be made as described in this rule (left-to-right implication).

This rule (DR5) is useful as facts representing geographic features can also be asserted explicitly in the KB. Thus implicit spatial regions can be derived from these geographic features by applying DR5. However, notice that such a derivation mechanism is specified in the first part of the rule description given above. On the other hand, the second part described above acts as an integrity constraint to prevent inconsistent *Stars* to be explicitly asserted, which might contradict the assumptions that a geographic feature is denoted by the maximal extension of a particular coverage and that it cannot contain or be contained by other features.

In the example illustrated in Figure 3.9, a *Star* fact concerning a geographic feature is used to derive *Star* facts about spatial regions, by applying DR5 and then DR1. If the geographic feature of this example is regarded as a *simple* feature, then this example illustrates the reverse of the process shown in Figure 3.8a. That is, different *homogeneous* regions ‘forested’ are derived from the feature. On the other hand, if such a geographic feature is regarded as a *compound* feature, it would not be possible to perform inferences which corresponds to the reverse of those shown in Figure 3.8b, since the extension of each different *homogeneous* region (covered by ‘veg-a’, ‘veg-b’, or ‘veg-c’) could not be determined. In this case, the result obtained by applying DR5 and DR1 to the compound feature would be exactly as shown in Figure 3.9, that is, different *heterogeneous* regions (‘forested’) could be derived from the feature.

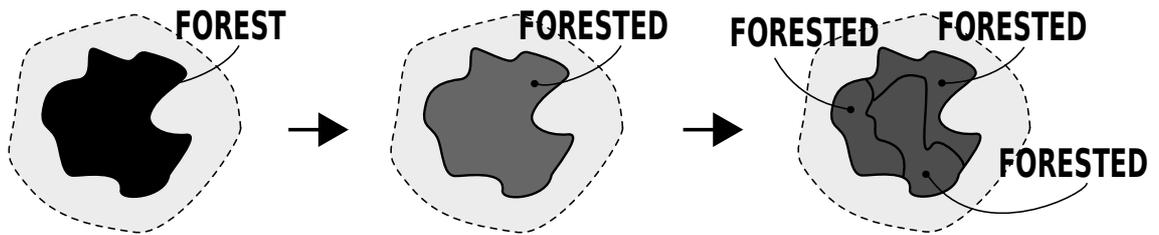


Figure 3.9: Example of logical inferences where spatial regions and their coverages are derived from a geographic feature (by applying DR5 then DR1).

The derivation rules described above enable the system to integrate spatio-temporal data based on different temporal granularities (e.g. days, months). By performing inferences determined by DR3, facts which hold at a certain time instant are used to derive implicit facts which hold at successive time instants. Then these implicit facts can be combined to originate other facts. An example is given in Figure 3.10 to illustrate a way these derivation rules can be combined to integrate data describing different portions of space at two distinct time instants (denoted by timestamps  $s_1$  and  $s_2$ ). In this figure, lines and fills have the same denotation as described for Figure 3.3, and dotted outlines are given just to provide improved visualisation of the illustration.

In this example, the KB consists of 7 explicit *Stars* representing regions covered homogeneously by 3 different types of coverages. In addition, it is assumed that there are CC facts asserting that a certain heterogeneous coverage can contain these homogeneous coverages. These explicit *Star* facts are illustrated in boxes 1 and 2, on the top of Figure 3.10 (containing 5 *Stars* associated with timestamp  $s_1$  and 2 *Stars* associated with timestamp  $s_2$ , respectively). The arrow from box 1 to box 3 illustrates an inference drawn (according to DR2) to derive two new facts representing heterogeneous regions from explicit facts associated with  $s_1$ . These derived regions are then used to infer (through DR3) *Stars* facts associated with  $s_2$  (see boxes 2, 3 and 4 in Figure 3.10). Finally, these inferred facts are then combined with facts explicitly asserted for  $s_2$  to originate new facts representing heterogeneous regions, by applying DR2 successive times (see boxes 4, 5 and 6 in Figure 3.10). Moreover, notice that the fact which represents the last scenario of this example, could still be used, for instance, to infer (via DR3) other facts which hold in the future or to infer the extension of a compound geographic feature.

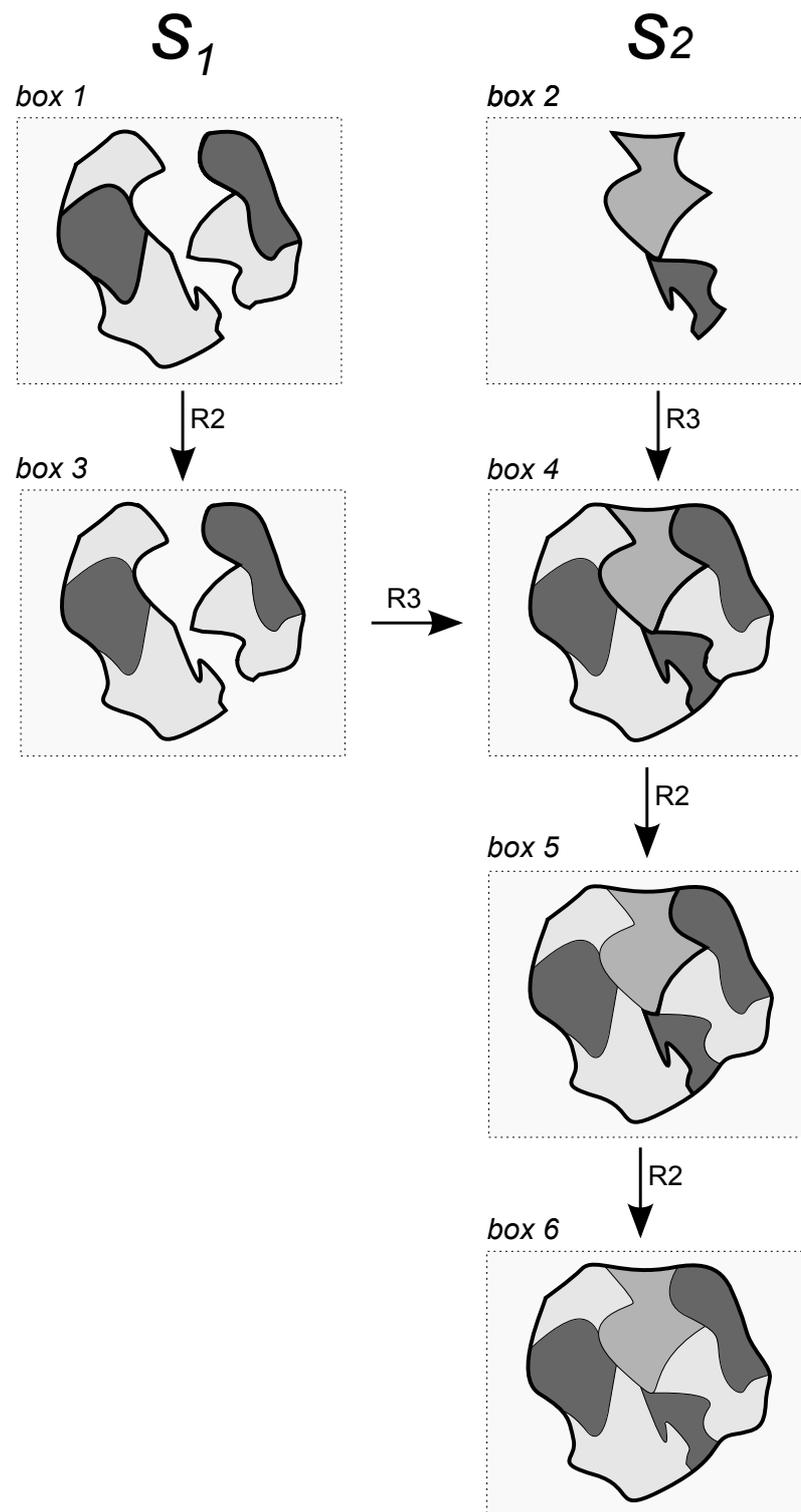


Figure 3.10: Combining derivation rules to integrate data describing different portions of space at two distinct time instants. First, *Star* facts associated with  $s_1$  are used to derive (through DR2) new facts associated with the same timestamp. These new facts are then used to infer *Stars* associated with  $s_2$  (via DR3). Finally, these inferred facts are combined with other explicit facts associated with the same timestamp to derive other *Stars* (through DR2).

### 3.5.3 Aggregated STARs

It has been discussed that space is conceived in the STAR model in accordance with the RCC theory of space, with the additional constraint that spatial regions have to be internally connected, that is, they may not consist of multiple disconnected pieces. Nonetheless, as discussed in Section 2.10, there are a number of examples where representing geographic features as a set of disconnected regions is critical for representing and reasoning about certain geographic phenomena.

Therefore the STAR model is extended to add the capability of deriving a fact representing a spatial region from a set of facts representing disconnected spatial regions. The formation of aggregated spatial regions is illustrated in Figure 3.11, where two spatial regions are derived from distinct sets of disconnected spatial regions of the same coverage (in this figure, lines and fills have the same denotation as described for Figure 3.5).

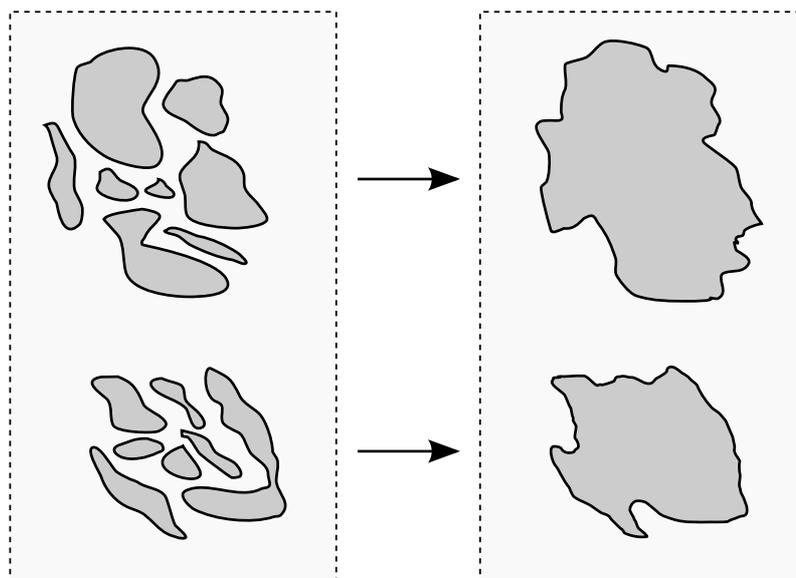


Figure 3.11: Example of spatial regions derived by aggregating disconnected spatial regions.

From the example of Figure 3.11, it can be seen that everything between those disconnected regions are completely abstracted in the representation of derived regions. This abstraction is similar to the traditional method of representing maps at different scales, where details are removed from the map as the scale decreases. This method is usually called ‘generalisation’ – though that term is not exactly appropriate. However, the motivation for the formation of such aggregates is not just a matter of data visualisation. In fact,

the criteria for determining aggregates relate to the sort of geographic phenomena that is intended to be analysed based on changes affecting them. Consequently, these criteria depend on factors such as the type of coverage associated with spatial regions and the distance tolerated between them. That is, by changing these factors, distinct aggregates can be obtained from a particular spatial dataset (i.e., without modifying the map scale).

Determining the appropriate *distance* between spatial regions for the formation of aggregations depends on many variables, and therefore this value could be specified by an expert, such as an ecologist. Clearly, this is a problem affected by *sorites vagueness*, where different interpretations might arise regarding the proper distance between the elements of an aggregate. Therefore the method of deriving aggregates proposed within the STAR model is based on *standpoint semantics*, where such a distance is regarded as a standpoint parameter. This parameter allows precisification values to be specified so that *Stars* derived by different aggregation criteria can co-exist in the KB. In this approach, the predicate  $\text{Star}(a, g, s)$  takes the following form:

$\text{Star}[d](a, g, s)$ , where  $d$  is the distance standpoint parameter.

*Star* facts of the form described above are called here ‘aggregated *Stars*’, whereas the distance parameter is also referred to as the ‘aggregation factor’. Additionally, facts where the standpoint parameter is not applied are considered equivalent to those where the distance parameter is zero (i.e.,  $\text{Star}(a, g, s) \equiv \text{Star}[0](a, g, s)$ ).

Aggregated *Stars* are derived as specified in DR6, which can be understood as an extension of DR2. This is as follows.

**Derivation Rule (DR6)** : if the distance between two spatial regions  $r_1$  and  $r_2$  is less than  $d$  at a certain time instant, and there exists a coverage  $c$  which can contain the coverages of both  $r_1$  and  $r_2$ , then there exists a spatial region  $r_3$  at that time instant, which has the coverage  $c$  and whose spatial extension corresponds to a concave hull comprising the extensions of the originating regions ( $r_1$  and  $r_2$ ).

The vague connectivity between regions is defined as a modification of the RCC relation *Externally Connected*  $\text{EC}(x, y)$ , as follows.

$$\text{D 3.8} \quad \text{EC}[d](x, y) \equiv_{def} [(\text{distance}(x, y) \leq d) \wedge \neg \text{O}(x, y)]$$

In Definition D3.8, the auxiliary function  $\text{distance}(x, y)$  has been employed. This

function returns a non-negative number representing the 2-dimensional Cartesian minimum distance between two regions in projected units. Any non-negative real number can be assigned to the parameter  $d$ , in order to be compared with the value calculated by the distance function. In addition, when the relation  $EC[d](x,y)$  holds for two spatial regions  $x$  and  $y$ , it is said that these regions are ‘vaguely connected’.

Once this new relation has been introduced, DR6 can be described in a different manner: if  $Star(a_1, g_1, s)$  and  $Star(a_2, g_2, s)$  hold, where  $g_1$  and  $g_2$  are vaguely connected by the factor  $d$  (i.e.,  $EC[d](g_1, g_2)$  holds); and there exists an attribute  $a_3$  where  $CC(a_3, a_1)$  and  $CC(a_3, a_2)$  hold; then the fact  $Star[d](a_3, g_3, s)$  is derived, where geometry  $g_3$  corresponds to a concave hull comprising  $g_1$  and  $g_2$  (subject to the condition that, for all  $d'$  that  $a_3$  must contain, there exists a *Star* associated with attribute by  $d'$  and whose geometry is connected to  $g_3$ ).

This derivation rule is specified in Axiom A3.19 (which is a modified version of Axiom A3.15), as follows.

$$\begin{aligned}
 \text{A 3.19 } & \exists a_1 a_2 [\text{Star}(a_1, g_1, s) \wedge \text{Star}(a_2, g_2, s) \wedge \\
 & \quad \text{CAtt}(a_1) \wedge \text{CAtt}(a_2) \wedge \text{CAtt}(a_3) \wedge \\
 & \quad \text{CC}(a_3, a_1) \wedge \text{CC}(a_3, a_2) \wedge \text{EC}[d](g_1, g_2)] \\
 & \rightarrow \exists a g [\text{CAtt}(a) \wedge \text{CC}_{\text{cond}}(a_3, a) \wedge \\
 & \quad \text{Star}(a, g, s) \wedge g = \text{concave-hull}(g_1, g_2)]
 \end{aligned}$$

$$\begin{aligned}
 \text{A 3.20 } & \text{CC}_{\text{cond}}(a_3, a) \wedge \text{Star}(a, g, s) \wedge \\
 & \quad \forall a' [\text{MC}(a_3, a') \rightarrow \exists g' [\text{Star}(a', g', s) \wedge \text{EC}[d](g, g')]] \\
 & \rightarrow \text{Star}(a_3, g, s)
 \end{aligned}$$

*Star* facts derived through DR2 and DR6 are semantically equivalent when the parameter  $d$  is zero, assuming the function concave-hull calculates the same result as function union where input geometries are spatially connected (which of course is an idealisation). Hence, when the formation of aggregated regions are intended, DR6 should be used in replacement of DR2. Then derived facts can be applied to any other derivation rule given above to produce other implicit facts.

For brevity, the axioms that specify those derivation rules will not be re-written using the the notation  $Star[d](a, g, s)$ . However, it must be highlighted that this is the actual notation used in those axioms. That is, wherever a term of the form  $Star(a, g, s)$  appear in those axioms, this should be read as  $Star[d](a, g, s)$ . Nevertheless, this does not include

facts of the form  $A\text{-Star}(a, g, s)$ , since aggregated *Stars* are always resulting from an inference and never explicitly asserted in the KB. Moreover, according to the way standpoint semantics is used within axioms, it should be noticed that an aggregated *Star* derived from other aggregated *Stars* is always associated with the same standpoint parameter of the originating facts (except, of course, for facts derived through aggregation). However, a *Star* whose aggregation factor is  $d$  (and whose attribute corresponds to a region coverage), can also be represented by a factor greater than  $d$ . This is specified in Axiom A3.21, as follows.

$$\text{A 3.21} \quad \text{Star}[d](a, g, s) \wedge \text{Catt}(a) \wedge d < d' \rightarrow \text{Star}[d'](a, g, s)$$

Evidently, the inclusion of DR6 may lead the system to derive overlapping *Stars* denoting spatial extensions of geographic features of the same type (associated with the same feature attribute). However, as they are associated with distinct aggregation factors, these *Stars* can be independently retrieved from the KB. Hence, as will be discussed in detail in the next section, the way a geographic feature evolves is modelled based on a set of *Stars* associated with a particular aggregation factor, and therefore distinct modelling of a feature evolution express different standpoints.

### 3.6 Modelling Geographic Features

Geographic features will be regarded as a particular kind of *endurant entity*. Although they differ in some way from artefacts or organisms, they share many properties with other kinds of *endurants*, for instance:

- Geographic features are regarded here as *discrete individuals*, and can be referred to by a proper noun (e.g., Amazon Forest, Atlantic Ocean, Antarctic Desert), a count noun (e.g., a glacier or even an oil slick on the sea), or by more complex sentences, such as ‘mountains over 1,500m in height’. That is, any maximal well-connected spatial region with explicit and well-defined spatial extension, which can be individuated based on a certain aspatial and atemporal characteristic (i.e., a region coverage) can be regarded as a feature.
- Geographic features have *well-defined spatial extensions*<sup>6</sup>.

<sup>6</sup>Actually, determining the spatial extension of a geographic feature is a matter of vagueness, as discussed in Section 2.8.3. However, it is assumed here that every feature in the model can be associated with a precise spatial extension. Methods of handling vagueness could be applied to assign precise spatial ex-

- Geographic features are able to *undergo change* (e.g., changes in shape or area); and they can change some of their parts while keeping their *identity* (e.g., a forest can be partially deforested while being still the same forest).
- Geographic features can have spatial parts, but do not have temporal parts. This means that one could not refer to something like ‘the earliest part of a feature’, or ‘the last 10 minutes of a feature’.

Although a geographic feature is regarded as having no temporal parts, it is said to have a *life*, which in turn is conceived as having temporal parts. That is, given a geographic feature  $f$ , the temporal parts of its life can be referred to by using expressions such as ‘the first 10 minutes of  $f$ ’s life’. The period of time in which a geographic feature is said to live corresponds to the *maximal interval* throughout which the feature maintains its identity. This is regarded as the interval on which the feature exists (i.e., it is ‘alive’). The *identity criteria* of a geographic feature is defined in terms of connectivity between its spatial extensions over a time interval, as follows.

$$\begin{aligned}
 \text{D 3.9 } \text{Lives}(f, i) &\equiv_{\text{def}} \\
 &\forall tr[(\text{ext}(f, t) = r) \rightarrow \text{In}(t, i)] \wedge \\
 &\forall tr[(b(i) \leq t < e(i)) \wedge (\text{ext}(f, t) = r) \\
 &\quad \leftrightarrow \exists t' r' [(t < t' \leq e(i)) \wedge (\text{ext}(f, t') = r') \\
 &\quad \quad \forall t'' r'' [(t \leq t'' \leq t') \wedge (\text{ext}(f, t'') = r'') \wedge \\
 &\quad \quad \rightarrow C(r, r'') \wedge C(r'', r')]]]
 \end{aligned}$$

According to Definition D3.9, a spatial region  $r$  corresponds to the extension of a feature  $f$  at time instant  $t$  only in case this time instant is in the interval  $i$ , which in turn corresponds to the feature’s life. Moreover, this definition assures that, for all time instants  $t$ , so that  $b(i) \leq t < e(i)$ ,  $f$  occupies a spatial region  $r$  at  $t$  if and only if there exists a spatial region  $r'$  occupied by  $f$  at a time instant  $t'$  (where  $t < t' \leq e(i)$ ); and all spatial regions  $r''$  occupied by  $f$  from  $t'$  to  $t''$  are connected to both  $r$  and  $r'$ .

From Definition D3.9, it can be realised that if the spatial extension of a geographic feature is known at time instants  $t$  and  $t'$  (i.e.  $\text{ext}(f, t) = r$  and  $\text{ext}(f, t') = r'$ ) and nothing is known about the extension of that feature in the meantime, then the spatial regions corresponding to those extensions (i.e.,  $r$  and  $r'$ ) must be spatially connected. This is true because DR3 specifies that if nothing is known about the coverage of  $r$  in the meantime,

---

tensions to vague features. An example of such an approach to dealing with this issue based on standpoint semantics [10] is presented by Bennett et al. [12, 13]

the region is said to maintain its coverage over that time interval. It can be perceived that this assumption reduces the range of problems the STAR model can be applied for practical purposes. That is, this assumption is particularly applicable to geographic phenomena which can be described in terms of changes affecting the spatial extension of geographic features (e.g., expansion of a forest), rather than phenomena described in terms movements performed by an object (e.g., movement of a car).

Of course the applicability of this model depends on the dataset used, which may vary in many aspects, such as in terms of their temporal granularity. In fact, the main goal of using this model is to extract knowledge from the dataset provided, and therefore this model should only be applied to problems where the aforementioned assumption is appropriate. For example, some problems which involves modelling the trajectory of a certain person may be represented by a dataset which is not appropriate for this model. To illustrate, suppose a person (of type scientist) occupies a certain seat on 01/01/2013 and *another* person (also scientist) does two weeks later, on 15/01/2013, and these are the only facts known about anyone occupying that seat in this period. Then these two facts give a pair<sup>7</sup>:

$$\langle \text{Star}(\textit{scientist}, \textit{the seat}, 01/01/2013); \text{Star}(\textit{scientist}, \textit{the seat}, 15/01/2013) \rangle$$

In this case, the system would infer that both people (occupying the seat on 01/01/2013 and on 15/01/2013) are the same individual, which is not true for the problem that the dataset is intended to represent. In contrast, if it is known that that a particular region on the earth surface is occupied by a forest on 01/01/2013 and then on 15/01/2013, they will probably be the same forest.

Conversely, suppose suppose it is known of a certain person that they are in Leeds (UK) on 01/01/2013 and in Scarborough (UK)<sup>8</sup> on 15/01/2013, and nothing is known about their whereabouts between these two times. This is encoded in the KB by the pair<sup>9</sup>:

$$\langle \text{Star}(\textit{scientist}, \textit{Leeds-UK}, 01/01/2013); \\ \text{Star}(\textit{scientist}, \textit{Scarborough-UK}, 15/01/2013) \rangle$$

In this case, the system would infer that both people (being in Leeds on 01/01/2013 and in Scarborough on 15/01/2013) are different individuals (even if they are of the same

<sup>7</sup>In fact, *Stars* should contain a geometry rather than a place description (i.e. ‘the seat’), but an abuse of the language was used here to facilitate the description of the example.

<sup>8</sup>These cities are approximately 67 miles far from each other.

<sup>9</sup>Place names were used here instead of geometries to facilitate the description of the example.

type – a scientist, since Leeds and Scarborough are not connected to each other). Clearly, this is not true for the problem at hand. Contrastingly, if it is known that a particular region on the earth’s surface (e.g., near Leeds) is occupied by a lake on 01/01/2013 and another region (e.g., near Scarborough) is occupied by a lake on 15/01/2013, these features will surely correspond to different lakes. In the kind of phenomena which are the target of this work, features’ movements are more frequently modelled as a consequence of successive extensions and contractions affecting the boundary of a geographic feature, but not as an action intentionally performed by the feature. However, this is not a restriction imposed by the model, since movements can still be modelled if the dataset contains the regions occupied by an object throughout the trajectory so that the requirement of continuity imposed by the identity criteria (specified in 3.9) is met.

A feature’s *Life Part* (LP) corresponds to any sub-interval of the lifetime interval (i.e., this is a ‘slice’ of a feature’s life). Whereas *Minimum Life Parts* (MLPs) represent LPs where the extension of the feature is known only at the beginning and end of the LPs, but not between them. This represents the most detailed information known about a geographic feature. At the data level, an MLP corresponds to a pair of the form  $\langle \text{Star}(a, g_1, s_1), \text{Star}(a, g_2, s_2) \rangle$ , where  $a$  is feature attribute;  $s_1$  is before  $s_2$  ( $s_1 < s_2$ ); the geometries are connected to each other ( $C(g_1, g_2)$ ); and there exists *no* fact  $\text{Star}(a, g', s')$  where  $s'$  is between  $s_1$  and  $s_2$  and  $g'$  is connected to both  $g_1$  and  $g_2$ . A feature life, in turn, is represented as a sequence of consecutive MLPs.

Figure 3.12 exhibits an illustrative scenario containing different spatial regions covered by a variety of different types of coverages (where distinct types of coverages are represented by using different colours). Thus, as discussed above, from these regions and their coverages, spatial extensions of different geographic features could be identified at different time instants, as the maximal well-connected regions of particular coverages. Then Figure 3.13 illustrates 7 *Stars* (associated with a given feature attribute), representing the spatial extension of a particular feature type, inferred from regions shown in Figure 3.12. As these *Stars* meet the identity criteria of a geographic feature, they are regarded as the spatial extensions of an individual feature that lives from  $t_1$  to  $t_7$ .

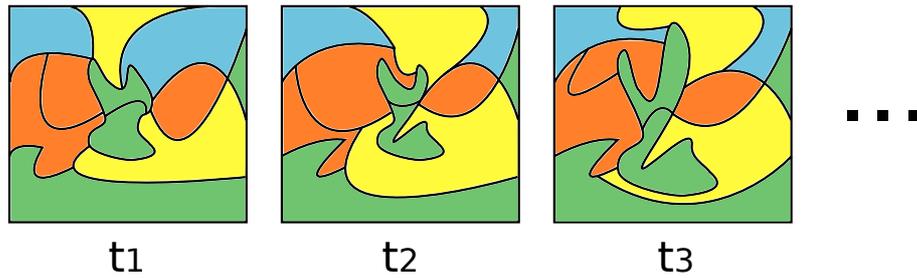


Figure 3.12: Different *Stars* representing regions with distinct coverages (at different times  $t_1, t_2, t_3, \dots, t_n$ ). Different colours are used to distinguish distinct types of coverages.

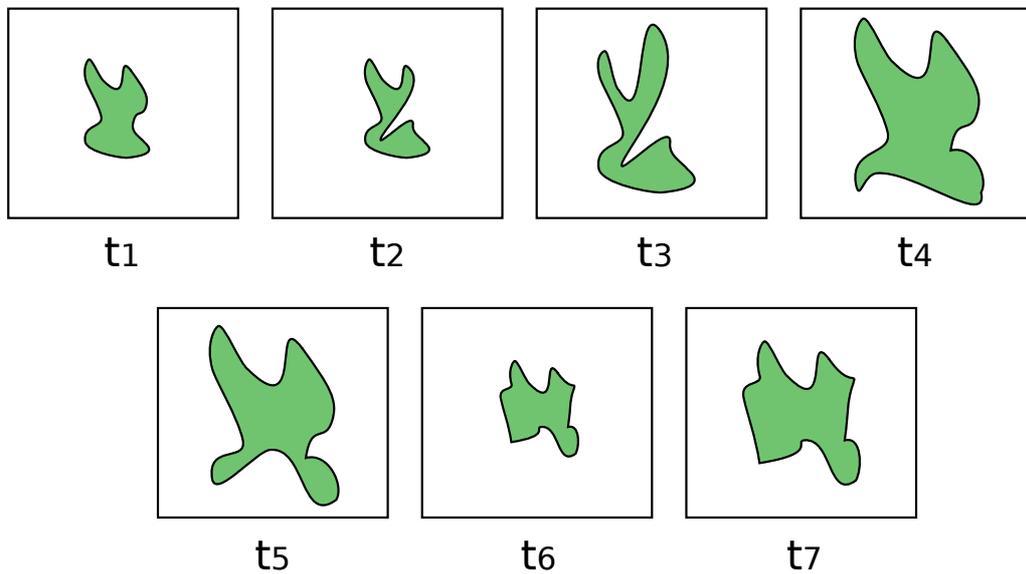


Figure 3.13: Spatial extensions of a particular geographic feature identified amongst the regions shown in Figure 3.12.

The geographic feature whose extensions over time are shown in Figure 3.13 is illustrated in Figure 3.14 as a spatio-temporal volume, representing an object which occupies a portion of geographic space at any instant of its existence. Figure 3.14 provides a visual representation of the 6 MLPs which constitute the geographic feature, represented as different slices of the spatio-temporal volume. Extensions of the feature at the beginning and end of each MLP correspond to those shown in Figure 3.13.

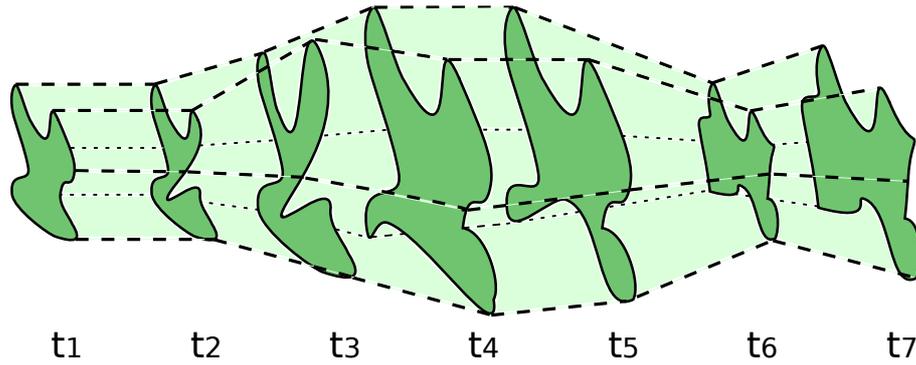


Figure 3.14: Visual representation of a geographic feature as a spatio-temporal volume. This feature contain 6 MLPs, illustrated as slices of the spatio-temporal volume. Extensions at the beginning and end of each MLP correspond to those shown in Figure 3.13.

The main goal in defining the concepts of *feature life*, *life part* and *minimum life part* is to provide an abstraction layer which allows higher level concepts describing dynamic elements of geographic space (e.g., events and processes) to be defined in terms of changing extensions of geographic features, that is, without the need to refer to lower level concepts (i.e. *Stars*). This makes the logical framework more independent from the data structure and allows the definitions of concepts relating to events and processes to be simpler and clearer.

The bridge between the logical and data levels is established by the definition of the predicate Minimum Life Part  $\text{MLP}(f, r_b, t_b, r_e, t_e)$  where  $f, r, t$  are variables of the logical language denoting, respectively, individual geographic features, spatial regions and time instants. For each instance of this predicate, the values assigned to  $r_b$  and  $r_e$  correspond to the spatial extensions of an individual feature  $f$  at time instants  $t_b$  and  $t_e$ , respectively, which represent the beginning and the end of each of  $f$ 's MLPs.

The MLP predicate is defined in terms of *Star* facts, of the form  $\text{Star}(u, r, t)$ , where  $u, r, t$  are variables of the vocabulary denoting, respectively, geographic feature types, spatial regions and time instants. This predicate is included in the logical framework to help establish the connection between the data and logical levels, by mapping them to the appropriate facts of the form  $\text{Star}(a, g, s)$ , and consequently mapping primitive concepts of the framework ( $u, r$  and  $t$ ) to elements of the domain (respectively, feature attributes, geometries, and timestamps). The MLP predicate is defined as follows.

$$\begin{aligned} \text{D 3.10} \quad \text{MLP}(f, r_b, t_b, r_e, t_e) &\equiv_{\text{def}} \exists u [u = \text{feature-type}(f) \\ &\quad \text{Star}(u, r_b, t_b) \wedge \text{Star}(u, r_e, t_e) \wedge t_b \prec t_e \wedge \text{C}(r_b, r_e)] \wedge \\ &\quad \neg \exists r', t' [(t_b \prec t' \prec t_e) \wedge \text{C}(r', r_b) \wedge \text{Star}(u, r', t')] \end{aligned}$$

Figure 3.15 illustrates different levels of abstraction for connecting the logical and data levels. In the middle layer, it can be seen both the primitive concepts of the logical framework and the concept of MLP, which is defined in terms of them. The next section describes another concept present in this layer – Spatial Change – which also helps establish the link between the top and bottom layers, so that the elements of the framework (e.g., events and processes) can be defined in a high level of abstraction.

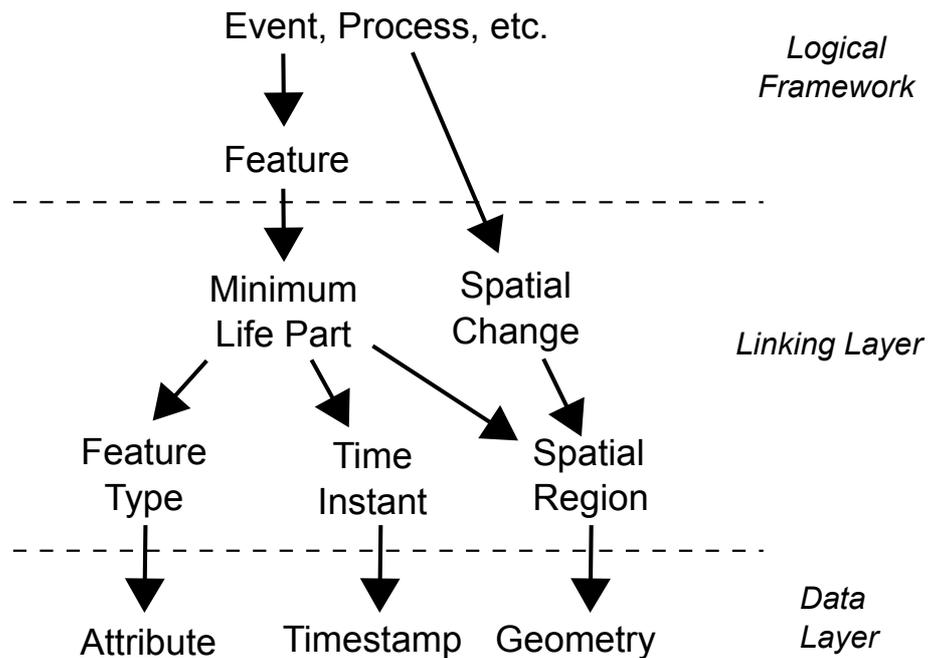


Figure 3.15: Layers of abstraction for connecting the logical and data levels.

### 3.7 Summary

This chapter presented a logic-based approach to representing spatio-temporal data and to modelling changing geographic features by establishing an explicit link between the logical and data levels. It has been shown that modelling the spatio-temporal data in a logical fashion enables the derivation of implicit data and provides a way to define changing geographic features so that they can be automatically identified at the data level.

The next chapter gives a complete description of the logical framework introduced here, which includes formal descriptions of events and process. It will be described how these concepts can be defined in terms of other abstract concepts presented in this chapter. Hence, although this grounding layer provides an explicit link between the data and the logical framework, it also makes the framework independent from data structure, as high

level concepts can be defined without referring to data elements (i.e. *Stars*).

# Chapter 4

## Logical Framework

---

### 4.1 Introduction

This chapter presents a *logical framework* for representing and reasoning about geographic events and processes. This framework, named REGEP (REasoning about Geographic Events and Processes), comprises formal descriptions of space, time, geographic features, events and processes and some relationships which hold between them. Chapter 3 introduced some elements of the logical language  $\mathfrak{R}$ , used in this framework. This Chapter gives a complete description of all the elements of  $\mathfrak{R}$ .

This chapter is organised as follows. An overview on the main motivations for the development of the REGEP framework is given in Section 4.2. Following this, Section 4.3 discusses the main fundamentals underlying the representation of events and processes<sup>1</sup>. Then Section 4.4 introduces its formal specification, by presenting its basic syntax and semantics. Other relevant predicates and relations are then presented in Sections 4.5, 4.6, 4.7, 4.8 and 4.9. Then Section 4.10 presents the approach to defining processes so that the framework becomes explicitly linked to the data level, by using the apparatus presented in Chapter 3. Finally, Section 4.11 highlights the main points discussed in the chapter.

---

<sup>1</sup>The fundamentals underlying the representation of other conceptual elements of this framework (i.e., space, time, and geographic features were discussed in Chapter 3.

## 4.2 Overview of the Problem

The Geographic Information Science community has been demanding more conceptualised ways of representing and querying geographic information, in particular information describing both space and time, enabling more comprehensive analysis of dynamic elements of geographic space. Representing geographic phenomena in terms of *events* and *processes* has been suggested by many authors [20, 27, 38, 91], and such conceptual entities appear to be significant in the way humans reason about changes affecting geographic space.

Foundational ontologies have been proposed to represent events and processes, such as BFO [15, 47, 58], DOLCE [41, 58] and SWEET [70]. Undoubtedly, upper-level ontologies can be used as useful guidelines for the development of semantic models and applications; however, they are mostly descriptive their concepts are not defined in sufficient level of detail to allow their use for reasoning purposes.

Formal theories of spatial changes [51, 84] and for modelling events and processes [6, 36, 47] have also been proposed. However, events and processes are often approached in the general sense, and their applicability to the geographic domain still requires further extensions and refinement. Moreover, although some works provide important directions, most of them are not yet implemented, and therefore their suitability for handling real-world data is not often discussed.

The *logical framework* presented in this chapter places a particular focus on the representation of geographic events and processes, encompassing their relationship with geographic features, which are said to participate in them. Of particular interest are geographic phenomena which can be described in terms of changes affecting the spatial extension of geographic features. Examples are deforestation, urbanisation and desertification, which can be described, respectively, in terms of changes affecting the spatial extension of forests, urbanised areas (e.g., cities) and deserts.

As discussed in Section 2.6, defining an appropriate representation for geographic events and processes requires dealing with issues regarding the relationship between these concepts and also between them and geographic features. Moreover, as discussed in Section 2.8.3, other crucial issues are how to define the relation between event and process types and their particular instances, and how to handle different kinds of *vagueness* to associate specific spatial and temporal boundaries with process instances.

Hence the REGEP framework includes an approach to handling spatio-temporal vagueness based on standpoint semantics [10], which enables the proposed reasoning mechanism to define temporal boundaries for geographic processes so that particular in-

stances of a given process type can be determined based on individual viewpoints.

There has been many disagreements in the literature about the appropriate representation of events and processes. The debate covers issues relating to the classification of these entities either as *endurants* or *perdurants* and tackles questions such as whether these entities can be affected by temporal gaps, or whether they possess the characteristic of undergoing change over time. Particular disagreements are also related to the way events and processes are interrelated (e.g., whether one is a subclass of the other). For this reason, many existing approaches have avoided providing precise logical definitions for these concepts and the relations between them.

However, the objective of the development of the formalism presented in this Chapter is not to enter into this debate by defending that this approach is the most appropriate from the philosophical point of view. Differently, the aim of this work is to develop a formal approach which considers the semantic analysis discussed in previous work and which provides representational flexibility for accommodating distinct viewpoints. Moreover, it is intended to produce a formalism in which the concepts are defined in a level of detail that enables reasoning; and that provides a comprehensive formal apparatus for the implementation of a system which can process real geographic data, as a contribution for the development of modern GIS with stronger basis on theory.

### 4.3 Events and Processes

Events are regarded here as *perdurant entities*, that is, entities whose properties are possessed timelessly and therefore are not subject to change over time. This is perhaps the most accepted view amongst different authors (e.g., [15, 35, 36, 47, 58]). On the other hand, a process is regarded as an entity which is subject to change over time (e.g., a process may be said to be accelerating or slowing down), and therefore a process is *not* regarded as perdurant entity as defended by some authors (e.g., [41, 47, 58]). Processes are conceived here in agreement with the concept of *time-dependent entities* described by Galton [35, 36].

In the work of this thesis, events are conceived in agreement with Galton's [35] view that "event is not something that can be said to exist from moment to moment in this way, rather it is something that, once it has happened, we can retrospectively ascribe to the time interval over which it occurred" (p. 04). Thus 'the forest is shrinking' does not describe an event, but rather a *process active* at a certain time instant. An event is usually associated with *precise temporal boundaries*, which may be denoted by the *culmination of a process* (i.e., when the goal in initiating it is realised). Hence, once after determined

this instant of culmination, one can retrospectively ascribe the shrinkage event to the time interval over which it occurred.

Some authors support the view that processes are entities which are regarded as self-connected wholes, and therefore cannot contain temporal gaps. For instance, Grenon and Smith [47] state that ‘processes have beginnings and endings corresponding to real discontinuities, which are their bona fide boundaries’ (p. 153). Moreover, the authors argue that ‘a given process may not be occurring at two distinct times without occurring also at every time in the interval between them’ (p. 153). Nevertheless, this assumption is still the subject of controversy for the modelling of geographic processes, since there are many examples of natural language descriptions of geographic processes where the existence of gaps seems to be acceptable.

To illustrate, suppose one intends to monitor deforestation in a given forest based on spatial data collected once a day, every day. Then it was observed that the forest shrank every day during 300 days, except between the 84<sup>o</sup> and the 86<sup>o</sup> days, and between the 145<sup>o</sup> and the 165<sup>o</sup> days. Deciding whether the same instance of a process proceeded over such 10-month period or distinct instances were separated by those periods of inactivity might depend on many factors. Judgement variables include the sort of geographic phenomena which is being analysed (e.g. deforestation), the agents involved (e.g. human action or wildfire originated from spontaneous combustion), the purpose (e.g. deforestation caused by human actions with purpose of wood trading), amongst others.

Hence, in the REGEP framework, a process is regarded as an entity which may be affected by temporal gaps. The approach to determining processes’ boundaries aims to provide a flexible mode of representing and reasoning about geographic processes. This approach is based on standpoint semantics and therefore threshold parameters are given to determine the range of variation over which the predicate is judged to be applicable.

## 4.4 Syntax and Semantics

This section describes the syntax and semantics of the logical language  $\mathfrak{R}$  employed in the framework. Relevant predicates and logical relations employed in this framework shall be introduced in Sections 4.5, 4.6, 4.7, 4.8 and 4.9. The formalism is described using definitions and axioms in first order logic, indexed by D and A, respectively. Free variables are implicitly universally quantified with maximal scope.

### 4.4.1 Syntax

This section describes the basic syntax of the logical language named  $\mathfrak{R}$  employed in the framework. Relevant predicates and logical relations employed in this framework shall be defined in Sections 4.8, 4.9 and 4.10.

The logical language  $\mathfrak{R}$  used in the framework comprises variables of 12 *nominal types* which can be quantified over. These types of variables are listed below, and the denotation of some of them shall be discussed later in this section. The vocabulary of  $\mathfrak{R}$  can be specified by a tuple  $\mathcal{V} = \langle \mathcal{V}_t, \mathcal{V}_i, \mathcal{V}_r, \mathcal{V}_f, \mathcal{V}_c, \mathcal{V}_u, \mathcal{V}_v, \mathcal{V}_e, \mathcal{V}_\varepsilon, \mathcal{V}_b, \mathcal{V}_p, \mathcal{V}_\pi \rangle$ , including:

- Time Instants,  $\mathcal{V}_t = \{\dots, t_i, \dots\}$
- Time Intervals,  $\mathcal{V}_i = \{\dots, i_i, \dots\}$
- Spatial Regions,  $\mathcal{V}_r = \{\emptyset, \dots, r_i, \dots\}$
- Geographic Features,  $\mathcal{V}_f = \{\dots, f_i, \dots\}$
- Coverage Types,  $\mathcal{V}_c = \{\dots, c_i, \dots\}$
- Feature Types,  $\mathcal{V}_u = \{\dots, u_i, \dots\}$
- Event-classifiers,  $\mathcal{V}_v = \{\dots, v_i, \dots\}$
- Event-types,  $\mathcal{V}_e = \{\dots, e_i, \dots\}$
- Event-tokens,  $\mathcal{V}_\varepsilon = \{\dots, \varepsilon_i, \dots\}$
- Process-classifiers,  $\mathcal{V}_b = \{\dots, b_i, \dots\}$
- Process-types,  $\mathcal{V}_p = \{\dots, p_i, \dots\}$
- Process-tokens,  $\mathcal{V}_\pi = \{\dots, \pi_i, \dots\}$

#### 4.4.1.1 Functional Terms

The logical language  $\mathfrak{R}$  contains the following *functions* to transfer information between distinct semantic types.

- $\text{event} : (\mathcal{V}_v \times \mathcal{V}_f) \rightarrow \mathcal{V}_e$ , that gives the event-type corresponding to the association between the specified event classifier and participant geographic feature.
- $\text{process} : (\mathcal{V}_b \times \mathcal{V}_f) \rightarrow \mathcal{V}_p$ , which returns the process-type corresponding to the association between the given process classifier and participant geographic feature.

- f-type :  $\mathcal{V}_f \rightarrow \mathcal{V}_u$ , that returns the type of a certain geographic feature.
- c-type :  $\mathcal{V}_r \rightarrow \mathcal{V}_c$ , which gives the coverage type of a certain spatial region.
- ext :  $(\mathcal{V}_f \times \mathcal{V}_t) \rightarrow \mathcal{V}_r$ , which returns the spatial region corresponding to the spatial extension of a feature  $f$  at time instant denoted by  $t$ .
- ext :  $\mathcal{V}_f \rightarrow \mathcal{V}_r$ , that returns the spatial region corresponding to the union of all spatial regions occupied by a feature  $f$  throughout its life.
- b :  $\mathcal{V}_i \rightarrow \mathcal{V}_t$ , which returns an instant  $t$  corresponding to the beginning of an interval  $i$ ;
- e :  $\mathcal{V}_i \rightarrow \mathcal{V}_t$ , which returns an instant  $t$  corresponding to the end of an interval  $i$  corresponding to the beginning and the end of a given time interval.

#### 4.4.1.2 Auxiliary Functions

The following *auxiliary functions* are employed to perform calculations over elements of the domain:

- length :  $\mathcal{V}_i \rightarrow \mathbb{Z}$ , which gives an integer value representing the length of a given interval  $i$ .<sup>2</sup>
- area :  $\mathcal{V}_i \rightarrow \mathbb{R}$ , that returns a real number representing the area of a region  $r$ .
- distance :  $(\mathcal{V}_r \times \mathcal{V}_r) \rightarrow \mathbb{Z}$ , which returns a non-negative number representing the 2-dimensional Cartesian minimum distance between two regions in projected units.
- union :  $(\mathcal{V}_r \times \mathcal{V}_r) \rightarrow \mathcal{V}_r$ , which returns a spatial region that corresponds to the spatial union of a pair of spatial regions.
- concave-hull  $(\mathcal{V}_r \times \mathcal{V}_r) \rightarrow \mathcal{V}_r$ , which returns a concave region that encloses the two specified regions.<sup>3</sup>

<sup>2</sup>The value returned by this function is an integer number which may vary according to the temporal granularity (e.g., days, microseconds) adopted to represent elements of domain, however this change does not affect the semantics.

<sup>3</sup>The concave hull algorithm used for implementing the system prototype is described in Chapter 5.

### 4.4.1.3 Atomic Propositions

Atomic propositions of  $\mathfrak{R}$  include the following binary logical relations:

- The 13 Allen relations between time intervals [1, 2] (described in Section 2.3).
- The operators  $\prec$ ,  $=$ , and  $\preceq$ , employed to represent temporal precedence and equality between instants.
- $\text{In}(i_1, i_2)$ , which is true if the time interval  $i_1$  is in the time interval  $i_2$  (see Definition D3.1 in Chapter 3).
- $\text{In}(t, i)$ , which is true just in case the time instant  $t$  is in the interval  $i$  (see Definition D3.2 in Chapter 3).
- The following RCC relations between spatial regions: *connected*  $C(\alpha, \beta)$ , *disconnected*  $DC(\alpha, \beta)$ , *overlaps*  $O(\alpha, \beta)$ , *externally connected*  $EC(\alpha, \beta)$ , *part of*  $P(\alpha, \beta)$ , *proper part of*  $PP(\alpha, \beta)$  and *equals to*  $EQ(\alpha, \beta)$ , where  $\alpha$  and  $\beta$  are region terms which may be either a region variable  $r_i$ , a term of the form  $\text{ext}(f_i, t_i)$  or the empty region constant  $\emptyset$ .
- $r_1 =_c r_2$ ,  $r_1 \neq_c r_2$  are true, respectively, just in case the spatial region term  $r_1$  denotes a region with the same type of coverage of region denoted by  $r_2$ ;  $r_1$  denotes a region with different type of coverage of  $r_2$ .
- $f_1 = f_2$ , which is true if  $f_1$  and  $f_2$  are geographic features which have the same identity criteria.

### 4.4.1.4 Propositional Constructs

$\mathfrak{R}$  also includes *propositional constructs*, which have one of the following forms:

- Holds-At( $\varphi, t$ ) and Holds-On( $\varphi, i$ ) assert that formula  $\varphi$  is true, respectively, at the time instant denoted by  $t$ ; and at every time instant  $t$  where  $\text{In}(t, i)$ .
- Occurs-On( $e, i$ ) asserts that an event of type  $e$  occurs on a time interval  $i$ .
- If  $\varphi$  and  $\psi$  are propositions of  $\mathfrak{R}$ , then so are the following:  $\neg\varphi$ ,  $(\varphi \wedge \psi)$ ,  $(\varphi \vee \psi)$ ,  $(\varphi \rightarrow \psi)$ ,  $\forall v[\varphi]$ , where  $v$  is a variable of one of the nominal types described earlier.

### 4.4.2 Semantics

An *attributed geographic model* is a structure  $\mathcal{M} = \langle \mathfrak{G}, \mathcal{V}, \mathcal{A} \rangle$ , where:

- $\mathfrak{G} = \langle \mathbb{R}^2, \langle T, \trianglelefteq \rangle, A, \mathcal{D} \rangle$  is a formal model of a geographic dataset:
  - $\mathbb{R}^2$  is the real plane, which will represent a portion of the earth’s surface under some specified projection,
  - $T$  is a set of time instants,
  - $\trianglelefteq$  is a total linear order over  $T$ ,
  - $A$  is a set of geographic attributes,
  - $\mathcal{D} \subseteq A \times \text{Poly}(\mathbb{R}^2) \times T$  represents the geographic attributed data as a set of tuples of the form  $\langle a, p, s \rangle$ , which correspond to the fact that attribute  $a$  holds for polygon  $p$  over instant  $s$ .<sup>4,5</sup>
- $\mathcal{V} = \langle \mathcal{V}_t, \mathcal{V}_i, \mathcal{V}_r, \mathcal{V}_f, \mathcal{V}_c, \mathcal{V}_u, \mathcal{V}_v, \mathcal{V}_e, \mathcal{V}_\varepsilon, \mathcal{V}_b, \mathcal{V}_p, \mathcal{V}_\pi \rangle$ , specifies the vocabulary of the representation language  $\mathfrak{R}$ . Each element of this tuple is the set of all symbols of a given type (as specified in the syntax section).
- $\mathcal{A} = \langle a_t, a_i, a_r, a_f, a_c, a_u, a_v, a_e, a_\varepsilon, a_b, a_p, a_\pi \rangle$  is a tuple of assignment functions specifying the denotations of all symbols in the vocabulary as follows:
  - $a_t : \mathcal{V}_t \rightarrow T$  maps time instant variables to time instants.
  - $a_i : \mathcal{V}_i \rightarrow \text{Int}(T)$ <sup>6</sup> maps interval variables to intervals.
  - $a_r : \mathcal{V}_r \rightarrow \text{Reg-Closed}(\mathbb{R}^2)$  maps region variables to regular closed<sup>7</sup> regions of the plane.
  - $a_f : \mathcal{V}_f \rightarrow (T \rightarrow \text{Poly}(\mathbb{R}^2))$  maps each feature symbol to a function from time instants to polygons (giving the spatial extension of the feature at each time instant).

<sup>4</sup>Here,  $\text{Poly}(\mathbb{R}^2)$  is the set of well-connected polygons over  $\mathbb{R}^2$ .

<sup>5</sup>In the geographic attributed dataset, attributes are employed to represent possible types of region coverage and types of feature. Therefore, in the semantic model, elements of type *feature type* and *regions coverage type* are used to represent existing corresponding attributes in the data model.

<sup>6</sup> $\text{Int}(T) = \{ \langle t_1, t_2 \rangle \mid t_1, t_2 \in T \wedge t_1 < t_2 \}$  is the set of all proper intervals over the time sequence  $\langle T, \trianglelefteq \rangle$ .

<sup>7</sup>A region  $r$  is said to be regular closed if  $r$  is the closure of its interior, i.e.,  $r = \text{closure}(\text{interior}(r))$ , where *interior*( $r'$ ) specifies the topological interior of a spatial region and *closure*( $r'$ ) the topological closure. Roughly, this means that the region contains no “loose points” and no “hanging lines”.

- $a_c : \mathcal{V}_c \rightarrow (T \rightarrow \text{Reg-Closed}(\mathbb{R}^2))$  maps coverage types to functions from time instants to regular closed regions of the plane. This gives the extension of the region having a given type of coverage at each time instants. In general this will be a multi-piece region.
- $a_u : \mathcal{V}_u \rightarrow 2^{(T \rightarrow \text{Poly}(\mathbb{R}))}$  maps each feature type to a set of mappings from time instants to polygons.
- $a_v : \mathcal{V}_v \rightarrow (\mathcal{V}_f \rightarrow 2^{\text{Int}(T)})$  maps each event-classifier symbol to a function from features to intervals (giving, for each feature, an interval over which an event occurs involving that feature).
- $a_e : \mathcal{V}_e \rightarrow (\mathcal{V}_v \times \mathcal{V}_f \times \text{Int}(T))$  maps each event-token symbol to a triple consisting of an event-classifier, a feature (the participant) and an interval (the interval over which this particular event-token occurs).  
Where if  $a_e(e) = \langle v, f, i \rangle$  then  $i \in a_v(v)(f)$ .
- $a_b : \mathcal{V}_b \rightarrow (\mathcal{V}_f \rightarrow 2^T)$  maps each process-classifier symbol to a function from features to set of time instants (giving, for each feature, a set of time instants corresponding to the time intervals over which a process proceeds involving that feature).<sup>8</sup>
- $a_\pi : \mathcal{V}_\pi \rightarrow (\mathcal{V}_b \times \mathcal{V}_f \times \text{Int}(T))$  maps each process-token symbol to a triple consisting of an process-classifier, a feature (the participant) and an interval (the interval over which this particular process-token proceeds).

## 4.5 Representing Events

This section describes the elements of the logical framework employed to represent events (i.e., event classifiers, types and tokens) and presents elementary logical relations which hold between events and geographic features. The semantic categorisation used in this formalism is based on that used in Versatile Event Logic (VEL) [6]. The representation of event occurrences shall be discussed in Section 4.8 and in Section 4.9.1.

*Event classifiers* identify general categories of events, independently of particular occurrences or participants. That is, it describes something that might happen in space and

---

<sup>8</sup>As discussed later in this Chapter, a process is regarded here as a temporal entity which may be *inactive* for certain time intervals within which an interval  $i$  over which the process is said to proceed. Thus, since the logical language deals with convex time intervals, this assignment function maps processes to set of time instants corresponding to subintervals  $i'$  of  $i$  where a process is *active*.

time without specifying any temporal information or relating any type of geographic feature. *Natural language verbs* (in the third person singular conjugation) are usually applied to name these classifiers. Examples of such verbs are ‘falls’, ‘expands’ and ‘shrinks’. Verbs whose denotation is attached to a particular sort of spatial object are avoided, such as ‘rains’. In this case, it would be preferred to represent ‘the raining event’ in terms of the ‘fall of raindrops’, for instance.

This sort of abstraction is desired in the representation of events so that a wider range of geographic phenomena can be represented by associating distinct geographic features with events classifiers. For example, ‘desertification’ and ‘urbanisation’ could be represented in terms of *expansion* of ‘arid’ and ‘built-up’ regions, respectively. This is particularly applicable to model the type of phenomena addressed in this work, which are those which can be represented in terms of spatial transformations of geographic features. Nonetheless, defining event classifiers at this level of abstraction is not a requirement for the applicability of this representational approach. That is, more specific event classifiers can be adopted in cases where the knowledge engineer finds more appropriate their use for modelling a particular domain or situation.

Events are also structured in terms of *types* and *tokens*. An *event type* involves a particular instance of a *geographic feature* as its participant. On the other hand, an *event token* denotes a *particular occurrence* of an event type, and is therefore associated with a *time interval* on which it occurs. For example, ‘Amazon rainforest shrinks’ describes an event type, since this might occur different times, corresponding to different instances of this event. Whereas ‘Amazon forest shrank from May/2006 to July/2006’ describes a particular occurrence of this type, that is, an event token. As seen in this last example, an event token can be referred to by using the past simple tense of the verb corresponding to its classifier, together with an explicit specification of a time interval.

A reified representation for event classifiers is adopted. Therefore an event type  $e$  denoting an expansion, for example, is related to an event classifier by  $\text{Event-Classifier}(\textit{expands}, e)$ . On the other hand, an event type is treated as complex nominals (i.e. functional terms). Thus an event type  $e$  is represented by  $e = \textit{event}(v, f)$ , where  $v$  is an event classifier and  $f$  a geographic feature which participates in this event.

The relation between an event classifier  $v$  and an event type  $e$  is defined as follows:

$$\text{D 4.1} \quad \text{Event-Classifier}(v, e) \equiv_{def} \exists f [e = \text{event}(v, f)]$$

For convenience, the following relation is also defined:

$$\text{D 4.2} \quad \text{Participant-In-Event}(f, e) \equiv_{def} \exists v [e = \text{event}(v, f)]$$

An event token is represented by a pair  $\varepsilon = \langle e, i \rangle$ , where  $e$  is an event type and  $i$  is the interval of occurrence of this event token. However, not all possible pairs  $\langle e, i \rangle$  denote an existing event token. Thus, the subset of existing event tokens is given by those pairs for which the proposition  $\text{Occurs-On}(e, i)$  is true.

## 4.6 Representing Processes

This section describes the elements of the REGEP framework employed to represent processes (i.e., process classifiers, types and tokens) and presents essential logical relations which hold between processes and geographic features. A more extensive discussion on the representation of geographic processes shall be given in Sections 4.8 and 4.9.

As discussed for events, *process classifiers* are used to describe processes without any association with temporal information or participants. Processes are also structured in terms of *types* and *tokens*. Whereas a process type denotes a series of changes involving a particular feature, a process token denotes an instance of a certain type of process. Hence, tokens are said to *proceed* on a specific time interval. The structure employed to represent process classifiers, process types, process tokens and the relation between processes and their participants is analogous to the one employed for events. Thus the following relations are also defined.

$$\text{D 4.3} \quad \text{Process-Classifier}(b, p) \equiv_{def} \exists f [p = \text{process}(b, f)]$$

$$\text{D 4.4} \quad \text{Participant-In-Process}(f, p) \equiv_{def} \exists b [p = \text{process}(b, f)]$$

A process token is represented by a pair  $\pi = \langle p, i \rangle$ , where  $p$  is a process type and  $i$  is the interval over which this process token is said to proceed. Therefore the subset of valid process tokens is composed by those pairs for which the proposition  $\text{Proceeds}[a_{th}](p, i)$

is true ( $[a_{th}]$  is a standpoint semantics parameter applied to handle temporal vagueness in this predicate and is described in detail in Section 4.9).

## 4.7 Relating Events and Processes

There have been disagreements in the literature about how *processes* and *events* are related to each other. While Sowa [82] defines event as a subclass of process, Pustejovsky [68] defines process as a subclass of event. They are also described as non-overlapping categories [2] or as subclasses of the same class (occurrence) [61]. Galton [36] suggests that events and processes can be related in many different ways (events can be described in terms of other events or in terms of processes, while processes can be described in terms of events or in terms of other processes).

According to Galton [36], some processes can be described in terms of their constituent events, whilst some events can be described as a ‘chunk of a process’. These relations between events and processes are of particular interest here. In the REGEP framework, the relation  $\text{Constituted-Of}(b, v)$  associates an event classifier with a process classifier. Asserting a fact using this relation means that occurrences of an event classified by  $v$  over a certain interval denote that a process classified by  $b$  proceeds in that interval. Conversely, the relation  $\text{Is-Chunk-Of}(v, b)$  associates a process classifier with an event classifier. Asserting a fact using this relation means that the occurrence of an event (classified by  $v$ ) on given time interval  $i$ , is determined by the fact that a process (classified by  $b$ ) proceeds on  $i$ . The meaning of these relationships between events and processes shall be further clarified in Section 4.8 and in Section 4.9.1.

## 4.8 Process Activeness and Event Occurrences

Existing spatio-temporal datasets do not usually consist of explicit assertions of events occurrences or process activity. Rather, they often contain elements describing changes of objects’ properties over time, from which events and processes can be inferred. For example, movements of objects can be identified from data on the position of such objects at different time instants. Similarly, the expansion of a built-up region can be inferred from data describing its spatial extensions at different times.

Deciding on whether data best reproduce the intended denotation of events or processes is a crucial issue for the development of a theoretical framework which is supposed to be implemented to operate on real datasets. On the one hand, a piece of data representing a single change affecting a feature (e.g., an increase in area between two distinct time

instants  $t_1$  and  $t_2$ ) is not sufficient to infer the occurrence of an event, since information on the feature's area after  $t_2$  would be required to determine the precise temporal boundary which is expected for an event. On the other hand, such a piece of data would not reproduce the density and homogeneity characteristics associated with a process. That is, whereas it can be inferred that an expansion process was active for some time between  $t_1$  and  $t_2$ , it is not possible to ensure that the process was active during the whole interval.

The approach taken for building this logical framework consists of abstracting away from the lack of information for representing the activeness of process. This is to say that, if it is known that a geographic feature changes from instant  $t_1$  to  $t_2$  and nothing is known about the period between them, it is assumed that the process which characterises this change is active at all time instants from  $t_1$  to  $t_2$ . The *activeness* of a process is represented by the predicates  $\text{Active-At}(p, t)$  and  $\text{Active-On}(p, i)$ . Whilst the former determines that a process of type  $p$  is active at a time instant  $t$ , the latter specifies that a process is active on a time interval  $i$ , meaning it is going on at every time instant within that interval.

Since the former is defined in a low level manner, that is, closer to the way data is represented, a discussion on its definition shall be given in Section 4.10.2, after introducing some other required. Whereas the latter is defined as follows.

$$\text{D 4.5} \quad \text{Active-On}(p, i) \equiv_{def} \forall t [\text{In}(t, i) \rightarrow \text{Active-At}(p, t)]$$

Beyond the predicates to represent the activeness of a process, it is also convenient to define a predicate which verifies whether a process is inactive at a certain time instant or on a given time interval. The latter is particularly useful for defining the predicate which determine event occurrences (presented later in this section) and the predicate which specifies whether a process proceeds (presented in Section 4.9). It should be noticed that there is a difference between a process being inactive during an interval and it not being active, that is,  $\text{Inactive-On}(p, i) \not\equiv \neg \text{Active-On}(p, i)$ . For instance, if an interval includes some parts where a process is active and others where it is not active, then  $\neg \text{Active-On}(p, i)$  will hold, but  $\text{Inactive-On}(p, i)$  should not hold. Hence, the predicates denoting a process inactivity are defined as follows.

$$\text{D 4.6} \quad \text{Inactive-At}(p, t) \equiv_{def} \neg \text{Active-At}(p, t)$$

$$\text{D 4.7} \quad \text{Inactive-On}(p, i) \equiv_{def} \forall t [\text{In}(t, i) \rightarrow \neg \text{Active-At}(p, t)]$$

It was discussed that an event can be said as 'made of' a process, and that the culmi-

nation of the process (i.e., when the goal in initiating it is realised) denotes the occurrence of an event. Therefore, in this logical framework, an event token is modelled as a chunk of a process bounded by temporal discontinuities, meaning that the process is inactive on both time intervals which meets and is met by the interval on which the event occurs.

Furthermore, since it is accepted that events can also be regarded as constituent of processes, this might lead to a continuous cycle. Thus it is important to distinguish event tokens which cannot contain other events of the same type. These are called *primitive* event tokens. Expressly, if a primitive event  $e$  occurs on an interval  $i$ , there is no sub-interval of  $i$  on which an event of the same type of  $e$  occurs.

It can be seen that allowing the existence of nested event tokens of the same type implies that events can also be affected by gaps, since an event token is necessarily bounded by temporal discontinuities, as discussed above. Therefore another property of primitive event tokens is that they are not affected by temporal gaps, meaning that the process of which it is made must be active throughout the whole interval on which the event is said to occur. The predicate to represent a primitive event occurrence is defined in D4.8, whilst the representation of non-primitive ones is discussed in Section 4.9.

$$\begin{aligned}
 \text{D 4.8} \quad \text{Occurs-On-Prim}(e, i) \equiv_{\text{def}} & \exists v p b i' i'' [ \\
 & e = \text{event}(v, f) \wedge p = \text{process}(b, f) \wedge \\
 & \text{Is-Chunk-Of}(v, b) \wedge (\forall t [\text{In}(t, i) \rightarrow \text{Active-At}(p, t)]) \wedge \\
 & \text{Meets}(i', i) \wedge \text{Met-by}(i'', i) \wedge \\
 & \text{Inactive-On}(p, i') \wedge \text{Inactive-On}(p, i'') ]
 \end{aligned}$$

Figure 4.1 exhibits a stretch of the timeline to illustrate possible primitive event tokens for an event type  $e$  (bold lines). In this figure, 3 occurrences of the same event type  $e$  are shown (on the intervals  $i_1$ ,  $i_2$  and  $i_3$ ). On the other hand, no event of type  $e$  occur (as a primitive token) on intervals  $i_4$ ,  $i_5$  or  $i_6$ , as such occurrences would not be in accordance with the properties of primitive event tokens discussed above.

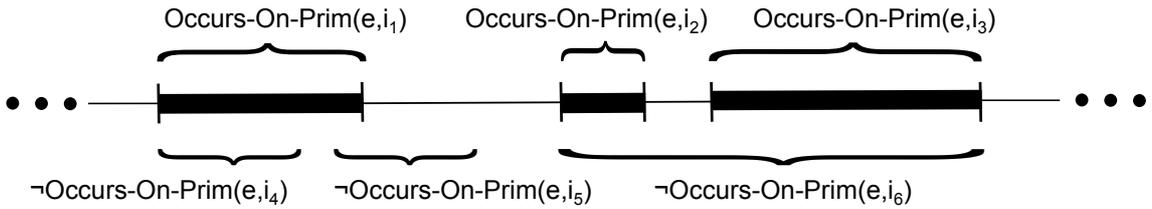


Figure 4.1: Stretch of the timeline exhibiting primitive event tokens of type  $e$  (bold lines). 3 distinct occurrences of the same event type ( $e$ ) are shown (on intervals  $i_1$ ,  $i_2$  and  $i_3$ , respectively). Primitive event tokens are modelled as bounded by temporal discontinuities, containing no temporal gaps, and no event of the same type can occur in a sub-interval of the interval on which it occurs. Thus no primitive event token of type  $e$  is identified on intervals  $i_4$ ,  $i_5$  or  $i_6$ .

Notice that the possibility of two event tokens of distinct types occurring in parallel was not ruled out. That is, given two intervals  $i_1$  and  $i_2$  on which tokens of two different event types occur, propositions  $\text{Overlap}(i_1, i_2)$  or  $\text{During}(i_1, i_2)$  might hold. For example, since different event types can affect the same geographic feature, such events could represent an object which expands and rotate over the same period of time. Similarly, the definitions presented here do not rule out that an event token can be followed by another token of a different type, that is, proposition  $\text{Meets}(i_1, i_2)$  might hold where  $i_1$  and  $i_2$  are time intervals on which two events of different types occur (even if they affect the same geographic feature).

## 4.9 Determining Process Boundaries

Section 4.8 described an approach to representing an event occurrence based on the fact that a process is active throughout the whole interval on which the event occurs. However, as discussed in Section 4.3, a process may be affected by *temporal gaps*. Such temporal gaps are represented here as *periods of inactivity* of a process. Thus a given process may be regarded as active and inactive at different times within an interval on which it is said to proceed. Such interval determines the explicit boundaries for a process and is applied to identify distinct instances of a certain process type.

The approach to determining such boundaries of geographic processes is based on standpoint semantics, so that particular instances of a given process type can be determined based on individual viewpoints. Therefore it is said that a process of type  $p$  proceeds on a time interval  $i$  if any subinterval  $i'$  of  $i$  on which it is inactive is of length less than a given *activeness threshold*  $a_{th}$  (i.e.,  $a_{th}$  is a standpoint parameter for that predicate).

This definition, however, leads to another important representational issue. As discussed so far, it seems acceptable to say that a process may proceed over relatively short time intervals between its constituent events, representing periods of inactivity of that process. However, it seems controversial to say that a process proceeds on a time interval  $i$  which is started or finished by periods of inactivity, without taking into account the past or future to look for occurrences of constituent events on intervals outside  $i$ . To illustrate, suppose a process whose inactivity threshold is 2 weeks. Then assume the process has been inactive for 5 days until the present time. It could be said the process is still active, since 5 days  $<$  14 days. However, how could one guess the process has not reached its culmination point?

This issue is addressed by defining a predicate  $\text{Proceeds-On-Max}[a_{th}](p, i)$ , which captures the maximum interval  $i$  over which a process proceeds. Hence the interval  $i$  must be bounded by temporal discontinuities which are *longer* than the threshold  $a_{th}$ . In other words,  $i$  is the maximum interval on which there exists no significant temporal gaps (longer than the specified threshold) in the process. This predicate is defined as follows.

$$\begin{aligned}
 \text{D 4.9} \quad \text{Proceeds-On-Max}[a_{th}](p, i) &\equiv_{def} \\
 &\exists i' i'' [\text{Starts}(i', i) \wedge \text{Active-On}(p, i') \wedge \\
 &\quad \text{Finishes}(i'', i) \wedge \text{Active-On}(p, i'') \wedge \text{Before}(i', i'')] \wedge \\
 &\exists i' i'' [\text{Meets}(i', i) \wedge \text{length}(i') > a_{th} \wedge \text{Inactive-On}(p, i') \wedge \\
 &\quad \text{Met-by}(i'', i) \wedge \text{length}(i'') > a_{th} \wedge \text{Inactive-On}(p, i'')] \wedge \\
 &\forall i' [\text{In}(i', i) \wedge \text{Inactive-On}(p, i') \rightarrow \text{length}(i') \leq a_{th}]
 \end{aligned}$$

According to the predicate described above, there may be many different intervals  $i$  on which a process of the same type (and consequently affecting the same individual geographic feature) proceeds. Therefore each of these intervals determines the temporal boundaries of an *individual* process, whose spatial boundaries are established by the spatial extension of participant geographic features. Once processes are individuated, it is possible to determine different (and possibly non-overlapping) intervals on which the same individual process is said to proceed. This is specified by the the predicate  $\text{Proceeds-On}[a_{th}](p, i)$ , which is defined as follows.

$$\text{D 4.10} \quad \text{Proceeds-On}[a_{th}](p, i) \equiv_{def} \exists i' [\text{Proceeds-On-Max}[a_{th}](p, i') \wedge \text{In}(i, i')]$$

Figures 4.2 and 4.3 illustrate different situations in which a process of type  $p$  is said to proceed. Figure 4.2 shows a process token which proceeds on interval  $i$ . Although

two periods of inactivity are identified within the process token, they are ignored as their duration are shorter than the specified activeness threshold  $a_{th}$ . The example of Figure 4.2 also exhibits how the logical constructs that appear in the Definition 4.9 match the illustrative situation.

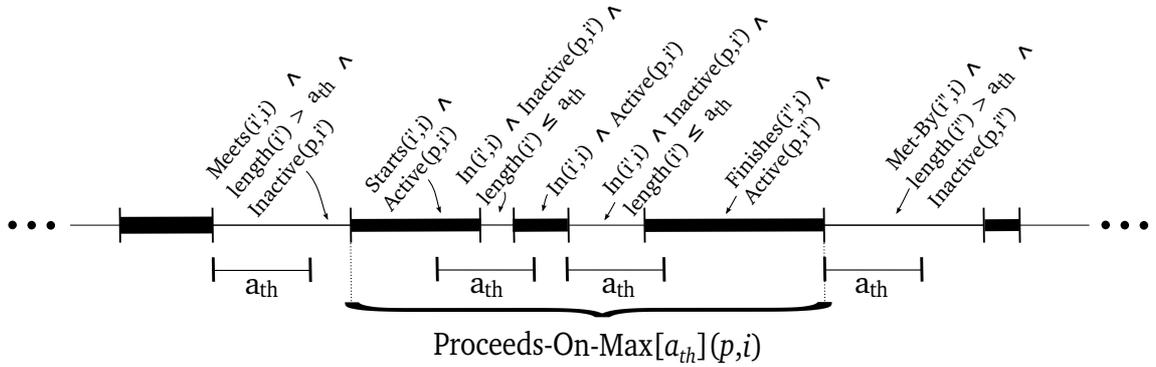


Figure 4.2: Stretch of the timeline exhibiting a process of type  $p$  which proceeds (maximally) on an interval  $i$ , where  $a_{th}$  represents the specified activeness threshold. The process token shown contains 3 distinct intervals on which the process is active; contains two intervals on which the process is inactive but whose lengths are shorter than the threshold; and is bounded by intervals over which the process is inactive and whose lengths are longer than  $a_{th}$ .

On the other hand, in the example of Figure 4.3, a smaller threshold is specified (compared to that shown in Figure 4.2), meaning that shorter periods of inactivity are permitted to regard that the process token exists. Figure 4.3 also illustrates distinct cases in which  $\text{Proceeds-On}(p, i)$  would hold for the proposed scenario.

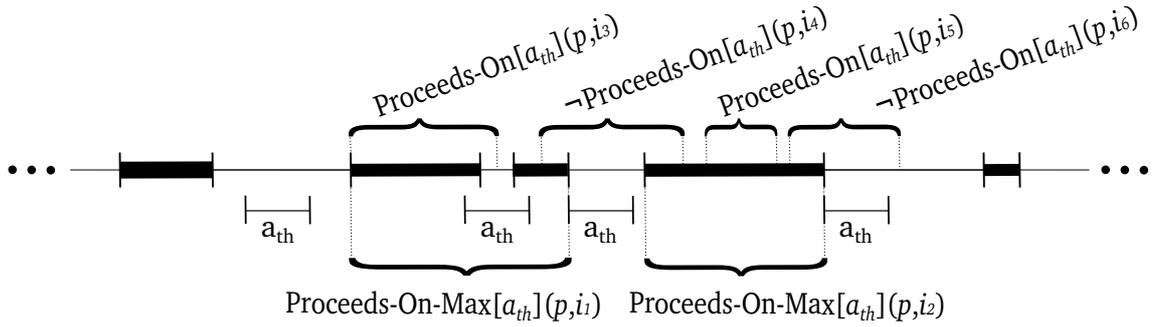


Figure 4.3: The same stretch of the timeline shown in Figure 4.2. However, in this example, a shorter activeness threshold  $a_{th}$  has been specified. Consequently, now two distinct process tokens are identified for intervals  $i_1$  and  $i_2$  (both of type  $p$ ). These tokens are separated by an interval over which the process is inactive and whose length is longer than  $a_{th}$ .

### 4.9.1 Deriving Events from Processes

It has been shown that primitive events are defined in terms of process activeness. Moreover, it has been discussed that an interval on which a process is said to proceed may contain intervals on which a process is active or inactive. Furthermore, it has been discussed that any interval on which a process proceed, bounded by intervals on which it does not proceed, can be regarded as a chunk of this process and therefore can be used to represent an event occurrence. Hence, an additional predicate must be used to represent non-primitive event occurrences. In the REGEP framework, a predicate is employed to represent event occurrences in general (including non-primitive ones), which is defined in terms of processes they are made of. From Definition D4.11, it can be seen that the interval on which an event occurs is equivalent to the the maximum interval on which a process is said to proceed.

$$\text{D 4.11} \quad \text{Occurs-On}[a_{th}](e, i) \equiv_{def} \exists v f p b [e = \text{event}(v, f) \wedge p = \text{process}(b, f) \wedge \text{Is-Chunk-Of}(b, v) \wedge \text{Proceeds-On-Max}[a_{th}](p, i)]$$

Finally, the Axiom A4.1 is specified to determine that a primitive token is also regarded as a event token in general where the activeness threshold is zero, meaning that the event is made of a process which is not affected by temporal gaps.

$$\text{A 4.1} \quad \text{Occurs-On-Prim}(e, i) \leftrightarrow \text{Occurs-On}[0](e, i)$$

## 4.10 Identifying Processes in Spatio-temporal Data

It was discussed that the process activeness can be identified from primitive elements grounded upon the data. This section presents a method of defining the predicate  $\text{Active-At}(p, t)$  in such a way that processes can be identified in a temporal series of topographic data. This approach uses the logical apparatus presented in Chapter 3, so that the REGEP framework becomes explicitly linked to the data level.

### 4.10.1 Spatial Changes

Since the aim is to represent processes in terms of spatial changes affecting geographic features, the relation *Spatial Change*  $SC(b, r_b, r_e)$  is employed to capture the intended denotation of a given process classifier. As for geographic features, this relation is defined in terms of primitive elements of the logical framework. However, a distinct definition of this relation should be provided for each process classifier  $p$ , where  $r_1$  and  $r_2$  represent spatial regions of a certain feature's MLP. Hence, these definitions determine how geometric computations should be computed at data level so that a spatial change can be identified within a feature's MLP.

To illustrate how a spatial change can be defined, definitions of the SC relation for 4 different process classifiers: *expanding*, *shrinking*, *extending* and *contracting* (Definitions D4.12 to D4.15) are presented below. These changes are illustrated in Figure 4.4.

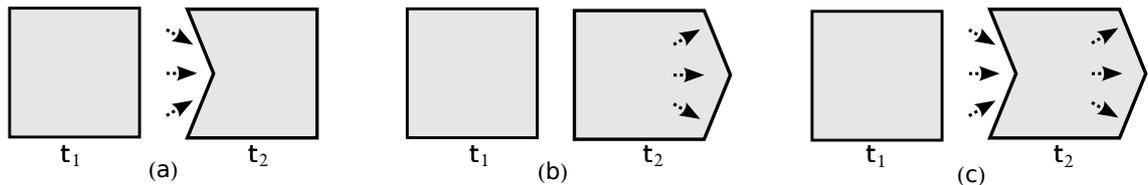


Figure 4.4: Examples of spatial changes. In (a) a feature contracts and shrinks; in (b) a feature extends and expands; in (c) a feature extends and contracts, but neither shrinks or expands

As shown in Definitions D4.12 and D4.13, *expansion* and *shrinkage* can be determined by comparing the area occupied by two given spatial regions.

$$D\ 4.12 \quad SC(\text{expanding}, r_1, r_2) \equiv_{def} \text{area}(r_2) > \text{area}(r_1)$$

$$D\ 4.13 \quad SC(\textit{shrinking}, r_1, r_2) \equiv_{def} \textit{area}(r_2) < \textit{area}(r_1)$$

Whereas *expands* and *shrinks* are defined quantitatively, *extension* and *contraction* can be defined qualitatively, in terms of mereological relationships between two given spatial regions. These spatial changes are specified formally in Definitions 4.14 and 4.15 respectively. In these definitions, PP, PO, EC are the RCC relations *proper-part*, *partially overlaps* and *externally connected*, respectively.

$$D\ 4.14 \quad SC(\textit{extending}, r_1, r_2) \equiv_{def} PP(r_1, r_2) \vee PO(r_1, r_2) \vee EC(r_1, r_2)$$

$$D\ 4.15 \quad SC(\textit{contracting}, r_1, r_2) \equiv_{def} PP(r_2, r_1) \vee PO(r_1, r_2) \vee EC(r_1, r_2)$$

### 4.10.2 Defining Process Activeness

Section 4.8 discussed the assumption held for representing the activeness of a process: if there is a piece of data that describes a change affecting a geographic feature between time instants  $t_1$  and  $t_2$  and nothing else is known between them, then a process characterised by this change is said to be active from  $t_1$  to  $t_2$ . Here such a piece of data corresponds to a feature's MLP, which represents the most detailed information held about changes affecting a geographic feature.

Thus the approach to defining process activeness is now presented. The predicate  $\textit{Active-At}(p, t)$  is defined in a general manner so that it can be applied to any process classifier associated with a spatial change affecting geographic features. In this definition, a process-type associated with a process classifier  $b$  is said to be active at a time instant  $t$  having a geographic feature  $f$  as its participant if  $t$  is between the time instants  $t_b$  and  $t_e$  representing the beginning and end of a certain  $f$ 's MLP, in which  $f$  changes as described by the process classifier.

$$D\ 4.16 \quad \textit{Active-At}(p, t) \equiv_{def} \exists b r_b t_b r_e t_e [ \\ p = \textit{process}(b, f) \wedge \textit{MLP}(f, r_b, t_b, r_e, t_e) \wedge \\ SC(b, r_b, r_e) \wedge (t_b \leq t < t_e)]$$

## 4.11 Summary

This chapter presented a *logical framework* for representing and reasoning about geographic events and processes, encompassing their relationship to participant geographic

features.

The framework comprises an approach based on standpoint semantics to handling spatial and temporal vagueness in order to associate specific spatial and temporal boundaries with event and process instances. Whereas temporal vagueness is considered explicitly in the definitions of Occurs and Proceeds predicates, spatial vagueness is handled for representing geographic features (Chapter 3), whose spatial boundaries correspond to those of the event and process tokens they participate in. Furthermore, it could be seen the REGEP framework can be adapted for representing and reasoning about many different geographic phenomena, and is flexible to enable reasoning based on distinct standpoints.

The next chapter presents a *system prototype* which implements the theoretical framework presented in this chapter and the STAR model described in Chapter 3.

# Chapter 5

## System Implementation

---

### 5.1 Introduction

This chapter describes a system prototype which implements the theory presented in Chapters 3 and 4. This is structured as follows. Section 5.2 gives an overview of the system prototype. This is followed by a discussion in Section 5.3 on the approach to managing data within the system. Then Sections 5.4 and 5.5 describe the approaches to implementing spatial and temporal reasoners, respectively. Then Section 5.6 discusses the implementation of the STAR model presented in Chapter 3. Following this, Section 5.8 describes the implementation of the logical framework presented in Chapter 4. Finally, Section 5.9 summarises the Chapter.

### 5.2 The System Prototype

A *system prototype* that can process real geographic data has been implemented to evaluate the applicability of the theory proposed in this thesis. This system prototype has been named Progress. The system takes temporal series of topographic data as an input and allows logical queries to be formulated about the data, returning textual and graphical information on events, processes, and the geographic features which participate in them.

The system prototype's main screen is shown in Figure 5.1. In this figure, the smaller window shown on the front is a command-line terminal used to formulate user queries and

to visualise textual results. The window on the back contains an interactive map for visualising spatio-temporal results in a graphical way. Beyond the mechanism for formulating and answering logical queries, the system provides several standard GIS functionalities which can improve the visualisation and analysis of results in many different forms. For example, query results can be overlaid on other thematic map layers.

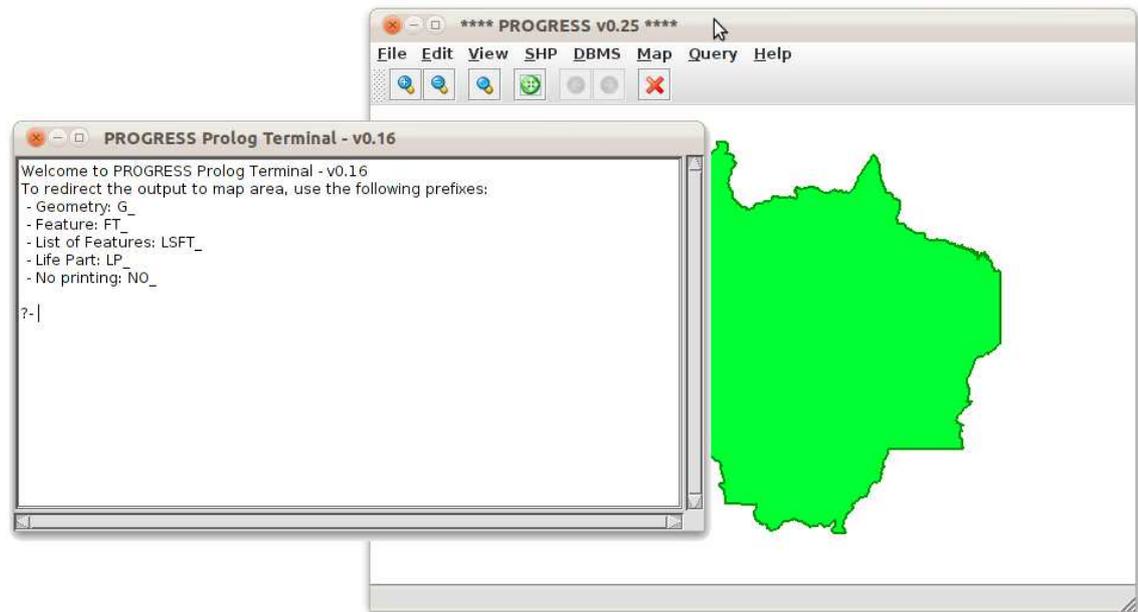


Figure 5.1: System prototype's main screen.

## 5.2.1 The System Architecture

The *system architecture* is shown in Figure 5.2. The system is structured in three main layers, named *data*, *processing* and *visualisation* layers. The *data layer* comprises the KB and the deductive mechanism described in Chapter 3. Facts are stored in a spatially enabled Database Management System (DBMS), whilst the deductive mechanism is implemented in the Prolog programming language. The DBMS used was PostgreSQL ([www.postgresql.org](http://www.postgresql.org)). PostgreSQL is a free, robust and well documented DBMS which contains a specialised module for managing spatial information (PostGIS, [www.postgis.net](http://www.postgis.net)). The Prolog implementation used for developing the prototype was SWI-Prolog ([www.swi-prolog.org](http://www.swi-prolog.org)). This is a free Prolog implementation which is bundled with interface libraries for Java and for ODBC (the Microsoft standard for Open Data Base Connectivity).

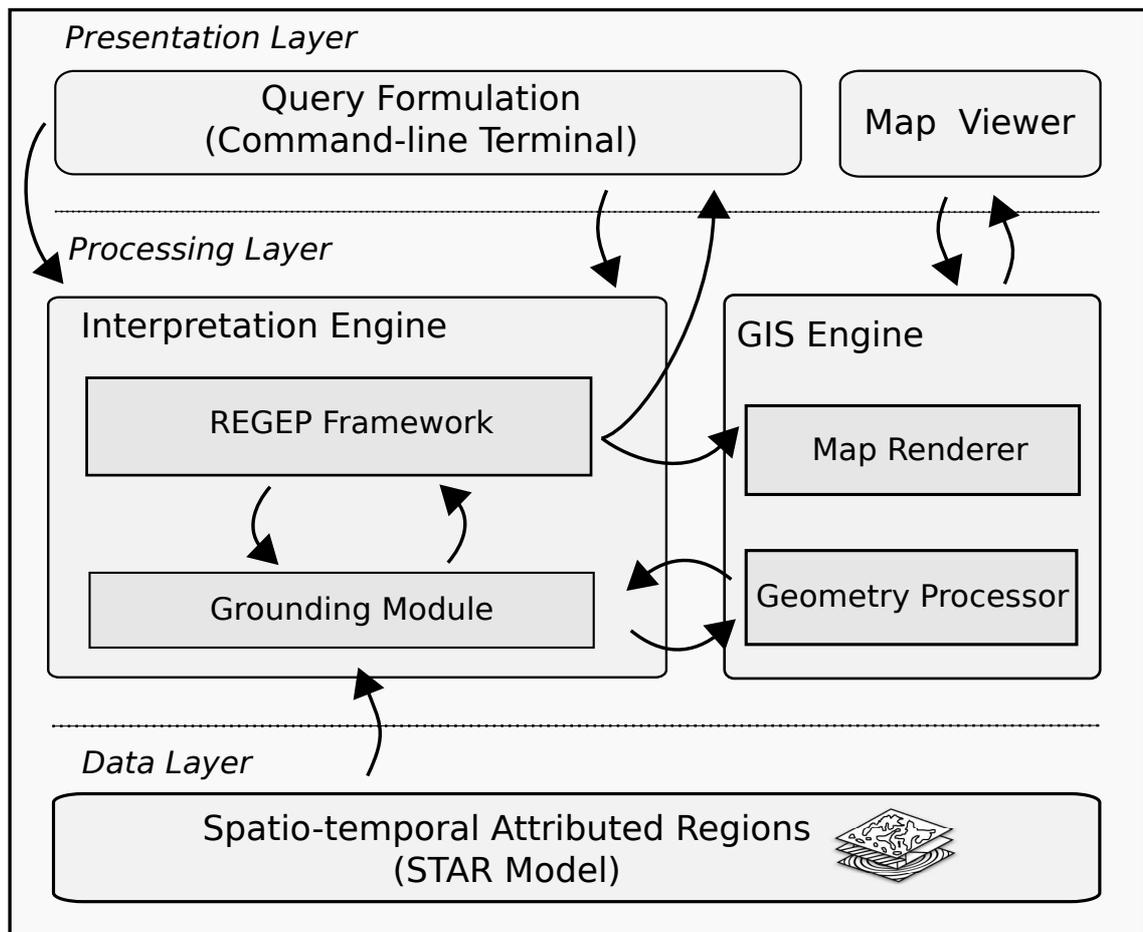


Figure 5.2: Architecture of the system prototype.

The *processing layer* comprises the *Interpretation* and *GIS Engines*. The former, also implemented in SWI-Prolog, includes the REGEP logical framework presented in Chapter 4, which processes the user queries; and a mechanism to ground the framework upon the data. The *GIS Engine* is implemented in Java programming language and uses built-in spatial functions provided by the DBMS. This module provides standard GIS functionalities, processes geometric computations required by the grounding module and renders maps (which may contain spatial information representing the result of a user query). The implementation of this module makes use of GeoTools ([www.geotools.org](http://www.geotools.org)), a Java code library for manipulating geospatial data.

The *visualisation layer* provides mechanisms for formulating user queries and visualising their results. The user interaction can be performed both textually (via a Prolog-like command-line terminal) and graphically (via an interactive map). Further details of the mechanisms for formulating and handling queries, as well as of the mechanism for gen-

erating and outputting results are described later in this chapter. Results obtained from experiments using the system with real-world topographic data are described in Chapter 6.

## 5.2.2 Logical Language Choice

As described in Section 5.2.1, Prolog has been chosen as the programming language to implement both the STAR approach presented in Chapter 3 and the logical framework presented in Chapter 4.

Prolog is a programming language that originates from formal logic, and is broadly used today for building applications which carry out tasks by making use of artificial intelligence techniques. The factor that primarily distinguishes Prolog from other popular programming languages is that it is based on declarative programming, whilst others are usually based on procedural programming. This declarative approach provides ways of representing the knowledge using a rule structure which is adequate for implementing the theory proposed in this thesis. Moreover, Prolog has logic programming as its backbone and is based on proof theory, so that it enables a computer program to draw conclusions by means of logical inference.

Although Prolog is considered a declarative programming language, it also allows some imperative programming elements to be implemented, which is particularly useful for writing programs that process numeric data, such as applications which manipulates spatial and temporal information. In addition, as a general purpose language, Prolog provides a number of built-in predicates to perform common tasks, such as input and output.

The most popular implementations of Prolog are SWI-Prolog and SICStus Prolog. The former is a free and open source, whilst the latter is a commercial implementation. Both implementations provide a variety of built-in functions for manipulating many different data types and offer interfaces for connecting to other programming languages (e.g., Java, C) and to Database Management Systems. The facts that SWI-Prolog is a free implementation, is well maintained and extensively documented has contributed to choose this to develop Progress. However, revising the system code to ensure that it works in both implementations would be fairly straightforward.

## 5.2.3 Query Handling

The central component of Progress' architecture is an interpretation mechanism which implements the theory presented in Chapters 3 and 4. This component is implemented as

a Prolog-style theorem-proving system, with the aim of answering queries about changing extensions of geographic features by means of logical inferences. Prolog provides logical reasoning via Horn clauses, and offers a mechanism for handling logical queries by attempting to find resolution refutation of the negation of the query. That is, it denies the alleged conclusion and then proves that this is inconsistent with the initial statements.

A Prolog program is executed as the user specifies a certain goal, called a query. By submitting a query to the Prolog interpreter, one asks whether it can prove that the query is true. If so, it answers ‘true’ and displays any variable bindings that it made in coming up with the solution. That is, if the negated query is refuted, this means that the query is a logical consequence of the program (with the appropriate variable bindings in place). On the other hand, if it fails to prove the query true (i.e. if the negated query cannot be refuted), it answers ‘false’.

Prolog’s interpretation mechanism assumes that its database contains complete knowledge of the domain it is reasoning about. This is known as the *Closed World Assumption* – that is, the Prolog interpreter concludes something is false if it cannot prove it is true given the facts and rules in its KB. This is also known as *negation as failure*.

The following query asks Prolog whether it can prove an event of type  $E$  occurs on a time interval  $I$ .<sup>1</sup>

```
?- occurs_prim(on,E,I).
```

Notice that, differently from first-order logic, Prolog recognises ordinary words beginning with a lower case letter as constants, whereas variables are represented by words beginning with a capital letter. The query above contains two uninstantiated variables  $E$  and  $I$ , and therefore Prolog will try to find values that match those consistently. Given the closed-world assumption, if Prolog cannot find any values from the existing KB that makes the statement true, then it fails.

If it can find from the existing KB any values that match the goal `occurs_prim(on,E,I)`, it displays the values for the first solution found, as for ex-

---

<sup>1</sup>The constant ‘on’ shown in the query is required due to a code reuse technique adopted. The use of this kind of constant will be discussed in Section 5.8.

ample:<sup>2</sup>

```
?- occurs_prim(on,E,I).
E = event(expands, feature(...)),
I = ['2004-07-01 00:00:00', '2004-11-01 00:00:00'] .
```

If the user press the ‘;’ key (semi-colon), then Prolog will try to find the next solution for the query, that is, another set of values bound to the variables which made the statement true. For example, this might be another interval on which an event of the same type occurs. This command can be repeated until Prolog finds all possible solution. Alternatively, at any time the user can press the ‘.’ key (period) to abort the query processing.

### 5.2.3.1 Argument Passing and Control

The unification algorithm of Prolog inference engine generates the most general unifier between two formulae. This provides a single mechanism for:

- passing parameters into and out of functions and predicates;
- constructing and accessing compound terms;
- comparing and assigning variables [26].

Prolog queries may contain free variables or not. When a query contains *no free variables*, the output of results is not of interest. Rather, it is of interest whether the proposition represented by the query holds within the model. An example query would be to confirm that an event of a given type occurred on a specified interval. Thus the expected result for a query of this kind consists of a simple *true* or *false* response. On the other hand, when a query contains *one or more free variables*, the objective is to find and display *variable bindings* that the system made in coming up with the solution for the proposition represented by the query. In Progress, variable values may be visualised either textually or graphically (additional details about the mechanism for outputting and visualising results will be described in Sections 5.2.3.3 and 5.2.4).

An important characteristic of Prolog is the flexibility in the manner arguments of a predicate can be used. That is, a predicate can be implemented in such a way that any argument can be queried and consequently the reasoning can be performed in different directions. This is known as the ‘reversibility’ of Prolog programs. An example is the

---

<sup>2</sup>For short, in this example and in other examples in this Section, some details of the representation of geographic features were omitted (were replaced by ‘...’). The method of representing geographic features in prolog is discussed in Section 5.6.

form in which the built-in predicate `append(X, Y, L)` can be used. This predicate defines the relation between three lists such that the third is the concatenation of the first two. With this predicate, it is possible to construct the list `L` (given as a free variable) from lists `X` and `Y` (given as constants):

```
?- append([re,mi], [fa, sol, la], L).
L = [re, mi, fa, sol, la] .
```

Alternatively, one might try to prove the relation with the first argument free and the other two constant. In this case, the system returns the list `X` that should be concatenated to the second to get the third. This is illustrated below.

```
?- append(X, [fa, sol, la], [re, mi, fa, sol, la]).
X = [re, mi] .
```

Similarly, the predicate `occurs_prim(on, E, I)` of the example above can be used in a query with just `I` or with no free variables:

```
?- occurs_prim(on, event(expands, feature(...)),
  ['2004-07-01 00:00:00', '2004-11-01 00:00:00']).
true.
```

For readability, the same query can also be written using instantiated variables:

```
?- E = event(expands, feature(...)),
  I = ['2004-07-01 00:00:00', '2004-11-01 00:00:00'],
  occurs_prim(on, E, I).
true.
```

There can be more than one goal in a query. In this case, they are said to be different subgoals of the query. Subgoals must be separated by commas and must be terminated with a period. Prolog will try to find a solution to the first subgoal and continue to the next subgoal only after the first subgoal is reached. For example, the following query checks whether an event of a given type occurs on a time interval before a specified interval.

```
?- occurs_prim(on, event(expands, feature(...)), I),
  before(I, ['2010-01-01 00:00:00', '2010-01-01 00:00:00']).
```

### 5.2.3.2 Searching for Solutions

In responding to queries, the Prolog interpreter uses a backtracking search process, as follows:

- If a goal matches with a fact, then it is satisfied.
- If a goal matches the head of a rule, then it is satisfied if all subgoals of the rules body are satisfied.
- When a subgoal fails, Prolog traces its steps backwards to the previous goal and tries to satisfy it again. In doing that all variables that were bound to a value when that goal was satisfied are now made free again. Then the Prolog tries to satisfy that goal again by binding its variables to different values.
- This continues until either the subgoal is satisfied or the KB has been exhausted. If the latter occurs, Prolog backtracks yet again to the subgoal just before the current subgoal.

Three Prolog's built-in predicates are of particular interest in queries formulated within Progress: `findall`, `bagof` and `setof`. These predicates are employed to retain multiple values bound to a certain variable for coming up with different solutions to a given goal. These values are unified with a specified variable of type 'list'. In these predicates the existential quantifier is represented by the infix operator '^'.

For example, the goal `bagof(A,G^S^star(A,G,S),Bag)` asks for the 'Bag' of A's such that there exists a G and there exists an S such that `star(A,G,S)`. Since these predicates take a goal as an argument, they are also called *metapredicates*. While the solution list output by `findall` and `bagof` may contain repeated values, `setof` removes all duplicate items. `findall` is equivalent to `bagof` with all free variables automatically bound with the existential operator '^', and succeeds with an empty list if no solution is found to a given goal. On the other hand, `bagof` and `setof` fail in that situation.

### 5.2.3.3 Progress' Command-line Terminal

The command line terminal was implemented in Java and includes the usual functionalities of a Prolog terminal. However, this terminal is connected to both the GIS and Interpretation Engines, so that results of queries processed by the latter can be visualised either textually or on the map pane. This terminal distinguishes from other Prolog terminals by the fact it provides to the user additional forms of controlling the way values

bound to variables are processed by the system. That is, variables can be named by using several special prefixes, which are used to inform the system these variables should be processed in a different manner. The following prefixes are used:

- *G\_*, which informs the system that the values expected to be assigned to the variable correspond to geometries. Thus, whenever a value is assigned to the variable named using this prefix, the graphical representation for the geometric value is shown on the map pane and its textual representation is suppressed from the terminal.
- *FT\_*, used to indicate that values assigned to the variable correspond to geographic features. Therefore, values assigned to a variable named with this prefix are displayed on the map pane and their textual form are not shown on the terminal.
- *LFT\_*, that indicates that the value held by the variable corresponds to a list of geographic features. These values are therefore shown on the map pane and hidden from the terminal.
- *LP\_*, used to inform the system that a variable will hold values corresponding to a feature life part. Therefore, these values are displayed on the map pane and are not shown textually.
- *NO\_*, which informs the system the values assigned to a variable should be displayed neither in the terminal nor in the map pane.

As will any usual prolog terminal, Progress' prolog terminal displays the solutions for a given query, by default, as a list of variables specified in the query and the respective values assigned to them. However, displaying values of geometric elements in a textual form is not usually helpful for users' analysis. Even if geometries are displayed using Well-Known Text (WKT)<sup>3</sup>, their representation often consists of a long list of coordinates which is difficult for a human to comprehend. Hence, by naming a variable using prefixes *G\_*, *FT\_*, *LT\_* or *LP\_*, the user instructs the system to hide the value from the terminal and to display it on the map pane, allowing them to visually analyse the solutions.

Furthermore, there are cases where the values assigned to a variable are not relevant to users, but where the variable is required to formulate a query. In these cases, one could just ignore the values shown at the terminal; alternatively, the user can order the system to hide these variables from the output, by naming them using the prefix *NO\_*. This is particularly important for hiding unwanted geometric variables containing long text strings.

---

<sup>3</sup>Well-known text (WKT), [http://en.wikipedia.org/wiki/Well-known\\_text](http://en.wikipedia.org/wiki/Well-known_text)

## 5.2.4 Result Visualisation

Progress provides several visualisation tools to support the analysis of results of queries formulated using the prolog terminal described in Section 5.2.3.3. A number of results evaluated by using these tools are discussed in Chapter 6.

As described in Section 5.2.3.3, it is possible to display on the map pane values of variables denoting either geographic features or features' life parts (LPs). The visualisation mechanism are the same for both situations, that is, when the user requests to display a geographic feature, the systems actually displays a LP which corresponds to the whole life of the feature.

Two visualisation modes are provided for LPs. The first, named *hulls*, aims to provide more syntactic information about the portion of space occupied by the feature throughout the LP. In this mode, just *one geometry* is displayed, which corresponds to the concave hull comprising *all regions* occupied by the feature during the LP. This visualisation mode is useful when the spatial information at every time instant within the LP is not essential for performing the intended visual analyses. For example, this mode can be used to visualise the result of a query for retrieving event occurrences, so that the whole region affected by each individual event can be easily identified. These hulls are displayed using translucent polygons.

The second mode, named *navigation*, provides more analytical information about the space occupied by the feature. In this mode, two buttons are provided to the user for navigating along the LP, so that they can visualise each known spatial region occupied by the feature at different time instants throughout the LP. This mode is useful when the way a feature evolves during a given interval is relevant for the intended visual analyses. In this mode, these different regions are displayed using opaque polygons

The user can switch between those visualisation modes at any time after the results are displayed. Moreover, when the analytical mode is active, the syntactic information is also displayed on the background for reference. Each piece of spatial information redirected from the terminal to the map pane is represented as a different map layer, so that they can be shown or hidden when convenient. Figure 5.3 shows an example of a query result visualised using mode 'navigation'. In this example, regions occupied by a geographic features at different time are shown using opaque polygons; whereas the whole region occupied by the feature throughout its life is shown using translucent polygons.

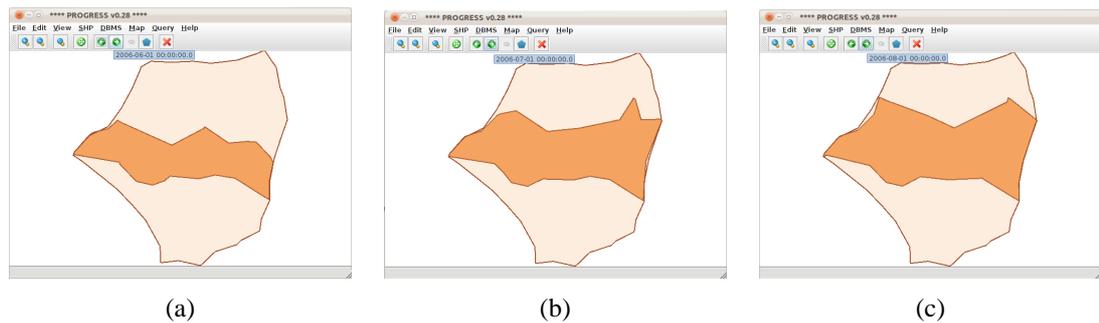


Figure 5.3: Example of results visualised using mode ‘navigation’, where regions occupied by a geographic features at different time are shown using opaque polygons; and the whole region occupied by the feature throughout its life is shown using translucent polygons.

### 5.3 Building a KB upon a Relational Database

The importance of DBMSs for applications developed in Prolog is often underestimated, since Prolog itself can manage significant amounts of data. Nonetheless using a DBMSs within these applications can provide numerous advantages [88]. Advantages of using DBMS include their support for transaction management and their improved mechanism for integrity and consistency check. Of particular interest in using a DBMS for developing Progress was the use of its facilities for storing, manipulating and accessing spatial information.

In Progress, modules developed in Prolog are connected to the DBMS via the SWI-Prolog ODBC Interface. ODBC (Open Database Connectivity) is a middleware API for accessing DBMSs, which is independent of database systems and operating systems, so that an application written using ODBC can be easily ported to other platforms. An important feature of ODBC explored within Progress is its support for parameterised execution of SQL strings. Whilst direct execution of literal SQL strings is a simple practical solution for infrequent calls, parameterised execution allows the database to pre-compile the SQL query and store the optimised code, making it appropriate for time-critical operations.

When processing a SQL ‘SELECT’ statement via a SWI-Prolog ODBC Interface, rows are returned one-by-one on backtracking as terms of a specific functor. The library pre-fetches the next value to be able to close the statement and return deterministic success when returning the last row of the result-set. In addition, this interface provides support for retrieving multiple rows via the `findall` predicate, where the result-set is returned as a list of user-specified terms.

Below is an example of a Prolog code which uses a ODBC connection to a DBMS (i.e.,

`pgconnection`) to retrieve *Star* facts explicitly asserted in the KB (i.e.,  $A\text{-Star}(a, g, s)$ ), as described in Chapter 3). This example describes a simplified version of the actual code used specified the system.

```
astar(A,G,S) :- odbc_query(pgconnection,
                        'SELECT attribute, geometry, timestamp FROM star',
                        row(A,G,T)).
```

Facts of the Progress' KB are stored within in a relational database. The database contains tables for storing *Stars*, attributes, CP relations, MP relations, features and feature lives, each of those corresponding to a specific table. To improve efficiency, records stored in these tables are indexed by a numeric identifier (ID). These IDs are also used to cross-reference elements between those tables. In addition, records stored in table 'star' are indexed by the column 'geometry', using specialised indexing structures for geometric data types provided by the DBMS.

## 5.4 Implementing the Spatial Reasoner

As discussed before, the Region Connection Calculus (RCC) has been employed as the theory of space for developing the formalism presented in this thesis. Moreover, as noted in Section 5.2.1, Progress prototype uses PostgreSQL/PostGIS to store and manage spatial data. Therefore, RCC has been implemented within Progress based on the spatial relations provided by PostGIS<sup>4</sup>.

On the one hand, RCC comprises a set of qualitative spatial relations based on the concept of 'connection' between regions (described in Section 2.2). On the other hand, PostGIS provides many different spatial relations, both qualitative and quantitative. Although some of their relations have similar or identical names, there is no direct mapping between them. For example, the RCC relation *overlaps*  $O(r_1, r_2)$  holds between two spatial regions if they share any common region. Divergently, the *overlaps* relation provided by PostGIS holds for two geometries if they share space but are not completely contained by each other. The latter, in turn, corresponds to the RCC relation *partially overlaps*.

Therefore, in Progress, each RCC relation is defined in terms of a conjunctions or disjunctions of PostGIS relations. For instance,  $O(g_1, g_2)$  will hold if  $ST\_Contains(g_1, g_2)$  or  $ST\_Contains(g_2, g_1)$  or  $ST\_Overlaps(g_1, g_2)$  holds, where  $ST\_Contains$  are  $ST\_Overlaps$  are spatial relations provided by PostGIS.

---

<sup>4</sup>PostGIS 2.0 Manual, Section 8.8 (Spatial Relationships and Measurements), [http://postgis.net/docs/reference.html#Spatial\\_Relationships\\_Measurements](http://postgis.net/docs/reference.html#Spatial_Relationships_Measurements)

In the current version of Progress, spatial regions are represented at the data level using PostgreSQL Polygon geometric type<sup>5</sup>. Geometries are stored and internally manipulated using the Well-known Binary (WKB)<sup>6</sup> format. Whereas the Well-known Text (WKT)<sup>7</sup> format is used for friendly visualisation of data and for explicitly asserting facts in the KB containing spatial information. An example of a polygon represented in WKT is ‘POLYGON((60 95, 95 95, 95 110, 60 110, 60 95))’.

## 5.5 Implementing the Temporal Reasoner

The temporal reasoner implemented in Progress is an extension of the reasoner developed as part of the *Context Broker Architecture* (CoBrA)<sup>8</sup>. CoBrA’s temporal reasoner consists of two main components: the *Base Temporal Reasoner*, comprising rules for processing ISO 8601 date and time descriptions, following the standards described in [14]; and the *Extended Temporal Reasoner*, that provide support for more sophisticated temporal reasoning, such as the *Interval Calculus*. The latter is based on the ontology and axioms described by Hobbs et al. [50].

The reasoner implementation is based on two temporal entities, namely, time instants and time intervals. These entities are represented as Prolog lists. A list representing an instant should contain a single element; whereas a list representing an interval should contain two elements, corresponding to the instants that begin and end the interval (in this order). Elements in a list are assumed to be single-quoted date-time descriptions in the ISO 8601 format. An example of a time interval which follows this structure is [‘2006-10-11 01:30:00’, ‘2013-05-15 12:00:00’].

## 5.6 Implementing the STAR Model

Chapter 3 presented a logic-based approach to representing geographic spatio-temporal data, which allows inferences to be made for deriving implicit Spatio-temporal Attributed Regions from other explicitly asserted in the KB. This section describes relevant aspects relating to the implementation of this approach.

---

<sup>5</sup>PostgreSQL Geometric Types, <http://www.postgresql.org/docs/8.2/static/datatype-geometric.html>

<sup>6</sup>Well-known Binary (WKB), [http://en.wikipedia.org/wiki/Well-known\\_binary](http://en.wikipedia.org/wiki/Well-known_binary)

<sup>7</sup>Well-known Text (WKT), [http://en.wikipedia.org/wiki/Well-known\\_text](http://en.wikipedia.org/wiki/Well-known_text)

<sup>8</sup>Context Broker Architecture (CoBrA), <http://cobra.umbc.edu/about.html>

### 5.6.1 Implementing Axioms for Integrity Checking

As noted in Chapter 3, the STAR model comprises two main sets of axioms, one for determining integrity constraints amongst *Star* facts and another which comprises rules for deriving implicit *Stars*. These axioms have been implemented using Prolog, and therefore are based on Horn Clauses. For this reason, integrity axioms are specified based on a special predicate `axiom(Name, Formula)`, where `Name` is an identifier for a given axiom and `Formula` is a formula that contradicts that axiom. Hence, the KB satisfies an integrity axiom referred to by `Name` if the proposition `check_axiom(Name)` is true, where the predicate `check_axiom` is implemented as follows.

```
check_axiom(Name) :- axiom(Name, Formula),
                    \+check_formula(Formula).
```

In `check_axiom`, the predicate `check_formula` is used to search for some violation of the formula `Formula`. Therefore the negation as failure succeeds if no fact inconsistent with the formula is found.

Below is an example of a formula implemented to check Axiom 3.10, described in Chapter 3. In this Prolog code, it can be seen the connective `and` is represented as a function symbol. Its logical meaning is captured by the implementation of the formula checking mechanism. This connective has been implemented to help specify conjunctions of sub-formulae within a given formula. Other similar connectives (e.g., `or`) have also been implemented to help specify formulae.

```
axiom(axiom_cp_catt_fatt_cardinality_2,
      and([
          fatt(AF), cc(AF, AC), cc(AF, AC1), AC\==AC1
      ])).
```

The predicate `check_multi_axoms(L)` has been implemented to check whether the KB satisfies a list `L` of axioms (where each element of `L` is an axiom identifier). Thus the truth of a proposition using this predicate, with `L` referring to all integrity axioms, is verified for *Star* facts explicitly asserted in the KB. This predicate is implemented as follows.

```
check_multi_axoms(L) :- loop_multi_axioms(L).
loop_multi_axioms([]).
loop_multi_axioms([Hd|Tl]) :- check_axiom(Hd),
                              loop_multi_axioms(Tl).
```

To gain performance efficiency, integrity axioms are not verified for each new fact asserted in the KB. Rather, the KB is often updated using batches of new facts, and therefore

`check_multi_axoms(L)` is checked only after each batch update.

## 5.6.2 Implementing Derivation Rules

Derivation rules (presented in Section 3.5.2) are implemented in the system in usual form `head :- body`, where the head always consists of the predicate `star/3` and the body consists of calls to predicates representing the rule's goals<sup>9</sup>. These goals must include at least one call to predicates `star/3` or `astar/3`. In addition, it should be noted that `astar/3` must not appear in the head of a rule, since this predicate denotes a *Star* that is explicitly asserted in the KB, whereas the head of a rule denotes the element derived by the rule. An example describing a simplified version of the Prolog code used to retrieve explicitly asserted *Stars* from the KB is given in Section 5.3. Sections 5.6.3 to 5.6.6 describe relevant aspects of the implementation of derivation rules.

## 5.6.3 Explicitly Asserting Derived Facts

There are certain cases in which deriving implicit facts from other existing ones requires performing costly spatial calculations. Therefore these inferences are not suitable to be made at runtime (i.e., query time). For this reason, Progress includes mechanisms for explicitly asserting certain derived facts in the KB, allowing them to be quickly recovered at runtime.

In the current version of the STAR model implementation within Progress, the following elements are explicitly asserted in the KB after being inferred from other existing facts.

- *Star* facts describing spatial extensions of geographic features at particular time instants. These facts are produced by performing Derivation Rules DR2 or DR6, and then DR4 (presented in Chapter 3).
- *Star* facts produced by the Derivation Rule DR3 (presented in Chapter 3), which deals with the principle of inertia.
- Facts describing geographic features and their lives.

Beyond the fact that these elements require significant time to be inferred, they have been chosen to be explicitly asserted in the KB because they can be finitely produced by

---

<sup>9</sup>Predicates specified in Prolog are described in the form `predicate_name/arity`. Therefore, `star/3` refers to the predicate `star(A,G,S)`, whose arity is 3. Variables A, G, and S denote, respectively, attributes, geometries and timestamps.

the derivation rules. Contrastingly, derivation rules which can produce an infinite number of elements are only used in the system to check whether a given hypothesis is true. However, inferences performed by the latter usually requires a small amount of time to be performed.

For instance, since a certain region may contain an infinite number of sub-regions, DR1 is only used in the system to evaluate propositions in which the variable representing the sub-region is instantiated. To illustrate, suppose the proposition `star('hot', 'POLYGON((6 6, 9 6, 9 9, 6 9, 6 6))', ['2012-01-01T00:00:00'])` (where 'hot' is a homogeneous coverage attribute) is given to the system and suppose it can derive `star('hot', 'POLYGON((2 2, 11 2, 11 9, 2 9, 2 2))', ['2012-01-01T00:00:00'])`, such that the first polygon is part of the second. Thus, once the latter is derived, the system can prove the given proposition is true considerably quickly by a single execution of DR1. Nevertheless, deriving the latter might require significant amount of time. Hence, if the latter is explicitly asserted in the KB, the whole proof could be quickly completed.

Therefore, by explicitly asserting derived *Stars* representing extensions of geographic features at particular time instants, the system can efficiently derive *Stars* via DR5. Then these implicit *Stars* can be used to derive other *Stars* via DR1 (also quite efficiently). The reasoning performed by the system is similar for facts describing extensions of simple or compound features. However, facts describing homogeneous regions used to derive extensions of simple features can be removed from the KB after asserting derived facts explicitly. This is possible because the originating facts can be derived back from the new fact. However, the same does not applies to compound features, since it is not possible to infer (from a single fact representing the extension of the feature) the exact extensions of their constituent regions with different types of coverages.

Given the fact that derived *Stars* representing spatial extensions of geographic features are explicitly asserted in the KB, it should be noted that, according to Axiom 3.18, when the inference is made from the left-hand side to the right-hand side (DR5), the *Star* fact which appears in the body of the rule can also be considered as expressed in the form  $A\text{-Star}(a, g, s)$ , meaning that the fact is explicitly asserted in the KB.

According to the axioms presented in Chapter 3, it can be noticed that DR3 could derive an infinite number of *Stars* from an existing *Star*, since there are an infinite number of time instants within any given interval. However, in Progress, this derivation rule is also used to derive *Stars* which will be explicitly asserted in the KB. But the system only derives *Stars* for timestamps already associated with other existing facts. That is, for each existing timestamp in the initial KB, the system applies DR3 to derive new elements by

instantiating the timestamp variable. Once the elements derived by DR3 are explicitly asserted in the KB, the system only uses this rule for evaluating propositions where the timestamp variable is instantiated. Therefore, since the inference mechanism has to consider only the last known timestamp before the specified one, these inferences are rapidly performed.

Furthermore, elements derived by DR3 should be asserted in the KB *before* executing the mechanism for deriving (and then explicitly asserting) spatial extensions of geographic features. To illustrate this need, suppose a *Star* associated with attribute  $a$  and timestamp  $s_2$  is derived via DR3 from an existing *Star* associated with the timestamp  $s_1$  (where  $s_1 < s_2$ ) and with the same attribute  $a$ . Then suppose the initial KB contains a *Star* fact associated with  $s_2$  whose geometry is externally connected to the geometry associated with the derived *Star*. Therefore, it is clear that these geometries should be considered in conjunction for deriving a *Star* representing the maximal extension of regions (at  $s_2$ ) whose coverages are both denoted by attribute  $a$ .

Another important aspect of the STAR model implementation is that, in some situations, derived facts which were explicitly asserted in the KB have to be deleted in case additional data are to be input to the system. This is needed when new facts include *Stars* associated with timestamps within the range of timestamps already known by the system (i.e., new timestamps might either match an existing one or be between other two already presented in the KB). This is specially important due the principle of inertia that is considered here. That is, a new fact may invalidate the assumption that something has not changed between two time instants.

#### 5.6.4 Implementing the Principle of Inertia

There is a slight difference in the way the principle of inertia is considered in pure logic and the way it is implemented within the system. According to Derivation Rule DR3 (Chapter 3), if a spatial region  $r$  covered by  $c$  at  $t_1$  is *partially clipped* between time instants  $t_1$  and  $t_2$ , then the coverage  $c$  will hold at  $t_2$  for every sub-region of  $r$  which have not been not clipped in that period. However, it should be noticed that axioms relating to DR3 do not specify a method of determining precise spatial extensions of sub-regions of  $r$  whose coverage persist at  $t_2$ . That is, Axiom 3.17 assumes the appropriate non-clipped sub-regions will be correctly derived by DR1.

On the other hand, as noted in Section 5.6.3, Progress does not apply DR1 for producing implicit regions. Consequently, in the example above, the system would not be able to infer a region that would make true the proposition existentially quantified on the

left-hand side of Axiom 3.17. This issue is solved in the system implementation by including in the derivation rule a method of calculating the spatial difference between the region  $r$  and the regions that have been clipped between  $t_1$  and  $t_2$ . The result of this calculation corresponds to the maximal sub-region of  $r$  that has not been clipped (i.e., which remains covered by  $c$  at  $t_2$ ). Therefore it represents the inferred region that will be explicitly asserted at  $t_2$ .

### 5.6.5 Calculating Maximal Extensions of Regions

In the theory presented in Chapter 3, maximal extensions of regions of a particular coverage are determined by performing DR2 or DR6 successively until the derived region can be applied to DR4 to derive the final result. Although the reasoning performed by DR2 or DR6 suits the problem from the conceptual point of view, it can be noticed that successive derivations by DR2 or DR6 may be computationally inefficient. This might occur due to the fact that these derivation rules consider only a pair of regions at a time to produce new regions that best represent the sum of the other two (which requires performing costly spatial calculations, such as for obtaining spatial unions or concave-hulls).

For that reason, these rules are implemented in Progress in a slightly different manner than that seen in Axioms 3.15 and 3.19. The main difference is that, in the system, the geometric representation of the region derived by DR2 or DR6 is not calculated for each call to the Prolog predicates corresponding to their implementations. In the system, when the body of the formulae is satisfied (in Axioms 3.15 or 3.19), the two regions considered in that are just ‘labelled’ as part of the final result. Then, for each region already labelled, the system successfully calls those predicates by specifying the region as a constant (i.e., an instantiated variable) and leaving the other variable free. Then regions bound to that free variable are successively labelled as part of the result. Eventually this process is exhausted and then the geometric representation for the aggregate (e.g., a concave-hull) is calculated by taking into account all labelled regions.

### 5.6.6 Calculating Spatial Aggregates and Concave Hulls

Section 5.6.5 described the implementation approach to determining maximal extensions of regions of a particular coverage. As discussed in Section 3.5.3, such maximal regions may be characterised by the formation of aggregates of regions which are in fact separated from each other.

In Progress, the predicate  $\text{Star}[d](a, g, s)$  (presented in Section 3.5.3) takes the form  $\text{star}(D, A, G, S)$ . Distance measurements for calculating aggregates are based on the

PostGIS function  $ST\_DWithin(g1, g2, d)$ , which returns true if the geometries  $g1$  and  $g2$  are within the specified distance  $d$  of one another<sup>10</sup>.

As specified in Axiom 3.19, the spatial extension of an aggregate is determined by a concave hull which encloses all regions that belong to an aggregate. The algorithm for calculating concave hulls used within Progress is that implemented in PostGIS function  $ST\_ConcaveHull$ <sup>11</sup>. The output of this algorithm depends on a numeric parameter (ranging from 0 to 1) which specifies the target percent of area of convex hull the solution will try to approach before aborting the execution of the algorithm. Moreover, choosing the appropriate value for this parameter involves a trade-off between accuracy and efficiency. Many different algorithms for calculating concave hulls are currently available, and the algorithm used within Progress can be changed straightforwardly.

Experiments<sup>12</sup> using the system with a real spatio-temporal dataset have shown that there is a substantial increase in the execution time when the percent target is reduced by just 0.01. For example, experiments have been conducted involving 260,762 polygons for generating 3,961 concave-hulls. Then it has been realised that executing these experiments using a typical desktop computer is impractical for values of target percent lower than 0.95. In addition, it has been observed that the difference in accuracy is not significant when this value varies between 0.95 and 0.99. Hence, the current version of Progress establishes a constant value of 0.99 for the target percent parameter.

However, there are certain cases (although considerably infrequent) in which this algorithm provides unsuitable outputs, leading the system to perform inadequate conclusions regarding changes affecting the extension of features over time. An illustrative scenario describing this problem is presented in Figure 5.4. This figure shows the concave-hull generated for a set of regions (corresponding to an aggregate) for time instants  $t_1$  and  $t_2$  (where  $t_1 < t_2$ ). In this figure, it can be noticed that the aggregate at  $t_2$  contains one additional element, but whose presence should not affect the concave hull established at  $t_1$ . That is, if these concave-hull represented extensions of a particular feature, the feature should be regarded as unchanged from  $t_1$  to  $t_2$ . Nevertheless, as seen in the figure, the presence of this new element caused the concave-hull generated for  $t_2$  to be smaller than that of  $t_1$ , leading to an improper conclusion that the region has shrunk in that period.

<sup>10</sup>[postgis.refractor.net/documentation/manual-svn/ST\\_DWithin.html](http://postgis.refractor.net/documentation/manual-svn/ST_DWithin.html)

<sup>11</sup>[postgis.org/docs/ST\\_ConcaveHull.html](http://postgis.org/docs/ST_ConcaveHull.html)

<sup>12</sup>Results of using Progress with temporal series of topographic data will be presented in Chapter 6.

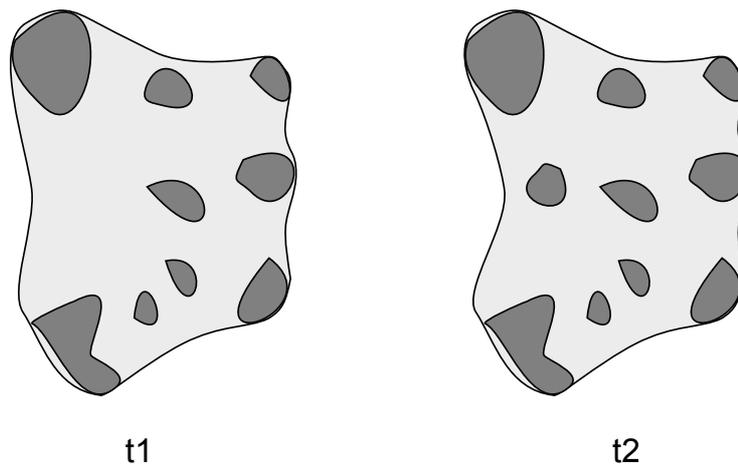


Figure 5.4: Unsuitable concave hulls generated for changing aggregates.

## 5.7 Identifying Geographic Features

As described in Section 3.6, a geographic feature is modelled as an entity which endures over time and that is said to have a life. Then Section 3.6 introduces the concepts of Feature Life, Life Part (LP), and Minimum Life Part (MLP), and defines the identity criteria of a feature based on the connectivity between its spatial extensions throughout its life.

In Progress, the predicate `feature(Aggr, Type, Life)` is defined, whose arguments correspond, respectively, to the aggregation factor employed to determine the spatial extension of the feature at a time; the feature type; and the feature life. The feature life is represented in the system as a list of MLPs, and each MLP is represented using the predicate `mlp(LG, LI)`, where LG and LI are lists of 2 elements, corresponding to the geometries and intervals at the beginning and end of an MLP, respectively.

As discussed in Section 5.6.3, geographic features are asserted in the KB with the objective of improving performance. The mechanism for individuating geographic features is triggered by calling the predicate `make_features(Aggr, Type)`. This mechanism retrieves all *Stars* facts describing spatial extensions of geographic features that match the aggregation factor `Aggr` and the feature type `Type`. That is, these facts are of the form `star(D, A, G, S)`, where D and A are unified with `Aggr` and `Type`, respectively.

The crucial point within the implementation of the predicate `make_features(Aggr, Type)` is a call to the predicate `feature_individuals(LStars, IndLife)`. This predicate is applied to identify individual features (according to the identity criteria) from *Star* facts representing

spatial extensions of many different features. In this predicate, `LStars` is a list of all *Stars* in the KB representing feature extensions (for a given aggregation factor and type), and variable `IndLife` is bound to values corresponding to a life of an individual feature for each solution of the predicate.

## 5.8 Implementing Predicates for Events and Processes

There are a few differences between the form in which predicates relating to events and processes are structured in pure logic and in the system implementation. One difference is that, in the system, a special predicate is used to represent temporal standpoint semantics parameters. This predicate takes the form `time_threshold(0,2,0,0,0,0)`, where arguments correspond to the number of years, months, days, hours, minutes, and seconds. This structure allows the system to perform temporal calculations involving these thresholds and timestamps represented using ISO 8601 date and time descriptions (as required by the temporal reasoner described in Section 5.5). This predicate is passed as arguments of other predicates as exemplified below.

```
proceeds(on, time_threshold(0,2,0,0,0,0), ProcType, Intv).
```

Another important distinction is that the system employs a special argument in predicates `active`, `proceeds` and `occurs` to denote their different variations, such as ‘active at’ and ‘active on’. For instance, the predicate `active` is structured as shown below.

```
active(Active, ProcessType, Interval)
```

In the Prolog predicate shown above, the argument `Active` should be unified with constants `on` or `at`. Specifying these variations using an argument (rather than defining different functors) enables code reuse and consequently makes the code more concise and maintainable. Moreover, these arguments are essential to enable the reversibility of these predicates, that is, to allow free variables to be specified in place of these arguments to ask the system to find appropriate values for them in different situations (though it is not implemented in the current version of Progress). In this case, these arguments would be fundamental for improving efficiency, since tasks which are common for all variations of a predicate can be processed first, and then auxiliary predicates are called to perform specific tasks (e.g., `active_aux(on, ProcessType, Interval)`).

As noted in Section 5.2.3.1, an important characteristic of Prolog is the flexibility in the manner arguments of a predicate can be used, allowing different arguments to

be queried by performing the reasoning in different directions. However, there are certain cases in which such a reversibility of Prolog programs is not possible. Implementing this reversibility is particularly challenging when dealing with spatial and temporal information. For example, the predicate `active(at, ProcessType, Instant)` is not supposed to be used with `Instant` as a free variable. This is not expected since there may be an infinite number of time instants within a given interval. On the other hand, `active(on, ProcessType, Interval)` might be included in a query with `Interval` instantiated or not.

Other cases within Progress implementation which requires special treatment include the case where a given predicate argument corresponds to an element that might have been explicitly asserted in the KB with the objective of gaining efficiency. Thus, the implementation of these predicates should include artifices to enable different possible reasoning flows. A typical example is for an argument representing a geographic feature<sup>13</sup>, where the system has to decide whether the feature will be directly retrieved from the KB or inferred at query time. Specific points in Progress' code where the reasoning direction should be controlled are determined by employing the predicates `nonvar(X)` or `var(X)`, that checks whether a variable `X` is instantiated or not, respectively.

There is still a relevant difference between theory and implementation which should be highlighted. In Chapter 4, the predicate `Active-At` is defined in terms of feature lives, whereas `Active-On` is defined in terms of the `Active-At`. However, this cannot be implemented in the system as it is not able to derive all time instants within an interval. Thus the system implements these predicates the other way round. That is, the activeness of a process on time intervals is inferred from the feature life, and then `Active-At` can be used by specifying a constant value to the argument representing the time instant.

Another important aspect of the implementation of Progress that is worth to be highlighted is that it explicitly asserts facts describing primitive events (beyond the elements mentioned in Section 5.6.3). The motivation for this is similar to that of asserting maximal extensions of geographic features. That is, while bottom-up reasoning (inferring events from MLPs) requires significant amount of time to be performed, top-down reasoning (e.g., to infer process activeness at a given time instant from primitive events) can be performed quite quickly.

---

<sup>13</sup>Arguments denoting geographic features are specified within the predicates representing event and process types, such as in `active(on, process(expanding, F), I)`.

## **5.9 Summary**

This chapter presented Progress, a system prototype implemented to evaluate the applicability of the theory proposed in this thesis. A description of the system's architecture and an overview on its main components has been given. Then specific discussions have been conducted on the approaches to implementing the spatial and temporal reasoners; the STAR model presented in Chapter 3; and the logical framework presented in Chapter 4.

# Chapter 6

## Results of Using the System with Spatio-temporal Data

---

### 6.1 Introduction

This chapter describes the experiments carried out using the system prototype (presented in Chapter 5) with temporal series of topographic data, and examines the results obtained from these experiments.

Results will be assessed for different components of the system, in particular the logic-based approach to modelling spatio-temporal data and the logical framework for representing and reasoning about geographic events and processes. The former will be examined based on the system's capabilities of inferring implicit spatio-temporal data in different situations, and of producing appropriate representations of geographic features that are said to exist over time. The latter will be tested based on its suitability for answering different types of logical queries.

The main objective of these experiments is to evaluate the applicability of the proposed theory to processing real spatio-temporal data, with particular attention to the consistency and coherence of output results. Although the computational performance is not the focus of the experiments, this chapter also describes the overall performance (in terms of executing time) for carrying out a variety of different tasks with the system.

This chapter is organised as follows. Section 6.2 describes the methodology applied

to evaluate the coherence and correctness of results produced by the system. Following this, Section 6.3 describes the case study considered for conducting the experiments and presents the dataset input to the system. Then tasks carried out for preprocessing the input data are described in Section 6.4. Then Section 6.5 examines the experiments and results obtained by using the STAR model to derive implicit spatio-temporal data and to individuate geographic features. Section 6.6 describes the results obtained when the system was tested for answering queries formulated using the logical language  $\mathfrak{R}$ , interpreted by the REGEP framework. Following this, Section 6.7 gives an overall analysis of obtained results and draw some conclusions. Finally, Section 6.8 summarises the chapter.

## 6.2 Evaluation Methodology

The process adopted for evaluating the system is based on objective criteria, to ensure it is clear what are considered successful results of using the theoretical model. Therefore, a set of logical queries and their respective expected results are set out in advance. As the system outputs match the expected results, these experiments indicate the system is effective for identifying the conceptual elements in spatio-temporal data (i.e., events, processes and their participant geographic features). In cases where the set of expected results is considerably large, a sample of this set containing a reduced number of items was considered. These cases will be described in detail in Section 6.6.

Several activities have been conducted to support the evaluation process, both to help set out expected results and to help assess the appropriateness of results. One activity was based on visual analysis of query results. This activity was conducted to support the evaluation of queries whose result consists of spatial elements. That is, produced results are displayed on the interactive map within Progress, and a variety of spatial relationships that were expected between spatial elements can be verified, such as overlaps and external connections. Moreover, expected differences in shape between geometries representing spatial extensions of a certain geographic feature at different times can also be examined.

Storing the spatio-temporal data in a relational database within a DBMS allows the system to be verified based on data analysis using tools provided by the DBMS. Hence, another supporting activity consisted of querying the spatio-temporal database using SQL (Structured Query Language) to verify the consistency and integrity of obtained results. This activity is particularly useful for the cases where the results generated by the system are explicitly asserted in the database (e.g., geographic features and certain *Stars*).

Furthermore, having a relational database whose elements are interrelated by unique numeric identifiers is an additional resource to reduce drastically the risk of misinterpre-

tations that may occur while conducting purely visual analysis. For example, different features may have similar shapes at the same time, which might cause difficulties for individuating them precisely by visual analysis. Also, checking whether a particular geographic feature (with a unique numeric identifier, e.g, 'ID = 243') is a participant in a certain event may provide more accurate results.

For many cases, a single query formulated within Progress using the logical language  $\mathfrak{R}$  would correspond to multiple complex queries using SQL to obtain equivalent results. In other cases, there is no query in SQL that corresponds to those formulated using  $\mathfrak{R}$ . Thus, in this activity the objective is not to obtain equivalent results using SQL for comparing against those produced by Progress. Rather, SQL queries are employed to verify the integrity and consistency of results of produced by the system's inference mechanism. For example, through SQL, it can be easily verified whether a geographic feature's life is consistent (e.g., whether their time instants are in ascendant order, whether consecutive spatial extensions are connected).

The experiments described in this chapter were conducted to evaluate the general suitability of the system for deriving implicit facts and for answering logical queries. However, a number of automated unit tests<sup>1</sup> have also been developed to ensure the correctness of the system's code (including the Prolog code that implements the proposed logical framework). Although those tests are based on reduced KBs, produced artificially (and manually) to represent simple test scenarios, they are important not only for ensuring the code is correct, but also for allowing the proposed theory to be extended safely whenever it is needed.

### 6.2.1 Auxiliary Tools

Some auxiliary tools were used to support the system evaluation process. As discussed before, the querying mechanism provided by the DBMS was used both to help set out expected results and to help examine the appropriateness of the results produced by the system.

Another auxiliary tool was also used to help explore spatio-temporal data and to support visual analysis of obtained results. This tool was uDig, an open source desktop tool for GIS data access, editing, and viewing. Figure 6.1 shows the uDig's main screen.

---

<sup>1</sup>Unit Testing, [https://en.wikipedia.org/wiki/Unit\\_testing](https://en.wikipedia.org/wiki/Unit_testing)

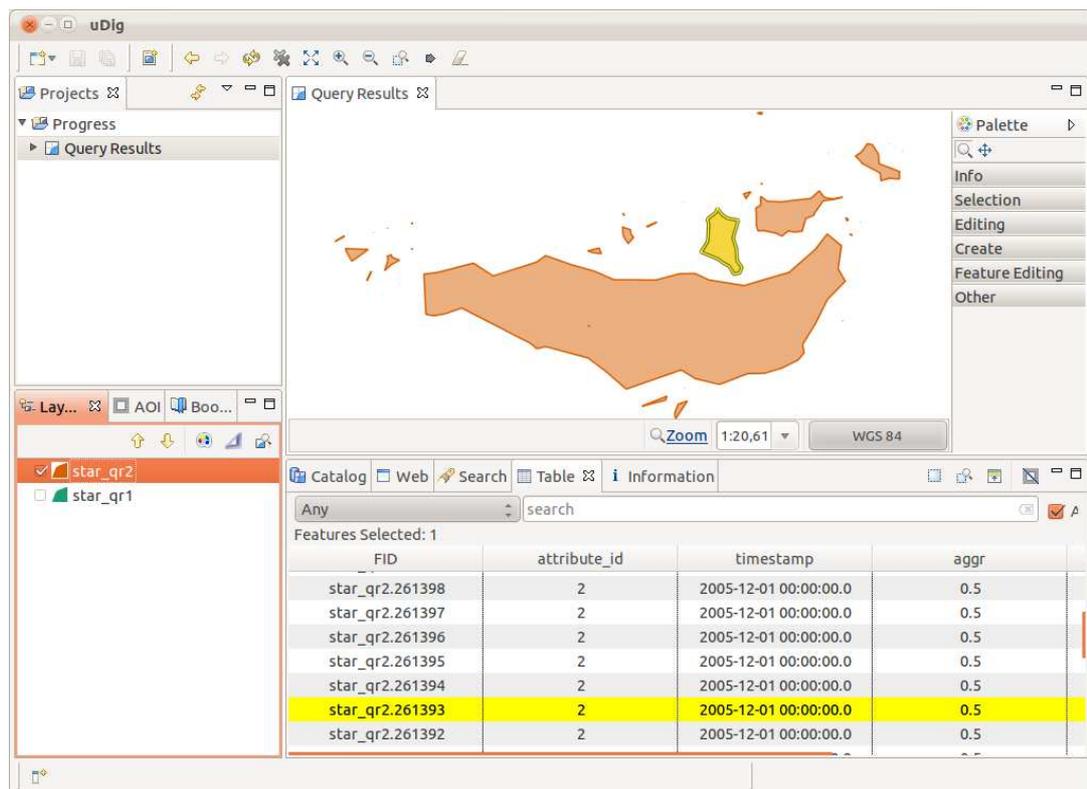


Figure 6.1: uDig's main screen.

Furthermore, an auxiliary tool was developed to support the activity of setting out expected results of logical queries. This tool can be executed within Progress and is capable of displaying temporal series of spatial data. These data can be loaded either from a set of Shapefiles<sup>2</sup> or from a spatial database stored within a DBMS. The main screen of this tool is shown in Figure 6.2.

In that visualisation tool, geometries associated with different timestamps can be visualised simultaneously, and different colours are employed to distinguish their temporal information. The system can render geometries using a set of 11 colours (shown on the bottom of Figure 6.2), ranging from light blue to red. Red is applied to colour the most recent elements (i.e., those associated with greatest timestamp) amongst those displayed, whereas blue geometries represent the oldest elements. For instance, suppose a temporal series containing polygons associated with different calendar years (from 1951 to 1970) is loaded in the system. In this case, 1970's geometries are coloured in red, 1969's ones in brown, 1968's ones in orange and so on. Since there are only 11 colours available, all polygons associated with years 1951 to 1959 are coloured in blue.

<sup>2</sup>Shapefile, <http://en.wikipedia.org/wiki/Shapefile>

This tool allows the user to select the portion of the time series that should be displayed (e.g., from 1955 to 1968) and to change the temporal granularity considered for displaying geometries, which is set to 1 by default. For example, by changing it to 2, the polygons of the example above would be displayed as follows: polygons associated with 1969-1970 in red; those corresponding to 1966-1967 in brown, and so on. This functionality is useful for changing the level of detail the data is observed, helping identify, for example, areas where spatial changes occur more frequently.

Figure 6.2 can be used to illustrate the way visual observations can be conducted using that tool. For instance, from the 8 polygons located approximately in the centre of the figure, an evolving geographic feature can be identified, whose spatial extensions at different times would correspond to the concave hull applied to the aggregation of the following polygons:

1. three polygons in light blue; then
2. previous polygons (item 1) + two polygons in light green; then
3. previous polygons (item 1 and 2) + one polygon in dark green; then
4. previous polygons (item 1, 2 and 3) + one polygons in yellow;

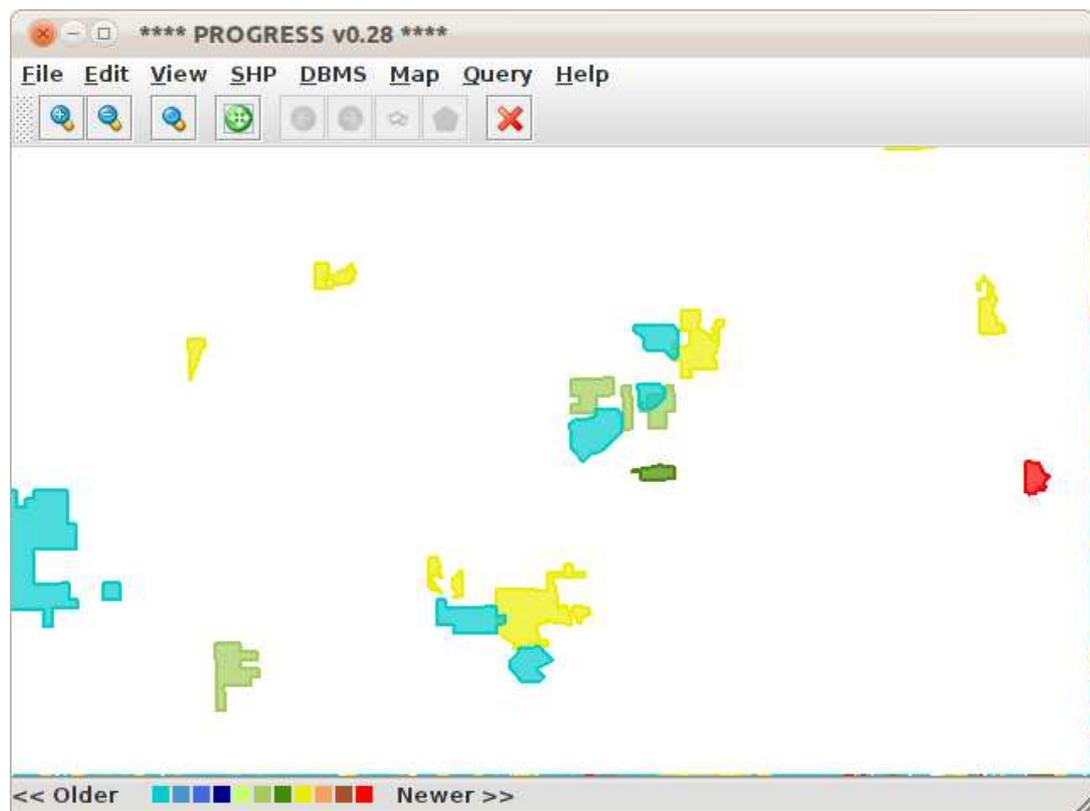


Figure 6.2: Visualisation tool main screen.

### 6.3 Case Study and Dataset Description

Experiments using Progress have been conducted in the form of a case study, investigating the phenomenon of deforestation in the Amazon rainforest in a 7-year period (between June 2004 and May 2011). The dataset used in this case study consists of distinct sets of polygons, each of which representing regions deforested in Brazilian Amazon in a particular calendar month. This 84-month period dataset contains a total of 47,459 polygons.

These data are produced as an output of DETER<sup>3</sup>, project at INPE<sup>4</sup>, that uses remote sensing techniques to detect land cover changes within the Brazilian Amazon area [80]. This dataset is in the public domain and is published on DETER's website. Moreover, an auxiliary dataset containing the Amazon boundaries has been used for an improved visualisation of results. This dataset is maintained by the Brazilian Ministry of Environment (MMA) and is available online for the public domain<sup>5</sup>.

<sup>3</sup>DETER (Near Real Time Deforestation Detection in the Amazon Region), <http://www.obt.inpe.br/deter/>

<sup>4</sup>INPE, National Institute For Space Research, <http://www.inpe.br/ingles>

<sup>5</sup>MMA open data, <http://mapas.mma.gov.br/i3geo/datadownload.htm>

Given the large number of polygons contained in that dataset, it has been helpful for carrying out experiments considering different scenarios, where events and processes can be identified under many different circumstances. Nonetheless, as this dataset is limited to a single type of coverage (i.e., denoting that a certain region is deforested), it has not been sufficient for evaluating the adequacy of the system for performing all the inferences described in Chapter 3. Hence, these specific capabilities have been evaluated using a number of synthetic datasets. Although these datasets contain a reduced number of polygons and shorter temporal series, they include various different attributes. Therefore these datasets are adequate for the objectives of the evaluation process. Experiments conducted based on synthetic data will be described in Section 6.5.1.

## 6.4 Data Collection and Preprocessing

The data generated by DETER Project are distributed in Shapefile format, where each file contains a set of polygons corresponding to regions deforested in a particular period of time. Data were released on a monthly basis since June 2004, and more recently they have been released every 15 days. As the experiments carried out in this thesis were based on data up to May 2011, the raw data collected from INPE consists of one shapefile per month.

Although these data follow an overall high quality standard, they are still the subject of certain inaccuracies and inconsistencies, which should be minimised before using them as an input in Progress. For this reason, different mechanisms for preprocessing the data have been developed. One of the problems affecting these data is the existence of unexpected overlapping polygons. Since each polygon represents a new deforested region (i.e., a region which is classified as ‘forested’ in one observation and then is classified as ‘not forested’ in the subsequent observation), two overlapping polygons associated with different periods could only exist if the intersection between them represented a region that has been deforested, reforested and then deforested again. However, in most cases, the temporal distance between them is not long enough to make possible the occurrence of a sequence of events of this nature. For instance, in some cases such a temporal distance is of just 1 or 2 months. Figure 6.2 has been produced using the raw data, and therefore some undesired overlaps (i.e., between polygons of different colours) can be visualised.

To deal with this problem, an algorithm to eliminate undesired intersections between polygons has been developed within a PostgreSQL stored procedure. The most important decision in developing this algorithm is whether the region corresponding to the intersection between two polygons should be regarded as a new polygon, or as part of one of

them. It has been observed that these overlaps frequently occur between polygons whose areas are considerably different (i.e., one is much larger than the other). In addition, there are many cases in which the smaller polygon is contained within the other. Thus, the developed algorithm generalises the solution by considering that the intersection between two polygons belongs to the larger polygon. Consequently, for the cases where one polygon is inside the other, the smaller polygons are removed from the dataset. As a result of this data preprocessing, overlapping polygons become externally connected, and therefore become useful elements for carrying out experiments for calculating spatial extensions of geographic features as maximal well-connected regions of particular coverages, as discussed in Chapter 3.

Another issue with these data is the existence of missing elements in the temporal series. That is, there are no data available for 7 months amongst the 84 of the series investigated. It is not clear whether these elements are missing due to some kind of technical difficulties faced by the provider or whether a missing element means that no deforested area has been identified. However, the existence of periods in which geographic features remain unchanged is useful for evaluating some characteristics of the proposed theory, such as the mechanism for determining geographic features' lives and the method of handling temporal vagueness. Thus, for this reason, it has been assumed that missing elements represent periods where no change have been observed.

Finally, it has been noticed the raw data contain a significant number of polygons (34,645 out of 39,428) which are extremely small in area and widely dispersed in space. Consequently, when these tiny polygons are processed for composing geographic features, each one originates a distinct feature that does not undergo any change over time. For this reason, these features are not relevant for evaluating the system in terms of its ability to detect events and processes. Such tiny polygons could only be used to compose extensions of geographic features which perform spatial changes by aggregating them using are considerably great aggregation factor. Nonetheless, the obtained result is again not relevant, as the area of the concave hull generated for aggregate is exorbitantly greater than the sum of the areas of originating polygons. Hence, it has been realised that these polygons could be removed from the dataset without causing significant negative impact to the experiment results. Conversely, by removing these elements, the dataset has become more concise, containing 4,783 polygons which originate features that in fact perform a variety of different changes and are therefore helpful for evaluating the system.

## 6.5 Evaluating the STAR Model

This section discusses the results obtained by using the system to derive implicit *Star* facts through the derivation rules described in Section 3.5.2.

### 6.5.1 Deriving *Stars* with Different Attributes

As discussed in Chapter 3, a number of implicit *Star* facts can be derived by relating facts of different attributes, such as deriving a heterogeneous region from a set of homogeneous regions. Thus a number of experiments have been conducted using the system prototype to evaluate such system's inferences capabilities. This section aims to describe some synthetic data produced for conducting this evaluation and to describe some results obtained from experiment using these data.

These experimental data comprise:

- Facts representing the following attributes:
  - Homogeneous coverage attributes: *fresh water*, *crop plantation*, *pasture*, *asphalt* and *built-up*;
  - Heterogeneous coverage attributes: *agricultural* and *urbanised*;
  - Simple feature attributes: *lake*;
  - Compound feature attributes: *farm* and *town*.
- *Star* facts associating polygons with different homogeneous coverage attributes and 5 different timestamps: 01/01/2009, 01/01/2010, 01/01/2011, 01/01/2012, 01/01/2013 (all at 00:00:00).
- Facts relating attributes:
  - $CP(\textit{crop plantation}, \textit{agricultural})$ ,  $CP(\textit{pasture}, \textit{agricultural})$ ,  
 $CP(\textit{asphalt}, \textit{urbanised})$ ,  $CP(\textit{built-up}, \textit{urbanised})$ ;
  - $MP(\textit{fresh water}, \textit{lake})$ ,  $MP(\textit{agricultural}, \textit{farm})$ ,  $MP(\textit{urbanised}, \textit{town})$ .

Figure 6.3 exhibits a graphic representation of the *Star* facts explicitly asserted in the synthetic KB. It can be seen in this figure that the knowledge base does not contain information on a particular coverage type for all time instants, what poses additional difficulties for understanding how a particular geographic feature evolves over time. Scenarios of this kind might be resulting from the integration of distinct datasets containing

information on different themes (e.g., one dataset containing information on fresh water, another on distinct land use related to agriculture).

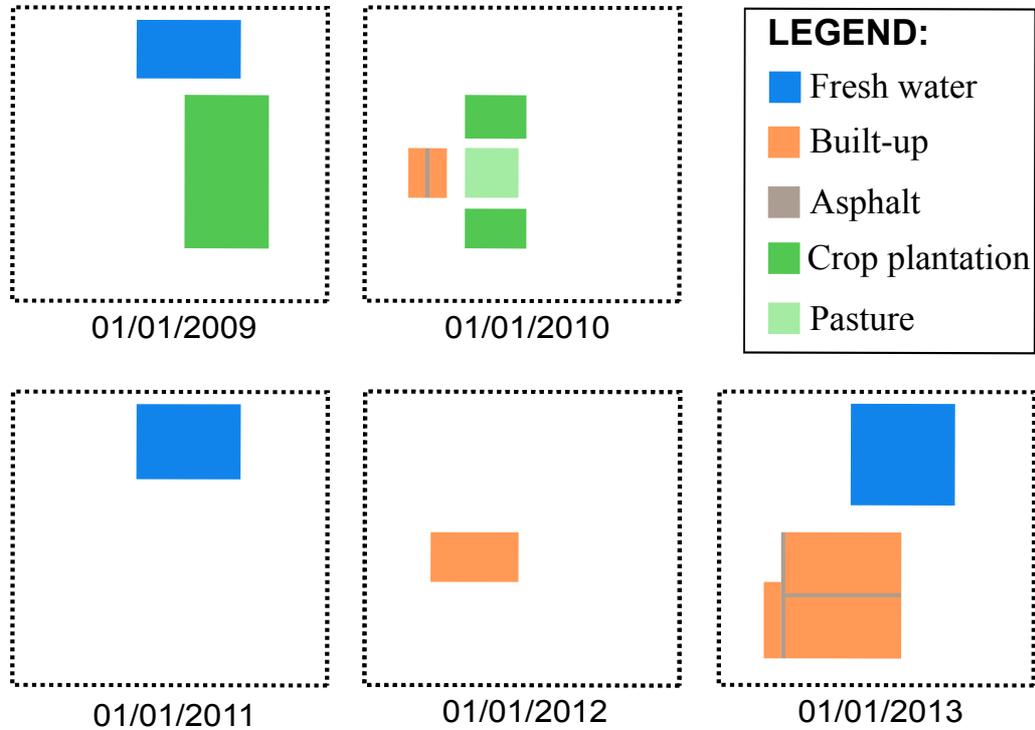


Figure 6.3: *Star* facts explicitly asserted in the synthetic KB.

From this initial KB, Progress successfully derived implicit *Star* facts according to Derivation Rule DR3, which considers the principle of inertia. Then derived facts have been explicitly asserted in the KB. Figure 6.4 shows those derived facts together with the facts initially asserted in the KB.

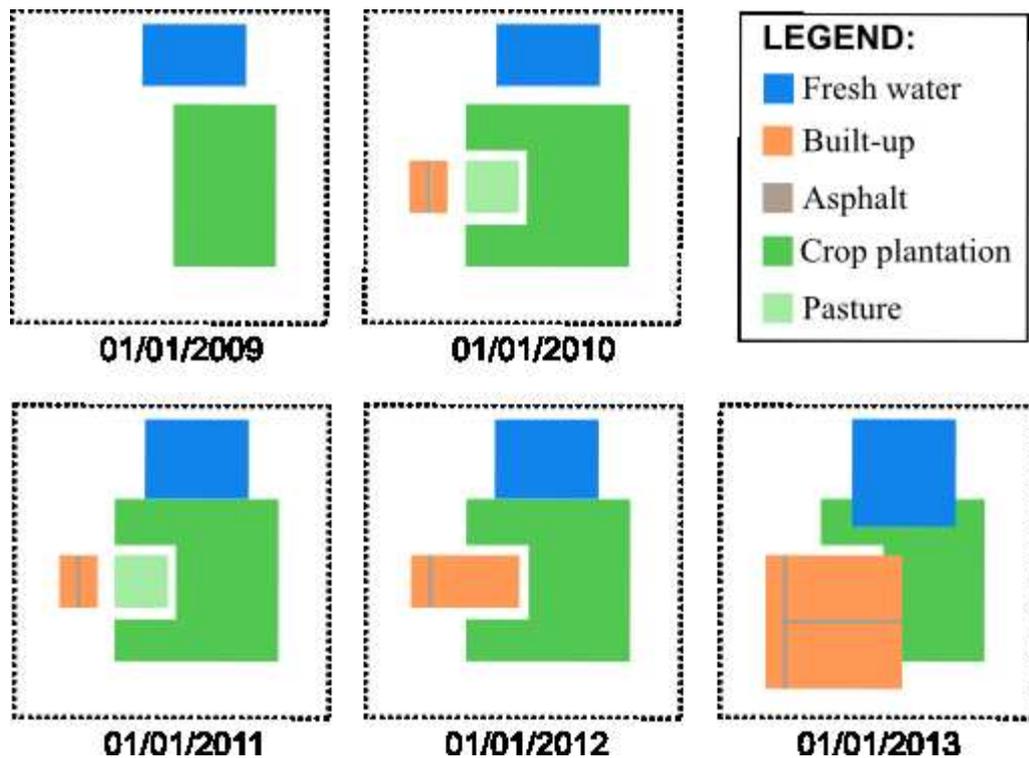


Figure 6.4: *Star* facts explicitly asserted in the initial database together with other *Star* facts produced by Derivation Rule DR3.

Now that the KB contains information on the homogeneous coverages that hold at all time instants, Progress correctly derives *Star* facts representing heterogeneously covered regions; identifies maximal extensions of well-connected regions for different coverages; and, based on the latter, infers the spatial extensions of lakes, farms and towns, which are the geographic features of interest in this test scenario. As described in Section 5.6, implicit facts representing spatial extensions of geographic features are also explicitly stored in the KB. A graphic representation of these facts is shown in Figure 6.5. By contrasting Figures 6.4 and 6.5, it can be noticed that disconnected regions of crop plantations and pasture have originated an agricultural region. This occurred because *agricultural* regions have been represented as *vaguely connected regions*, where an aggregation factor of ‘6.5’ has been specified (whilst the minimum distance between these regions is ‘5’).

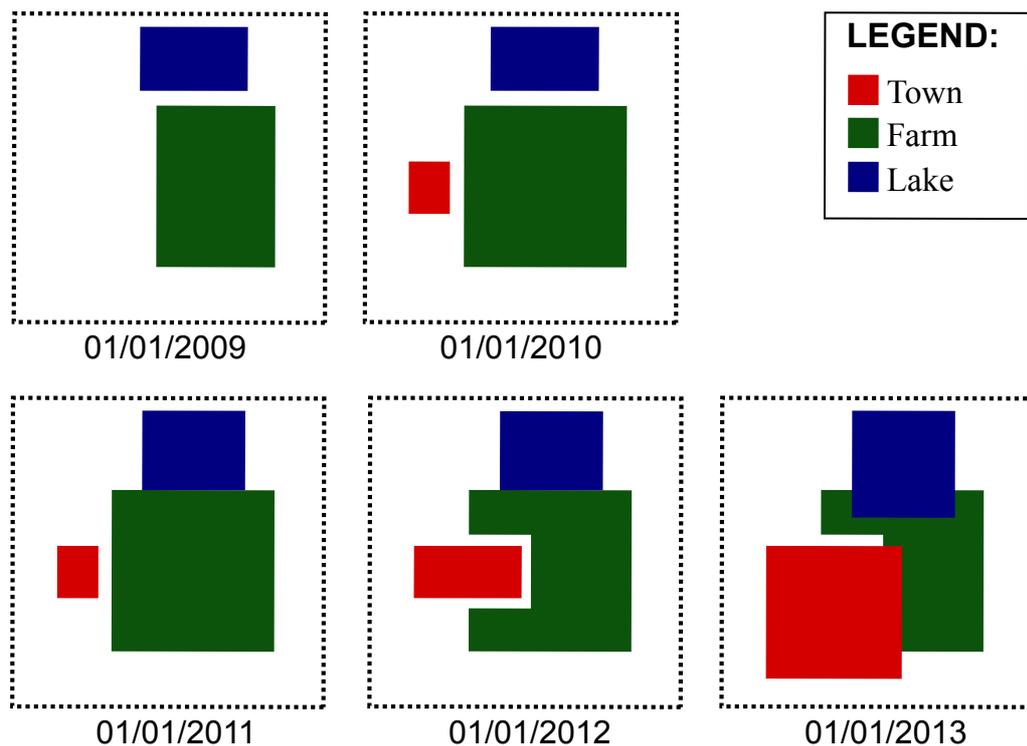


Figure 6.5: *Star* facts representing spatial extensions of geographic features. These facts have been produced by applying DR2 or DR6, and then DR4. Once the maximal extensions of geographic features are inferred by DR4, the corresponding *Star* facts are explicitly asserted in the KB.

Progress has been successful at processing the initial KB and producing appropriate implicit facts. As noted in Chapter 3, the goal of inferring the spatial extension of geographic features is to individuate these features and then analyse their evolution over time. However, even before individuating features, it is possible to formulate a variety of useful queries using the Progress command-line terminal. To illustrate, three examples of queries submitted to Progress are given in Queries 1, 2, and 3. Each example contains a textual description of the query, its representation using the syntax employed within Progress, and the obtained results.

The expected result for Query 1 is better visualised graphically. Thus, in the Prolog specification of this query (Query 1b), a variable is named using the prefix `G_` to instruct the system to display on the map pane the value(s) bound to this variable. In Query 1b, the numeric identifier (ID) 32 has been employed to denote the feature type *town*.

Query 1 is described as follows.

Q 1a *What were the spatial extensions of the existing towns at 2011 00:00:00?*

Using the Prolog-like syntax adopted in Progress, Query 1 can be specified as follows:

```
Q 1b ?- star(32,G_Town,'2011-01-01 00:00:00').
```

The system output for Query 1 is exhibited in Figure 6.6, where it can be seen that Progress displays a region whose shape matches the expected result (as shown in Figure 6.5). Progress displays the results by fitting the output to the screen. Therefore, when a single element is displayed, it is not possible to be entirely sure of whether it actually corresponds to the element which is expected to be displayed (or whether another polygon with the same shape has been displayed, for example). Hence, for most experiments, displaying more than one element on the map help examine the results, as distinct regions act as spatial references to each other<sup>6</sup>.

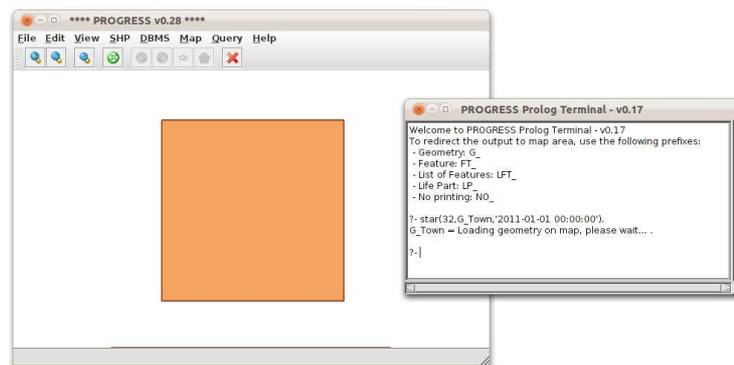


Figure 6.6: System output for Query 1.

Let us now describe an example of a query where the expected result consists of more than one polygon. This is described in Query 2a, as follows.

```
Q 2a What lakes and farms were externally connected at 2012 00:00:00?
```

The expected result for Query 2b is also required to be visualised graphically, and therefore the prefix `G_` is used again to name the geometry variables. In Query 2b, the IDs 21 and 31 correspond to the feature types *lake* and *farm*, respectively.

<sup>6</sup>Alternatively, a rectangle representing the area of interest can be displayed together with the results. For instance, this rectangle might correspond to the dotted frames shown in Figure 6.5

```
Q 2b ?- star(21,G_Lake,'2012-01-01 00:00:00'),
      star(31,G_Farm,'2012-01-01 00:00:00'),
      ec(G_Lake,G_Farm).
```

The system output for Query 2 is exhibited in Figure 6.7, where it can be seen Progress successfully displays the polygons which match the expected results shown in Figure 6.5.

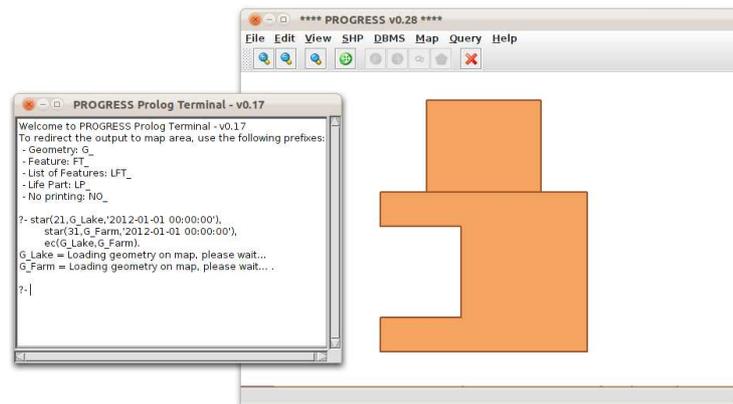


Figure 6.7: System output for Query 2.

Differently from the previous 2 examples, the expected answer for Query 3 is a numeric value (i.e., 1). Thus now there is no need for redirecting the system output to the map pane. In this case, the textual output for Query 3b is shown just below the query, exactly as displayed at the system's terminal. In addition, variables whose values are not of interest are named using the prefix `NO_`.

Query 3 is described as follows.

Q 3a *How many farms were there at 01/01/2012 00:00:00?*

Query 3b exhibits a form in which this query could be submitted to Progress. It can be seen that Progress' answer is again in accordance with the expected result.

```
Q 3b ?- findall(NO_Farm,
              star(31,NO_Farm,'2012-01-01 00:00:00'),
              NO_List),
      length(NO_List,N).
N = 1.
```

## 6.5.2 Identifying Spatial Aggregates

The system mechanism for deriving *Stars* that correspond to aggregation of other spatial regions has been evaluated. Before describing the results obtained from the experiments, a brief discussion is given on the approach to measuring distance between polygons at data level.

### 6.5.2.1 Distance Measurements

Whereas *cartesian coordinates* represent linear distance as plotted on a plane, *geographic coordinates* measure angular distance, and therefore units are given in ‘degrees’. Spherical coordinates describe angular coordinates on the globe, that is, a point is specified by the rotation angles from both a reference meridian (longitude), and from the equator (latitude). It is possible to treat geographic coordinates as approximate cartesian coordinates for performing spatial calculations. Nevertheless, measurements of distance will not be as accurate as measurements obtained by performing true spherical calculations. On a sphere, the size of one ‘degree square’ is not constant. This value is greater for regions closer to the equator and are smaller for those nearer to the poles.

However, cartesian calculations are computationally substantially less expensive than calculations on a sphere. One of the reasons is because spherical calculations require many trigonometric calculations, which are notably costly. For example, as described in [64], the cartesian formula for distance (Pythagoras) in Postgis involves just one call to the *sqrt()* function; whereas the spherical formula for distance (Haversine) involves two *sqrt()* calls, one call to *arctan()*, four calls to *sin()* function and two *cos()* calls. As discussed in [64], if distance measurements should be obtained from a dataset that is geographically dispersed (covering much of the world), the use of geographic coordinates is recommended. If, on the other hand, the spatial data is geographically compact (e.g., contained within a state), cartesian projection can be used without significant impact on accuracy.

The experiments carried out in this thesis are based on either datasets produced artificially or on a real dataset that is spatially compact (Section 6.3). In addition, maximal accuracy for distance measurements is not determinant to validate the theoretical approach proposed here. On the other hand, the experiments were carried out using a typical desktop computer, and therefore performance issues were considered relevant due to its impact on the development productivity (since experiments are repeated numerous times during the development process).

Hence, the experiments described here were conducted by adopting the *cartesian pro-*

jection, and therefore values employed for spatial aggregation factors are expressed in *angular degrees*. On the WGS84 spheroid at sea level at the equator, one latitudinal degree corresponds to *110.6 km*, whereas one longitudinal degree corresponds to *111.3 km*.

### 6.5.2.2 Calculating Spatial Aggregates

The mechanism for generating (maximal) spatial aggregates has been evaluated on the KB containing 260,762 facts representing distinct homogeneous regions. These facts correspond to those initially present in the KB together with the new facts produced by the derivation rule DR3 (presented in Chapter 3), which deals with the principle of inertia.

The algorithm has been executed by setting aggregation factors 0.5, 0.3 and 0.1. These factors correspond, respectively, to approximately 55.5 km, 33.3 km and 11.1 km. For these experiments, a fixed value of 0.99 has been adopted for the concave hull factor. As was to be expected, the results have shown that the smaller the aggregation factor, the greater the number of derived facts. For a factor of 0.5, it was found that 3,961 facts have been derived from the 260,762 previously asserted in the KB. However, when the factor is set to 0.3 and 0.1, the number of derived facts raises considerably — to 9,825 and 70,806, respectively.

On the other hand, the elapsed time for executing the algorithm with those 3 different aggregation factors has not changed significantly, ranging from just under 115 minutes to approximately 120 minutes. This execution time comprises the time to group regions, to determine the concave hull for each group and to explicitly assert them in the KB. The obtained results are summarised in Table 6.1, which shows that the number of facts in the KB has risen to 345,354 after executing the aggregation mechanism.

Table 6.1: Results obtained from executing the mechanism for generating spatial aggregates on a KB containing 260,762 facts, by setting 3 distinct aggregation factors and a fixed value for the concave hull factor. The resulting KB contains 345,354 facts.

<b>Aggregation Factor</b>	<b>Concave Hull Factor</b>	<b>Nº of Facts Generated</b>	<b>Execution Time</b>
0.5	0.99	3,961	114'49.562"
0.3	0.99	9,825	118'15.233"
0.1	0.99	70,806	120'04.430"

### 6.5.3 Execution Order of Derivation Rules

Although the order in which certain derivation rules are applied is not be relevant from the conceptual and logical point of view, this is be crucial within a system which implements these rules. This is particularly important for derivation rules used to produce facts to be explicitly stored in the KB. In general, these derivation rules are those which can produce a finite number of implicit facts from a finite KB. Examples of this kind of derivation rule are those applied to infer the maximal extension of regions of a particular coverage (DR2, DR4, DR6) and the rule used to make inferences about regions which remain unchanged over time (DR3). These derivation rules are described in Chapter 3. For short, the facts produced by those rules will be referred to here as *aggregated* and *replicated Stars*, respectively. Two approaches could be considered to execute these derivation rules, as follows:

- **Approach 1:** producing all replicated *Stars* and then calculating aggregates. By replicating *Stars* for different time instants, the number of facts per time instant increases considerably. Therefore, a greater number of facts should be considered for calculating aggregates at each different time instant. However, this approach can guarantee that resulting aggregates will be maximal, that is, will represent the extension of a certain feature which is said to exist at a particular time instant.
- **Approach 2:** deriving aggregated *Stars* first, and then inferring replicated *Stars*. However, it can be noticed that produced aggregates might not be maximal. For instance, a *Star* replicated from a time instant  $t_1$  to a time instant  $t_2$  could be spatially connected (at  $t_2$ ) to an aggregated *Star* produced for  $t_2$ . Thus, in order to obtain maximal aggregates, this approach requires re-calculating aggregates for each time instant after the replication process is completed. Although there is a drawback to calculating aggregates twice, the aggregation processes involve considerably fewer elements in comparison with the Approach 1.

Experiments have been conducted for both approaches, using the dataset of deforestation of the Amazon (described in Section 6.3) by setting different aggregation factors (i.e., 0.5, 0.3, and 0.1). A constant value for the concave hull factor has been adopted (i.e., 0.99). These experiments revealed that the execution times are similar between these approaches. In addition, it could be observed that none of them are better for all situations. The obtained results are summarised in Table 6.2.

Table 6.2: Results obtained for different approaches to ordering the execution of mechanisms to derive implicit *Star* facts and store them in the KB.

Aggregation Factor	Concave Hull Factor	Execution Time	
		Approach 1	Approach 2
0.5	0.99	115'55.961"	112'14.664"
0.3	0.99	129'21.632"	123'32.974"
0.1	0.99	121'10.829"	122'26.555"

The system's mechanism for replicating *Stars* via Derivation Rule DR3 has been examined separately from the mechanism for calculating aggregates. When executed on the initial KB containing 4,783 facts, it produces 255,979 new facts, resulting in a KB containing 260,762 facts. This process has been completed in approximately 66.4 seconds, which indicates this task represents the smallest portion of the execution time shown in Table 6.2. Whereas the greatest amount of time corresponds to the execution of the mechanism for calculating spatial aggregates.

As discussed in Chapter 3, when a *Star* is replicated from a time instant  $t_1$  to  $t_2$ , the mechanism should check whether there is no *Star* at  $t_2$  which intersects this and whose coverages are conflicting (i.e., no intersection is expected). However, since this experimental dataset consists of regions of a single type of coverage (i.e. 'deforested'), and no facts denoting 'reforested' regions are considered, no conflicting intersections are found during the replication process. Consequently, this process is completed with no overhead for calculating costly spatial differences between polygons. Hence longer execution times should be expected for performing this task in more complex scenarios.

#### 6.5.4 Geographic Feature Generation

The spatio-temporal dataset about deforestation in the Amazon (described in Section 6.3) was used to form the initial KB containing 4,783 facts of the form  $Star(c, r, t)$ , where  $c$  is a type of region coverage which denotes the spatial region  $r$  is 'deforested' at time instant  $t$ . From this initial KB, the derivation rules DR3 and DR6 were executed. Since some derived facts are explicitly asserted in the KB (as discussed in Chapter 5), the number of facts in the KB rose to 345,354 (as shown in Table 6.1). This KB contains facts corresponding to maximal aggregates of different coverages for each distinct time instant and for different aggregation factors.

By executing derivation rule DR5, these facts representing maximal aggregates are

identified and feature extensions can be derived by the system. These extensions correspond to the maximal extensions of well-connected regions of coverage  $c$ , where  $c$  is associated with a certain feature type  $f$  (using CP or MP relations). That is, implicit facts of the form  $\text{Star}(u, r, t)$  are derived by the system, where  $u$  denotes the type of geographic feature and  $r$  is the spatial region whose extension corresponds to the extension of that feature at time instant  $t$  (i.e.,  $\text{ext}(f, t) = r$ ). In the experiment described here, 84,577 new facts of the form  $\text{Star}(u, r, t)$  were derived by the system, which represents approximately 24.49% of the facts in the input KB. The geographic feature type  $u$  used in these experiments are described as ‘ex-forest’, whereas the phenomenon of deforestation is analysed in terms of *expansion* of these features.

The mechanism for generating geographic features from a KB containing *Star* facts were executed by setting 3 different spatial aggregation factors 0.5, 0.3 and 0.1. That is, for each different factor, a different subset of *Star* facts has been considered for producing features’ lives. As could be expected, for each distinct factor, a different number of features was produced, as well as distinct number of MLPs per feature were obtained. These results are summarised in Table 6.3. This table exhibits, for each aggregation factor, the total number of features produced; the average number of MLPs per feature; and the number of MLPs of the feature(s) with the shortest and longest lives, respectively.

Table 6.3: Results obtained when deriving geographic features from *Star* facts representing maximal spatial extensions of deforested region aggregates. For each aggregation factor evaluated, the number of different features identified and statistics on the number of MLPs per feature (i.e., average, minimum and maximum) are shown.

Aggregation Factor	Nº of Features	Nº of MLPs per Feature		
		Avg	Min	Max
0.5	130	33.04	1	83
0.3	296	31.98	1	83
0.1	1418	49.91	1	83

Furthermore, the mechanism for generating geographic features has been examined with regard to its execution time. This mechanism has been executed for the 3 subsets of *Star* facts representing the extensions of spatial aggregates with factors 0.5, 0.3 and 0.1 (in this order). From the first subset, containing nearly 4,000 originating facts, 130 geographic features have been identified in just over 10 minutes. For the second subset, which contains approximately 2.5 times the number of originating facts more than the first

subset (just over 9,800 facts), the execution time (nearly 38 minutes) was approximately 3.8 times the one observed for the first experiment. However, whilst the number of facts in the third subset is approximately 7 times the number of facts in the second subset, the execution time raised by approximately 37.5 times (to nearly 24h), compared to the second experiment. These results are summarised in Table 6.4.

Table 6.4: Execution time for deriving geographic features from *Star* facts representing maximal spatial extensions of deforested region aggregates. For each aggregation factor evaluated, the number of originating *Star* facts and the number of features identified are shown.

Aggregation Factor	Number of Originating Facts	Number of Features	Execution Time
0.5	3,961	130	10'02.264"
0.3	9,825	296	37'59.410"
0.1	70,806	1418	23°46'11.589"

## 6.6 Evaluating the Logical Framework

This section describes the results obtained from using the system to answer logical queries about geographic events and processes. As discussed before, these experiments are conducted in the form of a case study investigating the phenomenon of deforestation in Amazon rainforest. The KB used here consists of factual elements denoting geographic features of type 'ex-forest'. These features have been previously identified using 3 different aggregation factors (i.e., 0.5, 0.3 and 0.1) and have been explicitly asserted in the KB.

### 6.6.1 Logical Queries Used

A variety of different queries have been formulated to experiment the system. These queries have been divided into 3 groups:

- *Queries about processes*, used to examine the system output for different cases where a processes is said to be active and to proceed.
- *Queries about events*, in which a focus is placed on the use of the predicates for representing different types of event occurrences.

For each group mentioned above, queries vary in the following aspects:

- *Queries with no free variables.* As described in Section 5.2.3, when a query contains no free variables, the output is simply ‘true’ or ‘false’, according to whether the proposition represented by the query holds within the model. An example query would be to confirm whether an event of a given type occurred on a specified time interval. Since Progress is examined here mostly for consistency, a simple response acknowledging the query is true or false is sufficient for the output.
- *Queries with One or More Free Variables.* As noted in Section Section 5.2.3, the objective of using free variables in a query in to identify and display variable bindings that makes the proposition be true. Most queries of interest here will contain at least one free variable. Examples of values assigned to variables would be time intervals on which events of a certain type occur.

Queries also vary in terms of different spatial or temporal constraints used and in terms of values specified for the activeness threshold. In Addition, Prolog metapredicates `findall` and `setof` have been employed in different situations where multiple values bound to a single variable should be displayed together. The syntax of these functions and other Prolog elements are described in Section 5.2.3.

## 6.6.2 Queries about Processes

The discussion on queries formulated to experiment the system begins by describing those applied to obtain information on processes which are said to be active or to proceed. As mentioned before, in the case study followed here, deforestation is described in terms of expansion of geographic features of type ‘ex-forest’. Therefore, in all examples which follow, a single geographic feature type ‘ex-forest’ is considered, and a common classifier ‘expanding’ is employed to represent process types. Thus, for readability, this information will be suppressed in the description of some query examples presented here. Example queries are also described using the Prolog syntax adopted in Progress, in which the geographic feature type ‘ex-forest’ is referred to by its numeric identifier ‘4’.

### 6.6.2.1 Queries Containing No Free Variables

A number of different queries containing no free variables have been formulated, and the results obtained by some of them will be discussed here. These queries include those containing either instantiated variables (i.e., variables bound to a specific constant value) or anonymous variables, which are represented in Prolog by a single underscore ‘\_’ meaning ‘any term’.

As discussed in Section 4.10, the activeness of a process is defined from primitive elements grounded upon the data, and other predicates to represent characteristics of both processes and events are based on the concept of process activeness. Hence, the most general query about geographic phenomena that could be formulated within the system would be to confirm whether any process has ever<sup>7</sup> been active. In the context of the case study conducted here, this query could be described as follows.

Q 4a *Has deforestation ever been active in the Amazon?*

This query could be described using the terminology employed within the logical framework, as follows.

Q 4b *Has any process (whose classifier is ‘expanding’ and whose participant is a geographic feature of type ‘ex-forest’) been active at any time instant?*

As discussed before, the same process classifier (i.e. ‘expanding’) and geographic feature type (i.e., ‘ex-forest’) are considered in all examples presented in this section. Thus, for readability, this information will be suppressed in some query examples which follow.

Query 4b is formulated in terms of activeness of processes at time instants, and therefore a proposition using the predicate Active-At would suit its purpose. However, in the current version of Progress, this predicate is not supposed to be used with uninstantiated time instant variables, as it would require producing an infinite number of time instants. As noted in Section 5.8, process activeness is inferred from the difference between the spatial extension of a feature at two different time instants (i.e., within a certain MLP), which in turn correspond to a time interval. Therefore, this query could be specified in terms of process activeness on time intervals, as follows:

Q 4c *Has a process been active on any time interval?*

In Progress, this query could be easily specified by leaving the interval variable anonymous, as shown in Query 4d. In addition, by specifying a geographic feature as

---

<sup>7</sup>Although ‘ever’ is used here meaning ‘at all times’ (see <http://oxforddictionaries.com>), it should be noted that the set of all possible time instants at which a process might be active is limited by the KB. In the case study considered here, it is limited to the period between June 2004 and May 2011. That is, for any time instant  $t$  outside that interval,  $\text{Inactive}(p,t)$  holds (for any process type  $p$ ).

`feature(0.1,4,-)`, the system only considers features whose aggregation factor is '0.1' and type is '4' ('ex-forest'), without regard to the the feature's life (which indicated by the anonymous variable '-').

Q 4d ?- `active(on, process(expanding,feature(0.1,4,-)), _)`.

The answer provided by the system for Query 4 is 'true.', meaning that there is at least one feature and one time interval which satisfy the proposition represented by the query. The reliability of this result can be verified by visualising geographic features using the interactive map, in which it is possible to observe that many features expand over time. For instance, Figure 6.8 exhibits the spatial extension of a particular geographic feature<sup>8</sup> of type 'ex-forest' at 3 consecutive months, where it can be noticed the feature's spatial extension clearly expands between these 3 time instants (which determine 2 consecutive feature's MLPs). Additionally, according to the assumptions made in Chapter 4, this feature is also regarded as expanding throughout the 2 intervals bounded by these time instants.

---

<sup>8</sup>This geographic feature corresponds to the first solution for the query `FT_F=feature(0.1,4,-), afeatures(FT_F)`.

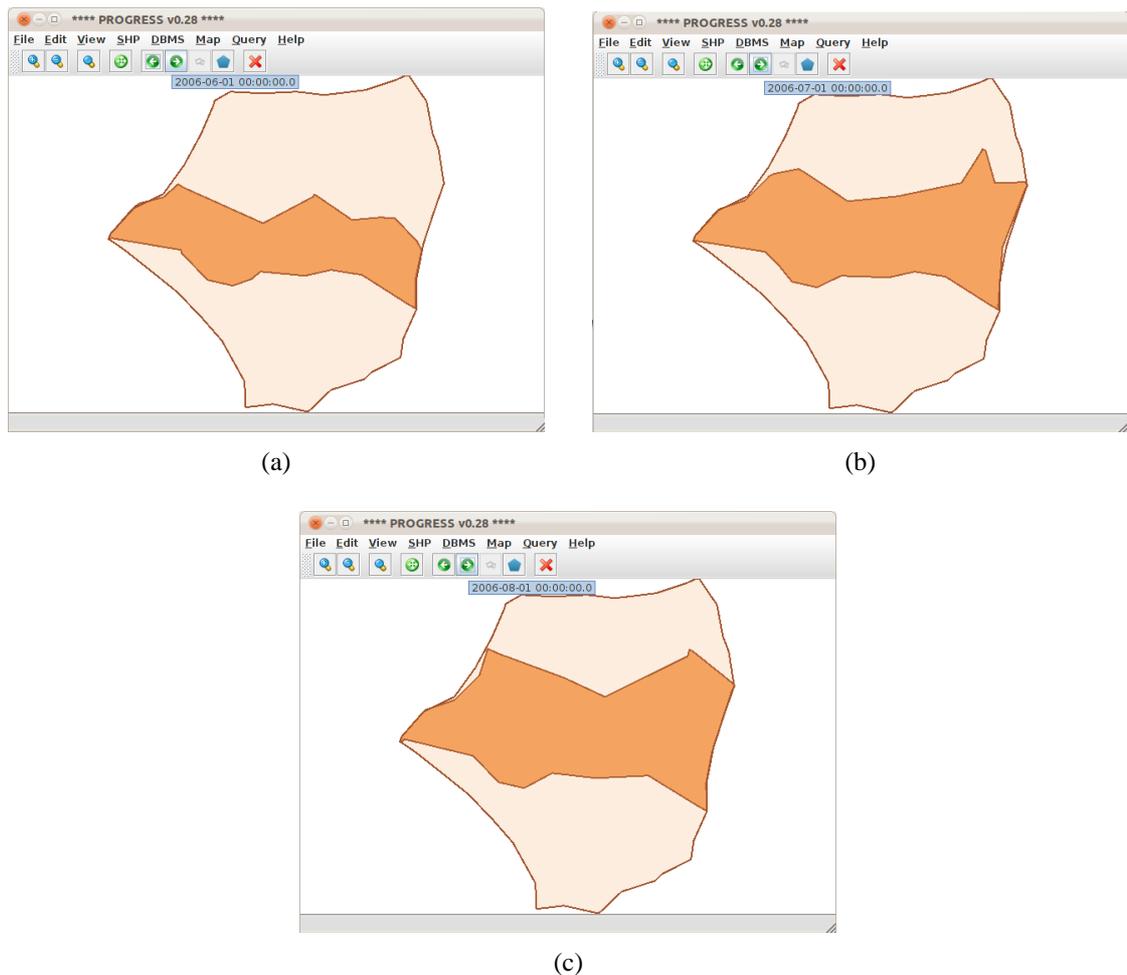


Figure 6.8: Extension of a particular feature (of type ‘ex-forest’) at 3 consecutive months, from which the activeness of a process (classified as ‘extending’) can be inferred on 2 distinct intervals.

Although the system outputs the correct answer for Query 4, this does not appear to be much help for exploring the case study considered here. Indeed, given its broadness, this query is not likely to be useful for most practical applications of the logical framework. Hence, further experiments discussed here will use queries containing at least one non-anonymous variable<sup>9</sup>.

Query 5 illustrates a scenario where the objective is to confirm whether a process is active at a particular time instant, without regard to the geographic features which participate in it. Query 5a is described in terms of the case study considered here, whereas

<sup>9</sup>Besides, of course, the constants already employed in Query 4 for distinguishing the active predicate, and for denoting the process classifier, the aggregation factor and the feature type (i.e., ‘on’, ‘expanding’, ‘0.1’, ‘4’).

5b and 5c present two different descriptions for this query employing the terminology adopted in the logical framework.

Q 5a *Was deforestation active in the Amazon on 15/04/2005 at 1am?*

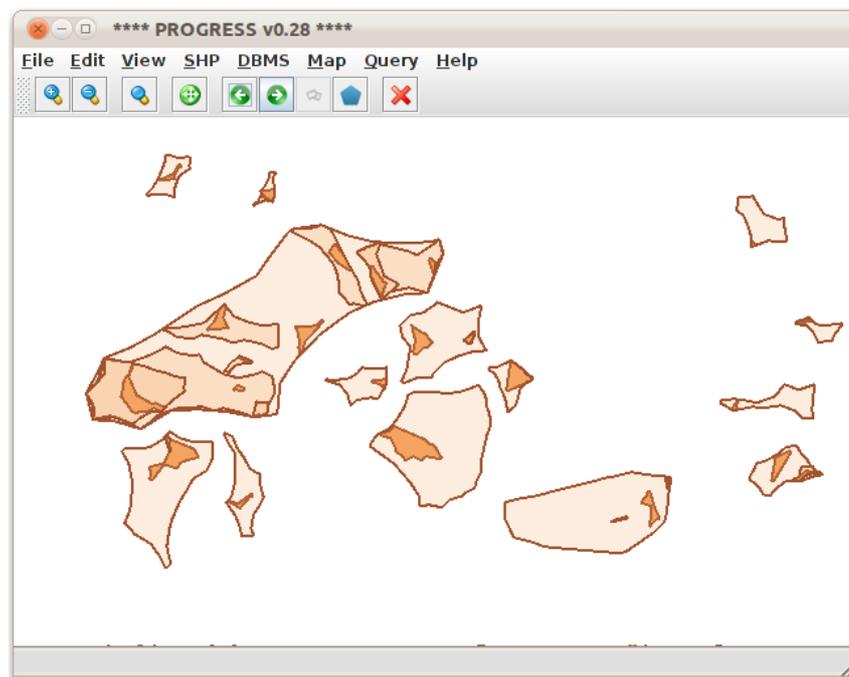
Q 5b *Was any geographic feature of type 'ex-forest' expanding on 15/04/2005 at 1am?*

Q 5c *Was a process active on 15/04/2005 at 1am?*

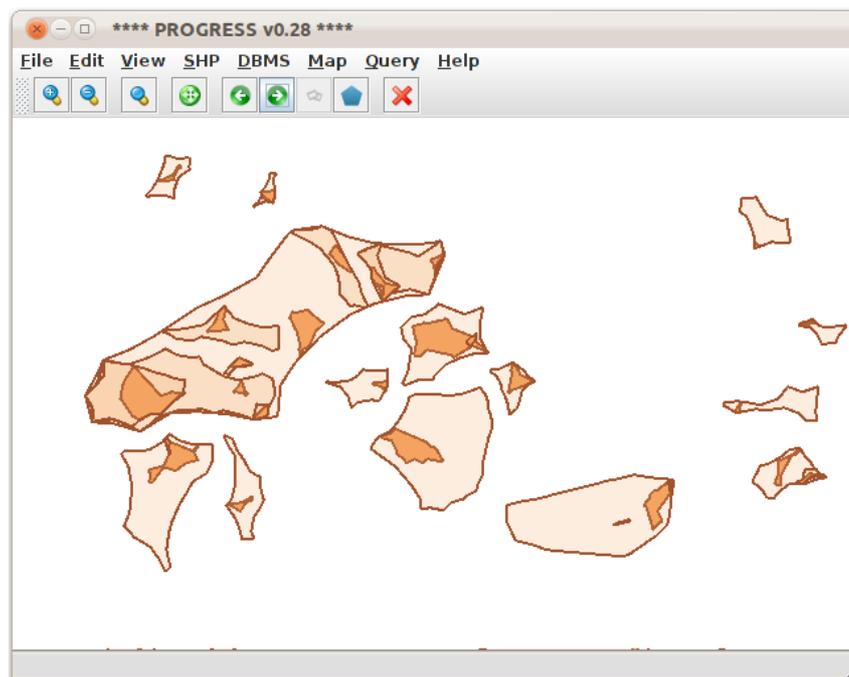
The way this query can be formulated in Progress is shown in Query 5d.

```
Q 5d ?- active(at, process(expanding, feature(0.1, 4, _)),
               ['2005-04-15 01:00:00']).
```

Similarly to Query 4, the answer provided by the system for Query 5 is just a statement 'true.', confirming that there is at least one feature for which the proposition. However, differently from the previous example, answers for 5 will depend on the time instant specified in the query. Figure 6.9 exemplifies how the appropriateness of this result can be assessed. In the example illustrated in this figure, 60 features (where aggregation factor is 0.1) have been loaded for visualisation. Then the spatial extensions of these features at different times can be examined to identify suitable test cases. Figures 6.9a and 6.9b show their extensions at 01/04/2005 00:00:00 and at 01/05/2011 00:00:00, respectively, where it can be seen that several features undergo change. Then Query 5 is formulated using a random time instant between those two other instants. In addition, other experiments have been carried out by first submitting the query containing a time instant chosen randomly, and then examining geographic features visually to confirm whether the results are adequate.



(a)



(b)

Figure 6.9: Spatial extensions of 60 different geographic features (of type 'ex-forest') at 2 consecutive months. It can be seen that some features change their spatial extensions.

In the example discussed for Query 5, a particular value has been specified for the time

instant variable, so that the objective of the query was to find any participant geographic feature which makes the proposition be interpreted as true for that particular time instant. Let us now present an example of query in which the aim is to check whether a process proceeds on any time interval, having a particular geographic feature as its participant. That is, now the variable representing a geographic feature is instantiated and the variable representing a time interval remains anonymous.

A textual description for this query following the case study about deforestation is given in Query 6a. This query is also described in Query 6b using the terminology employed in the logical framework. Notice that the former describes that a certain ‘area’ in the Amazon is deforested, whereas the latter describes affected areas in terms of geographic features.

*Q 6a Has deforestation proceeded in a particular area A in the Amazon until it reaches its culmination point?*

*Q 6b Has a (maximal) process proceeded (on any time interval) having the geographic feature f as its participant?*

For the purpose of carrying out experiments, a special predicate has been implemented for representing geographic features. This predicate contains an additional parameter corresponding to the numeric identifier (ID) used for indexing features at data level. This is useful for allowing a particular feature to be retrieved from the KB to conduct certain experiments, such as those carried out using Query 6.

The way Query 6 can be formulated in Progress is exemplified in Query 6c. In this example a particular geographic feature with an aggregation factor 0.3 has been examined. This feature is stored in the experimental DB using the ID 131, and therefore `feature(131,_,_,_)` has been employed to retrieve this particular feature.

Furthermore, a ‘2 months’ activeness threshold has been specified, which is represented by `time_threshold(0,2,0,0,0,0)`. Moreover, an anonymous variable ‘\_’ is used for time interval, so that the system returns `true` if there is at least one interval for which the proposition is true. In other examples, `Itv` is the variable used to replace such an anonymous variable in the query. Finally, the prefix `NO_` is used for naming the feature variable (`NO_F`). This commands the system to suppress the output of values bound to that variable, so that values bound to other variables of interest (e.g., `Itv`) can be easily visualised.

```
Q 6c ?- NO_F=feature(131,_,_,_), afeatures(NO_F),
      proceeds_max(on, time_threshold(0,2,0,0,0,0),
      process(expanding,NO_F), _).
```

Suitable test cases for the experiments of Query 6 have been obtained by executing a similar query where `Id` and `Itv` are given as free variables, so that values bound to them are returned by the system. Then, once values of `Id` and `Itv` are obtained, the corresponding features can also be examined in a visual manner using Progress's interactive map, in order to assess the reliability of results. Figure 6.10 gives an example of the execution of such a query, where the first results returned by the system are shown. Experiments using queries containing free variables will be discussed in detail in Sections 6.6.2.2 and 6.6.3.2.

```
PROGRESS Prolog Terminal - v0.17
Welcome to PROGRESS Prolog Terminal - v0.17
To redirect the output to map area, use the following prefixes:
- Geometry: G_
- Feature: FT_
- List of Features: LFT_
- Life Part: LP_
- No printing: NO_

?- NO_F1=feature(Id,0.3,4,NO_Life), afeatures(NO_F1),
NO_F2=feature(0.3,4,NO_Life),
proceeds_max(on, time_threshold(0,2,0,0,0,0),
  process(expanding,NO_F2), Itv).
Id = 131
Itv = ['2004-07-01 00:00:00','2006-11-01 00:00:00'];
Id = 131
Itv = ['2007-05-01 00:00:00','2007-10-01 00:00:00'];
Id = 131
Itv = ['2008-04-01 00:00:00','2008-12-01 00:00:00'];
Id = 131
Itv = ['2009-08-01 00:00:00','2010-03-01 00:00:00'];
Id = 131
Itv = ['2010-10-01 00:00:00','2010-11-01 00:00:00'];
Id = 132
Itv = ['2004-09-01 00:00:00','2005-03-01 00:00:00'];
Id = 133
Itv = ['2004-07-01 00:00:00','2004-09-01 00:00:00'];
Id = 133
Itv = ['2004-12-01 00:00:00','2005-01-01 00:00:00'];
Id = 134
Itv = ['2004-08-01 00:00:00','2005-01-01 00:00:00'];
Id = 135
Itv = ['2004-07-01 00:00:00','2004-10-01 00:00:00'];
Id = 136
Itv = ['2004-07-01 00:00:00','2004-10-01 00:00:00'];
?-
```

Figure 6.10: Executing a query at Progress Terminal to identify suitable geographic features and time intervals to be used in experiments for Query 6. Whilst variable `NO_F1` stores the feature representation containing its ID, `NO_F2` stores the representation required by other predicates (without ID). In addition, `afeatures` is an auxiliary function used to retrieve explicitly asserted features from the KB.

Furthermore, it should be highlighted that Query 6 can also be regarded as an auxiliary resource for evaluating the appropriateness of results produced by the query of Figure 6.10. That is, when answers for Query 6 do not match expected results, this might indicate a problem either in the mechanism for processing queries without free variables (e.g., Query 6) or in the mechanism for processing queries containing free variables (e.g., Figure 6.10).

### 6.6.2.2 Queries Containing Free Variables

From the examples given in Section 6.6.2.1, it can be seen that results obtained from queries containing no free variables are considerably limited, since the system only states whether query proposition is true or false without providing any information on values used for coming up with the solution(s). This section describes scenarios in which more elaborate queries are used to retrieve values corresponding to geographic features which participate in processes and/or temporal information (i.e., time instants and time intervals) associated with processes.

The following example (Query 7) illustrates a scenario in which one can identify deforested areas in the Amazon which are said to be expanding at a particular time instant. That is, it is possible to identify the areas affected by processes which are said to be active at a certain instant of time.

Different textual descriptions for this query are given in Queries 7a and 7b, as follows.

Q 7a *Where was deforestation active in the Amazon on 10/05/2005 at 05am ?*

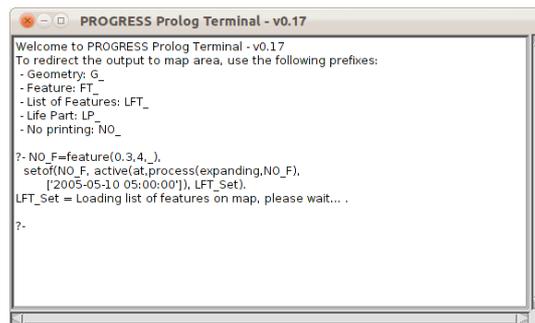
Q 7b *What geographic features have participated in a process which was active on 10/05/2005 at 05am ?*

This query can be specified in Progress as shown in Query 7c. In this query example, the aggregation factor 0.3 has been specified.

```
Q 7c NO_F=feature(0.3,4,_),
      setof(NO_F, active(at,process(expanding,NO_F),
                           ['2005-05-10 05:00:00']), LFT_Set).
```

Figure 6.11 shows the results obtained from executing Query 7 in Progress. Figure 6.11a exhibits the Progress Terminal, where the query is submitted using the format shown

in Query 7c. Figures 6.11b and 6.11c show the Progress map pane displaying the set of features that match the query. The value for this result set is bound to variable `LFT_Set`, which in turn corresponds to all values bound to variable `NO_F`. Figures 6.11b and 6.11c show the extensions of these features at 01/01/2005 00:00:00 and at 01/01/2006 00:00:00, respectively. In these figures, it can be seen that the spatial extensions of features expand between those time instants, which is in agreement with the query executed.



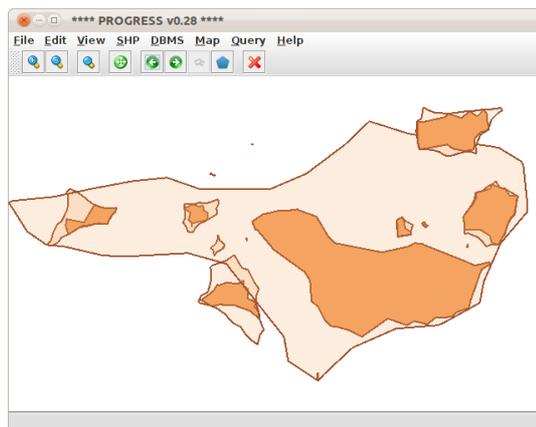
```

PROGRESS Prolog Terminal - v0.17
Welcome to PROGRESS Prolog Terminal - v0.17
To redirect the output to map area, use the following prefixes:
- Geometry: G_
- Feature: FT_
- List of Features: LFT_
- Life Part: LP_
- No printing: NO_

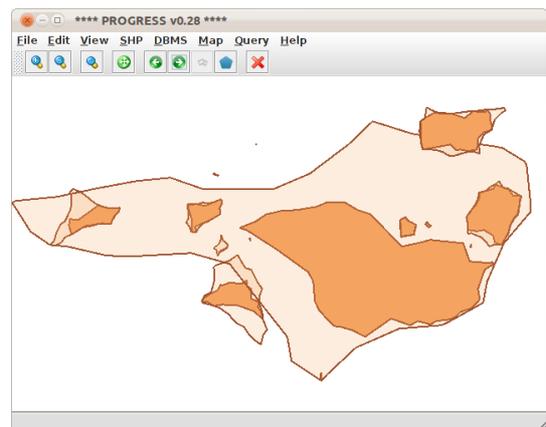
?- NO_F=feature(0.3.4._),
  setof(NO_F, active(at,process(expanding,NO_F),
    ['2005-05-10 05:00:00']), LFT_Set),
  LFT_Set = Loading list of features on map, please wait...
?-

```

(a)



(b)



(c)

Figure 6.11: Identifying features which participate in processes active at a given instant (Query 7). Figure (a) shows the query submitted to Progress' Terminal; Figures (b) and (c) shows the spatial extension of identified features at 01/01/2005 00:00:00 and 01/01/2006 00:00:00, respectively.

Query 8 is also employed to obtain information on processes' activeness. However, this is distinguished from previous queries as now the interval on which a process is said to be active is also returned by the system. Moreover, a temporal constraint is used in this query, so that only processes which are active on subintervals of a certain interval are considered. This query is textually described in Queries 8a and 8b.

Q 8a *Where and when was deforestation active between 01/01/2006 and 31/12/2006 ?*

Q 8b *What geographic features have participated in a process that was active on any subinterval of the time interval from 01/01/2006 00:00:00 to 01/01/2007 00:00:00?*

In Progress this query can be specified as shown in Query 8c.

```
Q 8c FT_F=feature(0.3,4,_),
      setof(Itv,
            (active(on,process(expanding,FT_F), Itv),
             int_in(Itv,
                   ['2006-01-01 00:00:00', '2007-01-01 00:00:00'])),
            Set_Itv).
```

After processing Query 8, Progress gradually displays the values bound to `Set_Itv` and `FT_F`, respectively, on the terminal and on the map pane. In this case, each different solution should be requested by the user by pressing semi-colon the ';' key. Figure 6.12 exhibits the first five solutions output for the query (involving 3 different geographic). These solutions contain, respectively: 2 sets of intervals over which the first feature is active (Figures (a) and (b)); 1 set of intervals over which the second feature is active (Figures (c) and (d)); and 2 sets of intervals over which the third feature is active (Figures (e) and (f)). From the examples shown in the Figure 6.12, it can be seen that the system groups the time intervals that relate to each particular feature matching the query, as a result of the use of the `setof` metapredicate.

Furthermore, the visual analysis of the output demonstrates the results are appropriate for the query submitted. This analysis is based on the visualisation of spatial extensions of displayed features at time instants in and out the output intervals. To illustrate, Figure 6.13 shows the spatial extensions (at two distinct time instants) of the 5 geographic features shown in Figure 6.12e. These time instants (01/05/2006 00:00:00 and 01/06/2006 00:00:00, respectively), denote the time interval `['2006-05-01 00:00:00', '2006-06-01 00:00:00']`, on which processes are active having 4 of these features as the participants. Thus, by comparing Figures 6.13a and 6.13b, it is possible to notice that these 4 participant features had their spatial extensions expanded in that period.



Figure 6.12: Results returned by the system for Query 8. Each different geographic feature which participates in an active process is shown on the map pane (Figures (a), (c) and (e)); and the set of intervals that match the query for a particular feature is shown at the terminal (Figures (b), (d) and (f)). These results correspond to the first five solutions output for the query, containing: 2 sets of intervals over which the first feature is active (Figures (a) and (b)); 1 set of intervals over which the second feature is active (Figures (c) and (d)); and 2 sets of intervals over which the third feature is active (Figures (e) and (f)).

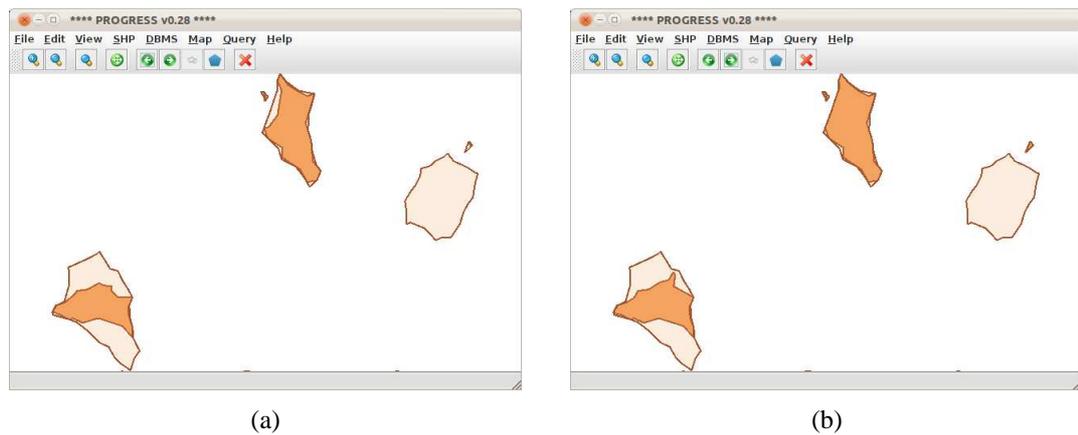


Figure 6.13: Spatial extensions of geographic features returned by the system for answering Query 8. These features extensions correspond to time instants 01/05/2006 00:00:00 and 01/06/2006 00:00:00, respectively. These instants, in turn, denote an interval on which 4 processes are active (according to Figure 6.12). By comparing Figures 6.13a and 6.13b, it is possible to notice that the 4 participant features had their extensions expanded in this period.

### 6.6.3 Queries about Events

This section describes the results of using the system to query event occurrences in different situations. As established in Section 6.6.2, in all examples presented here, geographic features are of type ‘ex-forest’, and ‘expands’ is a common classifier used to represent events.

#### 6.6.3.1 Queries Containing No Free Variables

Free variables have also been used for querying events. As described for queries about processes in Section 6.6.2.1, non-free variables comprise both instantiated variables and anonymous variables.

Query 9 illustrates a scenario in which the objective is to confirm whether an event occurred on a specific time interval, but without regard to the geographic features that participate in the event.

*Q 9a Did any deforestation event occur in the Amazon beginning on 01/07/2005 00:00:00 and ending on 01/09/2005 00:00:00?*

*Q 9b Did any event occur on the interval from 01/07/2005 00:00:00 to 01/09/2005*

00:00:00?

This query can be submitted to Progress as shown in Query 9c, where the activeness threshold has been set to zero and the aggregation factor has been set to 0.5.

```
Q 9c occurs(on, time_threshold(0,0,0,0,0,0),
           event(expands, feature(0.5,4,_)),
           ['2005-07-01 00:00:00', '2005-09-01 00:00:00']).
```

Appropriate scenarios for testing Query 9 can be produced in a similar manner as described for Query 6 (Section 6.6.2.1). That is, an auxiliary query containing free variables are used to identify geographic features and intervals associated with event occurrences. The output of this query is similar to the example shown in Figure 6.10. Then features and intervals identified are examined visually to ensure the reliability of results. Similarly to as noted in Section 6.6.2.1, the benefits of using both queries are reciprocal. That is, experiments using Query 9 may also be helpful for evaluating the mechanism for processing queries about event occurrences containing free variables.

### 6.6.3.2 Queries Containing Free Variables

Queries discussed in this section are also employed to obtain information on event occurrences. However, these queries are distinguished from those discussed in Section 6.6.3.1 as now free variables are employed to retrieve values corresponding to geographic features which participate in events and/or time intervals on which these events occur.

Query 10 illustrates a scenario where the aim is to identify the areas affected by deforestation events, without regard to the periods when these events occurred. This is described in Query 10a and 10b.

Q 10a *Where was Amazon deforested?*

Q 10b *What geographic features of type 'ex-forest' participated in events whose classifiers are 'expands'?*

The specification of this query in the form it should be submitted to Progress is shown in Query 10c. In this scenario, only events without temporal gaps are of interest, and therefore the activeness threshold `time_threshold(0,0,0,0,0,0)` is given. More-

over, the aggregation factor 0.5 has been specified for representing geographic features.

```
Q 10c  F=feature(0.5,4,_),
        setof(F, Itv^(occurs(on, time_threshold(0,0,0,0,0,0),
        event(expands,F),Itv)), LFT_Set).
```

Although the system returns appropriate results for Query 10c, it requires a significant amount of time to be processed. The reason for this inefficiency is that Progress unnecessarily examines the whole life of every geographic feature in the KB. By way of explanation, it should be noted that, in this query, the intervals on which events occur are not relevant. Therefore the system should examine the life of a given feature only until it finds any life part that matches the query. Then it should jump to the next feature in the KB that has not yet been checked.

Hence, an alternative approach for this query should use some artifice (such as the cut ‘!’ operator) to make the predicate `occurs` jump to another feature once a value for `Itv` which satisfies the query is found for the current feature. However, if the cut is placed just after `occurs`, it makes the system abort the execution just after finding the first value for `Itv`, without checking other features.

An alternative specification for this query that can improve drastically its processing time is shown in Query 10d. In this alternative approach, the `findall` metapredicate is used to retrieve all features from the KB, so that instantiated values of features can be passed as an argument to `occurs` predicate, which is evaluated independently for each distinct feature. In addition, an auxiliary predicate `succeeds` is employed. This predicate makes `occurs` abort when it finds the first solution. This predicate is implemented as follows:

```
succeeds(Goal) :- call(Goal), !.
```

The major drawback of using artifices such as those employed in Query 10d is the fact that the query specification becomes more distinct from the equivalent specification in pure logic. However, from the experiments conducted here, it can be concluded that these kinds of manipulations of the logical language are essential for practical purposes. Consequently, they could potentially be incorporated into the language.

```
Q 10d  findall(F, (F=feature(0.5,4,_), afeatures(F)), NO_L),
        setof(Ft, Itv^(member(Ft,NO_L),
        succeeds(occurs(on,time_threshold(0,0,0,0,0,0),
```

```
event(expands, Ft), Itv))), LFT_Set).
```

The output provided by Progress for Query 10 is shown in Figure 6.14. This output consists of geographic features which participate in events. In this example, a static map layer containing the Amazon boundaries is also displayed (on the background). In Figure 6.14a, results are displayed in mode ‘hulls’, whilst in Figure 6.14b results are exhibited in mode ‘navigation’ (where opaque polygons correspond to the earliest time instant in the results).

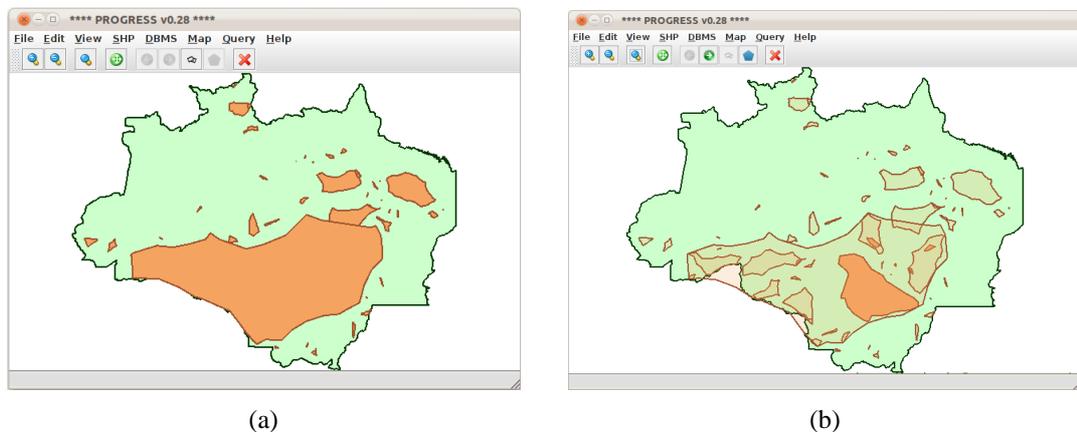


Figure 6.14: System’s answer for Query 10. The output consists of geographic features which participate in events. In (a), the results are shown in mode ‘hulls’; whereas in (b) result are exhibited in mode ‘navigation’.

The suitability of the system for answering queries about events and processes represented using different activeness threshold has been examined. An experiment is now discussed using Query 11, which aims to get information on geographic features and intervals associated with event occurrences. To illustrate, results obtained from submitting this query to the system using 2 distinct activeness thresholds – *2 months* and *zero* are presented.

Q 11a *Where and when was Amazon deforested?*

This query can submitted to Progress as shown below, where the activeness threshold *2 months* has been set.

```
Q 11b  ?- NO_F1=feature(Id,0.3,4,NO_Life),afeatures(NO_F1),
NO_F2=feature(0.3,4,NO_Life),
occurs(on, time_threshold(0,2,0,0,0,0),
       event(expands,NO_F2), Itv).
```

The specification of this query using the threshold *zero* is as follows.

```
Q 11c  ?- NO_F1=feature(Id,0.3,4,NO_Life),afeatures(NO_F1),
NO_F2=feature(0.3,4,NO_Life),
occurs(on, time_threshold(0,0,0,0,0,0),
       event(expands,NO_F2), Itv).
```

The first 18 results returned by Progress to Query 11b are shown in Figure 6.15. Each solution for this query consists of a value bound to variable *Id*, representing the numeric identifier for a particular feature; and a value bound to variable *Itv*, representing the interval on which an event occurs. For example, this result shows that, for feature with ID=131, 5 different events of the same type have occurred. Furthermore, it can be seen that the temporal distance between these event tokens range between 6 to 8 months, whilst the activeness threshold defined in this query was 2 months. This result means that the same area in the Amazon has been deforested during 5 different periods, separated from each other by at least half a year. These different events might indicate, for example, that this area has been deforested for the same purpose (e.g., illegal logging) but by distinct groups of people or organisations.

```
Id = 131
Itv = ['2004-07-01 00:00:00', '2006-11-01 00:00:00'];
Id = 131
Itv = ['2007-05-01 00:00:00', '2007-10-01 00:00:00'];
Id = 131
Itv = ['2008-04-01 00:00:00', '2008-12-01 00:00:00'];
Id = 131
Itv = ['2009-08-01 00:00:00', '2010-03-01 00:00:00'];
Id = 131
Itv = ['2010-10-01 00:00:00', '2010-11-01 00:00:00'];
Id = 132
Itv = ['2004-09-01 00:00:00', '2005-03-01 00:00:00'];
Id = 133
Itv = ['2004-07-01 00:00:00', '2004-09-01 00:00:00'];
Id = 133
Itv = ['2004-12-01 00:00:00', '2005-01-01 00:00:00'];
Id = 134
Itv = ['2004-08-01 00:00:00', '2005-01-01 00:00:00'];
Id = 135
Itv = ['2004-07-01 00:00:00', '2004-10-01 00:00:00'];
Id = 136
Itv = ['2004-07-01 00:00:00', '2004-10-01 00:00:00'];
Id = 136
Itv = ['2005-07-01 00:00:00', '2005-10-01 00:00:00'];
Id = 136
Itv = ['2006-05-01 00:00:00', '2006-06-01 00:00:00'];
Id = 136
Itv = ['2007-05-01 00:00:00', '2007-06-01 00:00:00'];
Id = 137
Itv = ['2004-07-01 00:00:00', '2004-09-01 00:00:00'];
Id = 138
Itv = ['2004-10-01 00:00:00', '2004-11-01 00:00:00'];
Id = 139
Itv = ['2004-07-01 00:00:00', '2004-09-01 00:00:00'];
Id = 141
Itv = ['2004-08-01 00:00:00', '2004-12-01 00:00:00'].
```

Figure 6.15: First 18 results returned by Progress to Query 11b.

Figure 6.16 shows the extension of feature 131 at 6 different time instants corresponding to the second result of Query 11b (shown in Figure 6.15), where the value of the variable `Itv` is [ '2007-05-01 00:00:00', '2007-10-01 00:00:00' ]. From this illustration, it can be seen that the feature remained unchanged for a certain period (i.e. interval [ '2007-08-01 00:00:00', '2007-09-01 00:00:00' ]); however, since this period is shorter than the activeness threshold of 2 months, a single event token has been identified (having feature 131 as the participant).

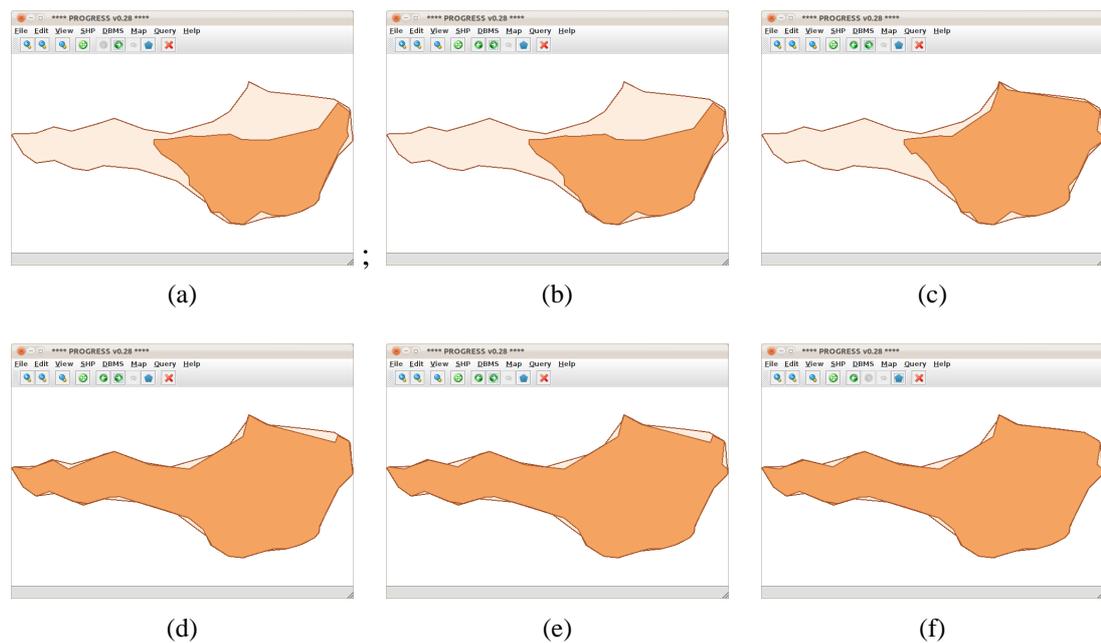


Figure 6.16: Extension of feature 131 at 6 different time instants corresponding to the second result of Query 11b, shown in Figure 6.15. These time instants are, respectively, 01/05/2007, 01/06/2007, 01/07/2007, 01/08/2007, 01/09/2007, 01/10/2007 (all of these at 00:00:00).

The result shown in Figure 6.16 is produced by using Query 12. This query commands the system to show only the life-part of the feature that matches the interval on which the event of interest occur. First the appropriate interval is caught by `begins`; then `feature_lp` is used to determine the value of the variable `LP_1`, corresponding to the life-part of the feature represented by variable `NO_F2` that matches the interval represented by

variable `Itv`.<sup>10</sup>

```
Q 12 ?- NO_F1=feature(Id,0.3,4,NO_Life),afeatures(NO_F1),
      NO_F2=feature(0.3,4,NO_Life),
      occurs(on, time_threshold(0,2,0,0,0,0),
      event(expands,NO_F2), Itv),
      begins(['2007-05-01 00:00:00'], Itv),
      feature_lp(NO_F2, Itv, LP_1).
```

The first 18 results returned by Progress for Query 11c is shown in Figure 6.17. This query has the same aim as Query 11b; however, in Query 11c, the activeness threshold has been set to *zero*. Comparing these results with those shown Figure 6.15, it can be noticed that now events have shorter durations (1.72 in average), and most of them are 1 or 2 months far apart in time.

---

<sup>10</sup>In this case, the prefix `LP_` has been used to instruct the system to output the value of this variable on the map pane; whereas the prefix `NO_1` commands the system to suppress the value of these variables from the output.

```
Id = 131
Itv = ['2004-07-01 00:00:00', '2004-10-01 00:00:00'];
Id = 131
Itv = ['2004-11-01 00:00:00', '2004-12-01 00:00:00'];
Id = 131
Itv = ['2005-02-01 00:00:00', '2005-06-01 00:00:00'];
Id = 131
Itv = ['2005-07-01 00:00:00', '2005-09-01 00:00:00'];
Id = 131
Itv = ['2005-10-01 00:00:00', '2005-11-01 00:00:00'];
Id = 131
Itv = ['2006-01-01 00:00:00', '2006-02-01 00:00:00'];
Id = 131
Itv = ['2006-03-01 00:00:00', '2006-04-01 00:00:00'];
Id = 131
Itv = ['2006-05-01 00:00:00', '2006-06-01 00:00:00'];
Id = 131
Itv = ['2006-07-01 00:00:00', '2006-11-01 00:00:00'];
Id = 131
Itv = ['2007-05-01 00:00:00', '2007-08-01 00:00:00'];
Id = 131
Itv = ['2007-09-01 00:00:00', '2007-10-01 00:00:00'];
Id = 131
Itv = ['2008-04-01 00:00:00', '2008-07-01 00:00:00'];
Id = 131
Itv = ['2008-08-01 00:00:00', '2008-09-01 00:00:00'];
Id = 131
Itv = ['2008-11-01 00:00:00', '2008-12-01 00:00:00'];
Id = 131
Itv = ['2009-08-01 00:00:00', '2009-09-01 00:00:00'];
Id = 131
Itv = ['2009-11-01 00:00:00', '2009-12-01 00:00:00'];
Id = 131
Itv = ['2010-02-01 00:00:00', '2010-03-01 00:00:00'];
Id = 131
Itv = ['2010-10-01 00:00:00', '2010-11-01 00:00:00'];
```

Figure 6.17: First 18 results returned by Progress to Query 11c

## 6.7 Overall Analysis and Conclusion

The aim of this chapter was to illustrate a number of different scenarios where Progress can be used to discuss the results obtained from using the system with spatio-temporal data, in order to assess the general applicability of the proposed theory. Once Progress has been evaluated and different results have been presented, the overall effectiveness of the system can be considered, to determine what Progress is successful at and where it is deficient. The success of Progress is measured in terms of how closely the results matched the expected results.

According to the results obtained, Progress was successful at deriving implicit *Stars* and at individuating geographic features, when compared to the evaluation criteria. Moreover, Progress was successful at identifying event occurrences, process activeness and process which are said to proceed. In addition, the system appeared to be suitable for answering logical queries about these entities using different standpoint semantics thresholds.

Nevertheless, the extent to which it was successful and matched the expected results varied between different test scenarios. There were a few cases in which inappropriate results were obtained for demarcating the spatial extensions of geographic features, but which were associated with the improper output of the concave hull function used. However, this function is provided by an external source and can be replaced straightforwardly by another one that best fits the system requirements. Hence this issue in no way discredits the success of Progress at implementing the proposed theory.

The effectiveness of the system in terms of execution time may also vary considerably between different test scenarios. As noted in Section 6.5.3, major impacts might be observed when *Stars* are frequently partially ‘clipped’, which requires the system to compute costly spatial operations between polygons. In addition, as noted in Section 6.6.3.2, it was noted that logical queries submitted to Progress sometimes must contain a number of programming artifices to gain efficiency, which makes queries more dissimilar to their equivalent formulation in pure first-order logic.

This chapter presented a variety of queries for obtaining information on events and processes. However, further queries could also be developed of a similar nature to those described here. Such queries might include, for instance, a variety of other spatial and temporal constraints, using relations from RCC and Allen’s Algebra.

A great challenge for developing the work of this thesis was to find a dataset containing real-world spatio-temporal data which could be applied to evaluate all important aspects of the work. As described in Section 6.3, experiments using Progress were con-

ducted with a dataset containing data about the phenomenon of deforestation in the Amazon rainforest in a 7-year period (between June 2004 and May 2011). The dataset used in this case study consists of distinct sets of polygons, each of which representing regions deforested in Brazilian Amazon in a particular calendar month. This 84-month period dataset contains a total of 47,459 polygons.

Unfortunately the dataset obtained is limited to a single type of coverage (i.e., denoting that a certain region is deforested), and therefore it was insufficient for evaluating the adequacy of the system for performing all the inferences described in Chapter 3, specially the derivation rules which relates homogeneous and heterogeneous coverage attributes. However, it should be noticed that this does not affect the evaluation of the mechanism for modelling geographic features' lives, as it only manipulates geometries representing maximal extents of certain coverages (at distinct times), and therefore the points evaluated do not depend on the fact that the extent of a feature is originated from a homogeneous or a heterogeneous region. Furthermore, given the large number of polygons contained in that dataset, it appeared to be adequate for conducting the necessary experiments to evaluate the logical framework presented in Chapter 4. With this dataset, it was possible to consider different test scenarios, where events and processes could be identified under many different circumstances.

To deal with the lack of real-world spatio-temporal containing heterogenous regions, additional synthetic data were produced to simulate different test scenarios. Although this data contains reduced number of polygons and shorter temporal series, it includes various different attributes. Therefore this data appeared to be adequate to evaluate this particular portion of the system. Some experiments conducted based on synthetic data were described in Section 6.5.1.

## 6.8 Summary

This chapter discussed the experiments conducted using Progress system prototype with temporal series of topographic data. The system effectiveness for deriving implicit *Stars* and for individuating geographic features was evaluated using a large dataset – containing a single attribute type; and using a reduced synthetic dataset comprising many distinct attributes which are interrelated in different forms. Then the general suitability of the system for answering different types of logical queries has been evaluated. Results obtained from these experiments were examined individually, before considering the results overall.

# Chapter 7

## Conclusions and Future Work

---

### 7.1 Thesis Overview and Contributions

The final goal of this research was to develop a knowledge representation approach to identifying geographic events and processes in temporal series of topographic data. Of particular interest in this work were events and processes which can be represented in terms of spatial transformations affecting the spatial extensions of geographic features. To achieve this goal, a formalism for representing events and processes has been developed, as well as methods of representing the spatio-temporal data and to explicitly link the formal framework to the data. Moreover, a system has been implemented to evaluate the applicability of the proposed theory.

Chapter 3 presented a logic-based approach to representing temporal topographic data that allows implicit data to be derived by means of logical inference. The chapter also described an approach to representing changing geographic features based on primitive elements of data. The approaches presented in Chapter 3 represent an important contribution to different fields of research, such as GIScience and Knowledge Representation.

First, the proposed approach to representing spatio-temporal data can be used as a mechanism for supporting the integration of spatio-temporal data originating from heterogeneous sources. Such an integration task involves a number of challenges which can be easily overcome with the proposed mechanism. These difficulties include issues on spatial, temporal, and thematic dimensions. Although distinct datasets may describe

different things about different portions of geographic space, in distinct periods of time, these data are often complementary in many aspects. Since the proposed approach is provided in the form of a framework, such a complementariness can be represented within the framework in the form of logical relations between thematic descriptions and the regions which can potentially be associated with them (e.g., ‘a forested region must contain a subregion covered by vegetation  $a_1$ ’). Then, once elementary knowledge about the domain is encoded in the system to form the initial KB, many different inferences can be performed to generate a variety of implicit data.

Moreover, the approach presented in Chapter 3 represents a relevant contribution to research on methods of ontology grounding. A characteristic of ontologies of geographic domain is that they are likely to contain concepts which can be effectively grounded upon data. In this work, ontology grounding is considered not only as a method of linking primitive symbols to elements of data, as seen in previous work. The grounding layer proposed here also allows high level concepts to be defined in terms of the primitive ones, that is, without concerns about the data structure. Such a grounding mechanism proposed here is based on an approach to representing geographic features, which can act as an abstraction layer to allow other conceptual elements of dynamic geographic space (e.g., events and processes) to be defined in a high level of abstraction.

In Chapter 4, a framework for representing and reasoning about geographic events and processes was presented. The representation of events and processes is still the subject of considerable controversy in the literature. For this reason, some previous work avoided providing precise definitions for certain concepts. Conversely, the proposed formalism includes a number of precise logical definitions, with the aim of applying this for processing real topographic data. The framework provides a method of handling spatial and temporal vagueness based on standpoint semantics [10]. By incorporating a vague reasoning approach into the reasoning stage, it was hoped to allow different event and process instances to be determined based on individual viewpoints, which therefore enables the framework to be applied to a broad range of situations.

Together with the mechanisms described in Chapter 3 for modelling spatio-temporal data and for representing geographic features, it is hoped that the framework presented in Chapter 4 can provide an improved method of querying spatio-temporal data. Researchers in Geographic Information Science (GIScience) have investigated means of providing more conceptualised methods of manipulating and querying spatio-temporal data. Recent developments include conceptual models for spatio-temporal data (e.g., [65]), which are frequently described using the entity-relationship model (ER) and Unified Modelling Language (UML). However, despite their expressiveness for describing real-world enti-

ties, they lack in providing a method of linking the conceptual and data layers so that reasoning is allowed on spatio-temporal data. Object-oriented approaches have also become of interest (e.g., [97]), since they can provide a model which is both concrete (i.e., implemented in software) and described in a more conceptualised fashion. Nonetheless, inference capabilities of these models are still limited, and consequently queries tend to become more complex and less expressive.

To evaluate the applicability of the proposed theory to process real topographic data, a system prototype was developed, named Progress. This system, presented in Chapter 5, takes temporal series of topographic data as an input, and, through first order logic querying, allows information on event occurrences and process activity to be identified within the data. Although some previous work in the field of geographic/spatial knowledge representation provide important directions, most of them are not yet implemented, and therefore their suitability for handling real-world data is not often discussed. Hence, implementing a system which applies the proposed theory represents a significant contribution to the field of research, as it requires dealing with a variety of issues which are often ignored.

The results of using the system with spatio-temporal data were discussed in Chapter 6. These results indicated the overall success of Progress at deriving implicit data and at individuating geographic features. In addition obtained results indicated the system was successful at answering logical queries about events occurrences and process activities considering different standpoint semantics thresholds.

## 7.2 Discussion and Future Work

In addition to the achievements of this work, it is important to consider its limitations, and to determine where further improvements might be required. The work within this thesis could be extended in many directions; thus, looking at these limitations will help evaluate the extensions that would be relevant.

The work presented in this thesis placed particular focus on the representation of geographic phenomena which can be described in terms of changing spatial extensions of geographic features. Moreover, the *kinds of geographic features* that can be represented are limited to those features whose extension at a particular time can be defined as a 2-dimensional polygon corresponding to some portion of the earth's surface. Clearly, these geographic phenomena represent only a small portion of the geographic domain. Therefore, the approaches developed here may not be applicable to represent other *kinds of phenomena* within the domain. Other kinds of phenomena include, for instance, those

addressed by Devaraju and Kuhn [27], where a process is regarded as having physical objects and substances as their participants. The representation of those phenomena involves considering certain physical and chemical transformations that might occur amongst events and process participants. An example of a phenomenon of this nature is ‘evapotranspiration’.

Moreover, there are other kinds of phenomena that can be described in terms of changes of values of attributes observed for a particular region, without regard to changes in the spatial extension of the region. The representation of these kinds of phenomena might be based, for example, on geo-sensors applied to a particular region whose spatial extension is static over time. An example of geographic phenomena of this nature is described by Kulik et al. [56], in which vegetation modification events are examined. The logical framework presented in this thesis can potentially be extended to deal with other kinds of phenomena, without much modification to most of its formal apparatus, including the approach to modelling temporal aspects of events and processes, to determining the relationships between them and the method of handling temporal vagueness.

Although this work concentrates on the representation of geographic features whose spatial extensions are subject to change over time, the focus was not placed on the development of a logical language which can represent many different *spatial changes* that might affect these features. Rather, this thesis presents logical definitions of some spatial changes to illustrate how they can be specified within the framework; and then it explores one of them (i.e., expansion) to carry out experiments using the system prototype. Spatial changes affecting 2-dimensional polygons have already been extensively discussed in the literature. Therefore, in this work, efforts have been directed to design the framework in such a way that additional spatial changes can be defined with no impact to the rest of the semantics. Therefore, an extension of this work would be to provide definitions for a larger number of spatial changes which may affect spatial extensions of features. Other changes that might be included are, for instance, deformation and rotation, as suggested by Claramunt et al. [21].

Amongst the most important limitations of this work is the restricted variety of relationships between events and processes. The framework presented in this thesis only provides a way to represent events as chunks of processes, and to represent processes in terms of their constituent events. Further expansions to the logical framework presented in this thesis could incorporate *other relationships* between these concepts. Several relationships that could potentially be incorporated to the framework are described by Galton [36] (e.g., transitions, repetition, composition, specification).

Desired capabilities which are not present in this framework include a method of rep-

representing 1 to n relationships between different event and process classifiers. For example, an event could be determined by a chunk of two different process that proceed in parallel. Methods of specifying relationships between different event classifiers and between distinct process classifiers are also desired. Moreover, even more complex situations could be represented by incorporating methods of modelling *relationship patterns* between events and processes, similar to as developed within the semantic formalism proposed by Claramunt and Thériault [20], which incorporates the Event Pattern Language (EPL) [42, 43] to model changing elements of geographic space. Using such a kind of language, occurrences of an event associated with a certain classifier can be identified by matching patterns of occurrences of events associated with other classifiers. In these languages, event/process patterns are specified using expressions which resembles regular expressions<sup>1</sup>. However, additional capabilities to represent certain temporal aspects (e.g., duration) are still the subject of further investigation.

In this thesis, a *process* is regarded as an entity which is subject to change over time. However, the approach to representing these changes is considerably limited in the framework proposed in this work (i.e., the representation of process change is mostly based on the concept of process activeness). Further expansions to this work could therefore consider a number of different *properties* that could be ascribed to processes. For example, a process may be described as being constant, or intermittent, or slowing down, or accelerating. The representation of these changes requires dealing with different kinds of vagueness, and standpoint semantics appears to be applicable to most situations. The incorporation of an improved representation of process properties (together with the provision of methods of specifying relationship patterns between events and processes, described above) would make the logical framework an important resource for the development of theories of *causality* for geographic phenomena. For example, as described by Kulik et al. [56], deforestation caused by different agents leads to different *impacts* on the vegetation. Therefore, if the cause (i.e., origin) of a phenomenon is unknown, it might be inferred by analysing its impact on geographic space.

The approaches described in Chapter 3 to modelling spatio-temporal data and representing geographic features have some limitations and therefore could be improved in several forms. *First*, the *geometric representation* of *Stars* is restricted to 2-dimensional polygons. Therefore, the model could be improved to allow the representation of other geometric types (e.g., points and lines), as well as to enable 3-dimensional representation of space.

Another limitation within the approach to representing spatio-temporal data is that the

---

<sup>1</sup>Regular Expressions - [http://en.wikipedia.org/wiki/Regular\\_expression](http://en.wikipedia.org/wiki/Regular_expression)

*representation of compound geographic features* is currently determined by part-of relations which may hold between homogeneous coverage attributes and *one* heterogeneous attribute. Although this is sufficient for representing many different types of geographic features, this could be improved to represent more complex scenarios. As discussed earlier, features are maximal well-connected extents of their corresponding coverage attribute; however, given the limitation of this model, no other feature can be proper part of this (neither of different type or of the same type). Therefore the system does not allow, for example, a city to be part of an island (i.e., a maximal chunk of urbanisation to be part of a maximal chunk of land). Thus a potential enhancement would be to allow features to be represented based on a *multiple-level attribute hierarchy*, where geographic features could contain other features of different types. The current version of the proposed framework is based on a polymorphic relation  $CC$ , which relates a pair of attributes of different types. Observe that it works well for the reduced variety of scenarios which can currently be represented. However, for representing more complex scenarios, the use of distinct relations with different properties would become essential (such as  $CP_{cc}$ ,  $CP_{cf}$ , and  $CP_{ff}$ , relating, respectively, a pair of coverage attributes, a coverage attribute and a feature attribute, and a pair of feature attributes).

Moreover, an improvement to the approach to representing geographic features would affect the method of inferring the type and the spatial extension of geographic features. It might incorporate other existing approaches to handling *spatial vagueness*. For example, Bennett et al. [12, 13] proposes a method of handling vagueness in which the geographic feature type can be inferred based on different geometric characteristics (e.g., a water body can be classified as ‘river’ or ‘lake’ depending upon its level of ‘linearity’). More crucially, a more complex representation of a *feature life* should be developed. The proposed model is significantly limited in this aspect, and consequently is not capable of representing effectively with splits, merges, and trajectories affecting features. It should be observed that this affects directly the interpretation of the identity of events and processes. For example, consider two disconnected regions which undergo urbanisation, characterising two distinct processes going on. Then, at a certain time, these urbanised regions get connected to each other, therefore characterising a single process going on for the whole region.

### 7.3 Conclusion

This chapter has summarised the achievements and limitations of the work presented in this thesis, as well as considered future expansions. It is hoped that this work can represent

a significant step towards a more concrete integration between semantic-based formalisms and real-world applications in GIS. It is also hoped that the formalisms and the system developed in this work can act as a basis for future expansion, to further improve the representation of geographic phenomena within GIS.

# Bibliography

- [1] J.F. Allen. Maintaining knowledge about temporal intervals. *Communications of the ACM*, 26(11):832–843, 1983.
- [2] J.F. Allen. Towards a general theory of action and time. *Artificial intelligence*, 23(2):123–154, 1984.
- [3] Karl-Heinrich Anders. A hierarchical graph-clustering approach to find groups of objects. In *Proceedings 5th Workshop on Progress in Automated Map Generalization*, pages 1–8. Citeseer, 2003.
- [4] Grigoris Antoniou. A tutorial on default logics. *ACM Computing Surveys (CSUR)*, 31(4):337–359, 1999.
- [5] Chris Bailey-Kellogg and Feng Zhao. Qualitative spatial reasoning extracting and reasoning with spatial aggregates. *AI Magazine*, 24(4):47–60, 2003.
- [6] B. Bennett and A.P. Galton. A unifying semantics for time and events. *Artificial Intelligence*, 153(1-2):13–48, 2004.
- [7] Brandon Bennett. A categorical axiomatisation of region-based geometry. *Fundamenta Informaticae*, 46(1):145–158, 2001.
- [8] Brandon Bennett. Space, time, matter and things. In *Proceedings of the international conference on Formal Ontology in Information Systems-Volume 2001*, pages 105–116. ACM, 2001.
- [9] Brandon Bennett. Spatial vagueness. In Robert Jeansoulin, Odile Papini, Henri Prade, and Steven Schockaert, editors, *Methods for Handling Imperfect Spatial Information*. Springer, 2010.
- [10] Brandon Bennett. Standpoint semantics: a framework for formalising the variable meaning of vague terms. In Petr Cintula, Christian Fermuller, Lluís Godo, and Petr

- Hajek, editors, *Understanding Vagueness — Logical, Philosophical and Linguistic Perspectives*. College Publications, 2011.
- [11] Brandon Bennett, Anthony G. Cohn, Paolo Torrini, and Shyamanta M. Hazarika. A foundation for region-based qualitative geometry. In *Proceedings of the European Conference on Artificial Intelligence (ECAI-2000)*, pages 204–208, 2000.
- [12] Brandon Bennett, David Mallenby, and Allan Third. Automatic grounding of vague geographic ontology in data. In *GeoComputaton, Proceedings of the second international conference*. Online, 2007.
- [13] Brandon Bennett, David Mallenby, and Allan Third. An ontology for grounding vague geographic terms. In *Proceedings of the International Conference on Formal Ontology in Information Systems (FOIS 2008)*, pages 280–293. IOS Press, 2008.
- [14] Paul Biron and Ashok Malhotra. XML schema part 2: Datatypes second edition: Xml schema part 2: W3c recommendation 28 october 2004. <http://www.w3.org/TR/xmlschema-2/>, 2004. Accessed on 22/05/2013.
- [15] T. Bittner and B. Smith. Formal ontologies of space and time. *IFOMIS, Department of Philosophy, Leipzig, Buffalo, University of Leipzig, University at Buffalo and NCGIA*, 17, 2003.
- [16] T. Bittner, M. Donnelly, and B. Smith. A spatio-temporal ontology for geographic information integration. *International Journal of Geographical Information Science*, 23(6):765–798, 2009.
- [17] Elena Camossi, Michela Bertolotto, and Elisa Bertino. Multigranular spatio-temporal models: implementation challenges. In *Proceedings of the 16th ACM SIGSPATIAL international conference on Advances in geographic information systems*, page 63. ACM, 2008.
- [18] C. Claramunt, M. Theriault, and C. Parent. A qualitative representation of evolving spatial entities in two-dimensional spaces. *Innovations in GIS V*, pages 119–129, 1997.
- [19] Christophe Claramunt and Marius Thériault. Managing time in GIS an event-oriented approach. In *Recent Advances in Temporal Databases*, pages 23–42. Springer, 1995.

- [20] Christophe Claramunt and Marius Thériault. Toward semantics for modelling spatio-temporal processes within GIS. *Advances in GIS Research I*, pages 27–43, 1996.
- [21] Christophe Claramunt, C. Parent, and Marius Thériault. Design patterns for spatiotemporal processes. *Searching for Semantics: Data Mining, Reverse Engineering*, pages 415–428, 1997.
- [22] Bowman L Clarke. A calculus of individuals based on connection. *Notre Dame Journal of Formal Logic Notre-Dame, Ind.*, 22(3):204–219, 1981.
- [23] A. G Cohn and S. M Hazarika. Spatio-Temporal continuity in geographic space. In M. Duckham and M. F. Worboys, editors, *Meeting on Fundamental Questions in Geographical Information Science*, pages 1–26. University of Leeds, 2001.
- [24] A G Cohn, B Bennett, J Gooday, and N Gotts. RCC: a calculus for region-based qualitative spatial reasoning. *GeoInformatica*, 1:275–316, 1997.
- [25] Anthony Cohn, Brandon Bennett, John Gooday, and Nicholas Gotts. Representing and reasoning with qualitative spatial relations about regions. *Spatial and temporal reasoning*, pages 97–134, 1997.
- [26] Peter Leslie Davies. *Prolog and expert systems*. PhD thesis, Durham University, 1987.
- [27] Anusuriya Devaraju and Werner Kuhn. A Process-Centric ontological approach for integrating Geo-Sensor data. In *Proceedings of the International Conference on Formal Ontology in Information Systems (FOIS 2010)*, 2010.
- [28] Pasquale Di Donato. Geospatial semantics: a critical review. In *Computational Science and Its Applications–ICCSA 2010*, pages 528–544. Springer, 2010.
- [29] M. Egenhofer. Reasoning about binary topological relations. In *Advances in Spatial Databases*, pages 141–160. Springer, 1991.
- [30] M.J. Egenhofer and R.D. Franzosa. Point-set topological spatial relations. *International Journal of Geographical Information Science*, 5(2):161–174, 1991.
- [31] Jérôme Euzenat and Angelo Montanari. Time granularity. *Foundations of Artificial Intelligence*, 1:59–118, 2005.

- [32] Andrew U. Frank, Irene Campari, and Ubaldo Formentini, editors. *Theories and Methods of Spatio-Temporal Reasoning in Geographic Space*, volume 639 of *LNCS*, Pisa, Italy, 1992. Springer.
- [33] A. Galton. A formal theory of objects and fields. In *COSIT 2001: Proceedings of the International Conference on Spatial Information Theory*, pages 458–473, London, UK, 2001. Springer-Verlag. ISBN 3-540-42613-2.
- [34] A. Galton. Desiderata for a spatio-temporal geo-ontology. *Spatial Information Theory*, 2825:1–12, 2003.
- [35] A. Galton. On what goes on: The ontology of processes and events. In B. Bennett and C. Fellbaum, editors, *Proceedings of the International Conference on Formal Ontology in Information Systems (FOIS 2006)*, volume 150. IOS Press, 2006.
- [36] A. Galton. Experience and history: Processes and their relation to events. *Journal of Logic and Computation*, 18(3):323–340, 2007.
- [37] A. Galton. Spatial and temporal knowledge representation. *Earth Science Informatics*, 2(3):169–187, 2009.
- [38] A. Galton and R. Mizoguchi. The water falls but the waterfall does not fall: New perspectives on objects, processes and events. *Applied Ontology*, 4(2):71–107, 2009.
- [39] Antony Galton. Fields and objects in space, time, and space-time. *Spatial cognition and computation*, 4(1):39–68, 2004.
- [40] Antony Galton and Michael Worboys. Processes and events in dynamic geo-networks. In *GeoSpatial Semantics*, pages 45–59. Springer, 2005.
- [41] A. Gangemi, N. Guarino, C. Masolo, A. Oltramari, and L. Schneider. Sweetening ontologies with DOLCE. *Knowledge engineering and knowledge management: Ontologies and the semantic Web*, pages 223–233, 2002.
- [42] Narain H Gehani, Hosagrahar V Jagadish, and Oded Shmueli. Composite event specification in active databases: Model & implementation. In *VLDB*, volume 92, pages 327–338. Citeseer, 1992.
- [43] Narain H Gehani, Hosagrahar V Jagadish, and Oded Shmueli. Event specification in an active object-oriented database. In *ACM Sigmod Record*, volume 21, pages 81–90. ACM, 1992.

- [44] Michael Gelfond. Answer sets. *Foundations of Artificial Intelligence*, 3:285–316, 2008.
- [45] Michael F Goodchild and James Proctor. Scale in a digital geographic world. *Geographical and environmental modelling*, 1:5–24, 1997.
- [46] Ian N Gregory and Richard G Healey. Historical gis: structuring, mapping and analysing geographies of the past. *Progress in Human Geography*, 31(5):638–653, 2007.
- [47] Pierre Grenon and Barry Smith. SNAP and SPAN: towards dynamic spatial ontology. *Spatial Cognition & Computation*, pages 69–104, 2004.
- [48] Anil Gupta and Kumar Anil Gupta. *The Logic of Common Nouns: an investigation in quantified modal logic*. Yale University Press New Haven, 1980.
- [49] Trevor M Harris, L Jesse Rouse, and Susan Bergeron. The geospatial semantic web, pareto gis, and the humanities. *The Spatial Humanities: GIS and the Future of Humanities Scholarship*, pages 124–142, 2010.
- [50] Jerry R. Hobbs, George Ferguson, James Allen, Richard Fikes, Pat Hayes, Drew McDermott, Ian Niles, Adam Pease, Austin Tate, Mabry Tyson, and Richard Waldinger. A DAML ontology of time. <http://www.cs.rochester.edu/~ferguson/daml/daml-time-nov2002.txt>, 2002. Accessed on 19/06/2013.
- [51] Kathleen Hornsby and Max J. Egenhofer. Identity-based change: A foundation for spatio-temporal knowledge representation. *International Journal of Geographical Information Science*, 14:207–224, 2000.
- [52] GM Jacquez, S. Maruca, and M.J. Fortin. From fields to objects: a review of geographic boundary analysis. *Journal of Geographical Systems*, 2(3):221–241, 2000.
- [53] A. Jakulin and D. Mladenic. Ontology grounding. In *Proceedings of the Conference on Data Mining and Data Warehouses (SiKDD 2005)*, 2005.
- [54] R. Kowalski and M. Sergot. A logic-based calculus of events. *New generation computing*, 4(1):67–95, 1986.
- [55] Werner Kuhn. Geospatial semantics: why, of what, and how? In *Journal on data semantics III*, pages 1–24. Springer, 2005.

- [56] Inga Kulik, Kathleen Stewart Hornsby, and Ian D Bishop. Modeling geospatial trend changes in vegetation monitoring data. *Computers, Environment and Urban Systems*, 35(1):45–56, 2011.
- [57] Nina Siu-Ngan Lam and Dale A Quattrochi. On the issues of scale, resolution, and fractal analysis in the mapping sciences\*. *The Professional Geographer*, 44(1): 88–98, 1992.
- [58] C. Masolo, A. Gangemi, N. Guarino, A. Oltramari, and L. Schneider. Wonderweb EU project deliverable d18: The wonderweb library of foundational ontologies, 2003.
- [59] J. McCarthy and P.J. Hayes. Some philosophical problems from the standpoint of artificial intelligence. *Machine Intelligence*, 4:463–502, 1969.
- [60] Rob Miller and Murray Shanahan. The event calculus in classical logic–alternative axiomatizations. *Electronic Transactions on Artificial Intelligence (<http://www.etaij.org>)*, 4, 1999.
- [61] A.P.D. Mourelatos. Events, processes, and states. *Linguistics and philosophy*, 2(3): 415–434, 1978.
- [62] Philippe Muller. Topological spatio-temporal reasoning and representation. *Computational Intelligence*, 18(3):420–450, 2002.
- [63] Philippe Muller et al. A qualitative theory of motion based on spatio-temporal primitives. *Proceedings of the 6th International Conference on Knowledge Representation and Reasoning (KR)*, 98:131–141, 1998.
- [64] OpenGeo. Introduction to PostGIS. Section 17: Geography. <http://workshops.opengeo.org/postgis-intro/geography.html>, 2013. Accessed on 15/03/2013.
- [65] C. Parent, S. Spaccapietra, and E. Zimányi. The MADS data model: Concepts to understand the structure of your spatial and temporal data. *Journal of Informative Modelling for the Architectural Heritage*, 0(1):51–64, 2006.
- [66] Donna Peuquet, Barry Smith, and Berit Brogaard. The ontology of fields. *Proceedings of summer assembly of the university consortium for geographic information science*, 2(3):5, 1998.

- [67] Donna J. Peuquet. It's about time: A conceptual framework for the representation of temporal dynamics in geographic information systems. *Annals of the Association of American Geographers*, 84(3):441–461, 1994.
- [68] J. Pustejovsky. The syntax of event structure. *Cognition*, 41(1-3):47–81, 1991.
- [69] D.A. Randell, Z. Cui, and A.G. Cohn. A spatial logic based on regions and connection. *Proceedings of the 3rd International Conference on Knowledge Representation and Reasoning (KR)*, 92:165–176, 1992.
- [70] R.G. Raskin and M.J. Pan. Knowledge representation in the semantic web for earth and environmental terminology (sweet). *Computers & Geosciences*, 31(9):1119–1125, 2005.
- [71] Raymond Reiter. The frame problem in the situation calculus: A simple solution (sometimes) and a completeness result for goal regression. *Artificial intelligence and mathematical theory of computation: papers in honor of John McCarthy*, 27: 359–380, 1991.
- [72] Femke Reitsma and Thomas Bittner. Scale in object and process ontologies. In *Spatial Information Theory. Foundations of Geographic Information Science*, pages 13–27. Springer, 2003.
- [73] Erik Sandewall. Towards the validation of high-level action descriptions from their low-level definitions. *AI Communications*, 9(4):214–224, 1996.
- [74] S. Scheider, A. Devaraju, K. Janowicz, P. Maue, S. Schade, C. Keßler, J. Ortmann, M. Bishr, T. Fincke, T. Weigel, et al. Grounding geographic information. In *Proceedings of the 12th AGILE International Conference on Geographic Information Science*, 2009.
- [75] Hedda Rahel Schmidtke and Woontack Woo. A size-based qualitative approach to the representation of spatial granularity. In *International Joint Conferences on Artificial Intelligence (IJCAI)*, volume 2007, pages 563–568, 2007.
- [76] S. Schockaert, M. De Cock, C. Cornelis, and E.E. Kerre. Fuzzy region connection calculus: Representing vague topological information. *International Journal of Approximate Reasoning*, 48(1):314–331, 2008.
- [77] M. Shanahan. The event calculus explained. *Artificial intelligence today*, pages 409–430, 1999.

- [78] Murray Shanahan. *Solving the frame problem: a mathematical investigation of the common sense law of inertia*, volume 436. MIT press Cambridge, 1997.
- [79] Murray Shanahan. The frame problem. In Edward N. Zalta, editor, *The Stanford Encyclopedia of Philosophy*. Stanford University, winter 2009 edition, 2009.
- [80] Yosio Shimabukuro, Valdete Duarte, Liana Anderson, Dalton Valeriano, Egídio Arai, Ramon Freitas, Bernardo Friedrich Rudorff, and Maurício Moreira. Near real time detection of deforestation in the Brazilian Amazon using MODIS imagery. *Ambiente & Água-An Interdisciplinary Journal of Applied Science*, 1(1):37–47, 2007.
- [81] Barry Smith. Agglomerations. In David M. Mark Christian Freksa, editor, *Spatial information theory. Cognitive and computational foundations of geographic information science*, pages 267–282. Springer, 1999.
- [82] J.F. Sowa et al. *Knowledge representation: logical, philosophical, and computational foundations*, volume 511. MIT Press, 2000.
- [83] J Steenbergen, JMDD Baars, MR van Stralen, JA Craeymeersch, et al. Winter survival of mussel beds in the intertidal part of the Dutch Wadden sea. *Monitoring and assessment in the Wadden Sea*, ed. K. Laursen, pages 107–111, 2005.
- [84] J. Stell and M. Worboys. A theory of change for attributed spatial entities. *Geographic Information Science*, pages 308–319, 2008.
- [85] Michael Thielscher. Introduction to the fluent calculus. *Electronic Transactions on Artificial Intelligence* (<http://www.etaij.org>), 3, 1998.
- [86] Laure Vieu. *Sémantique des relations spatiales et inférences spatio-temporelles: une contribution à l'étude des structures formelles de l'espace en Langage Naturel*. PhD thesis, Université Paul Sabatier, 1991.
- [87] Alfred North Whitehead, David Ray Griffin, and Donald W Sherburne. *Process and reality: An essay in cosmology*. Cambridge University Press Cambridge, 1929.
- [88] Jan Wielemaker. SWI-Prolog ODBC Interface. <http://www.swi-prolog.org/pldoc/package/odbc.html>, 2013. Accessed on 20/02/2013.
- [89] Stephan Winter. Ontology: buzzword or paradigm shift in GI science? *International Journal of Geographical Information Science*, 15(7):587–590, 2001.

- [90] Frank Wolter and Michael Zakharyashev. Spatio-temporal representation and reasoning based on RCC-8. In *Proceedings of the 7th International Conference on Principles of Knowledge Representation and Reasoning (KR2000)*, pages 3–14. Morgan Kaufmann, 2000.
- [91] M. Worboys and K. Hornsby. From objects to events: GEM, the geospatial event model. *Geographic Information Science*, pages 327–343, 2004.
- [92] Michael Worboys. Event-oriented approaches to geographic phenomena. *International Journal of Geographical Information Science*, 19(1):1–28, 2005.
- [93] Liu Yaolin, Martien Molenaar, Menno-Jan Kraak, and Liu Yanfang. Spatial object aggregation based on data structure, local triangulation and hierarchical analyzing method. *Geo-spatial Information Science*, 5:44–54, 2002.
- [94] May Yuan. Representing complex geographic phenomena in GIS. *Cartography and Geographic Information Science*, 28(2):83–96, 2001.
- [95] L. A Zadeh. Fuzzy sets. *Information and control*, 8(3):338–353, 1965.
- [96] L. A. Zadeh. Fuzzy algorithms. *Information and Control*, 12:94–event102, 1968.
- [97] C. Zaki, M. Servières, and G. Moreau. Transforming conceptual spatiotemporal model into object model with semantic keeping. In *SECOGIS*, 2011.