Conceptual Modeling Techniques for Spatiotemporal Applications

by

Rosanne Price

B.A. with Departmental and General Honors, M.Sc.

School of Computer Science and Software Engineering

Monash University

Australia

Thesis submitted in fulfilment of the requirement

for the degree of Doctor of Philosophy

November 2001

Table of Contents

3 A Conceptual Modeling Language for Spatiotemporal Applications

4 Mapping STUML to UML

6 Topological Constraints in Spatial Part-Whole Relationships

7 Conclusion

List of Figures

List of Tables

Abstract

Spatiotemporal applications manage data with spatial and temporal properties, exemplified by Multimedia Information Retrieval and Geographic Information systems. The former class of applications involves catalog and retrieval of multimedia artifacts in large collections, including image, graphic, and video sequences with spatiotemporal semantics. In addition to such multimedia artifacts, Geographical Information systems manage spatially referenced data such as soil acidity distribution and property boundaries. Temporally referenced data is typically required for trend analysis to support decision making and planning. Such applications are further characterized by the need to manage complex spatial objects formed from spatial sub-units and to specify constraints on their topology, as illustrated by an administrative region divided into non-overlapping land-use zones.

Although specialized software for multimedia and spatial data management exists, currently available commercial products offer very little support for the temporal dimension. There is nothing comparable to the suite of theoretical and practical tools and techniques available for business system development. High level support for modeling complex spatial objects, including both spatial and non-spatial characteristics, and specification of their topological constraints is also missing. For example, current topological classification schemes do not consider the range of spatial object types required to provide general support for spatial applications nor the set-based constraints required for describing the topology between complex object sub-units.

This thesis addresses the problem of providing support for conceptual analysis and design of spatiotemporal applications, primarily in the geographic applications context. The goal is to develop modeling techniques that are flexible enough to support a wide range of spatiotemporal applications yet simple enough to be used in the early stages of application development. A graphical modeling language is proposed that offers built-in support for capturing spatially referenced, time-varying information, based on extending the well-accepted and well-supported object-oriented standard for graphical modeling languages, the Unified Modeling Language. Spatiotemporal semantics are modeled by defining a minimal set of base constructs that can be combined and applied in a consistent and orthogonal manner to provide expressive power without sacrificing simplicity or understandability.

The specific problem of modeling complex spatial objects and topological constraints on their spatial sub-units is then further investigated. Five different types of complex spatial relationships that are of general utility in spatial applications are identified based on their implied semantic constraints (both spatial and non-spatial) and formally defined using a consistent classification framework. A practical approach to integrating support for complex spatial objects into conceptual modeling languages is demonstrated using the proposed spatiotemporal language.

Finally, the thesis proposes a simple but comprehensive method of describing topological constraints for a complex spatial object during application analysis and design. The proposed classification scheme and modeling constructs cater for the wide range of irregular and composite spatial object types required for general support of geographic applications. Binary and n-ary (set-based) topological relationships are used to model constraints between the complex spatial object and its sub-units and between sub-units respectively.

Declaration

I declare that the thesis contains no material that has been accepted for the award of any degree or diploma in any university and that, to the best of my knowledge, the thesis contains no material previously published or written by any other person except where due reference is made in the text.

Signed

Date

School of Computer Science and Software Engineering

Monash University

Caulfield East, Vic. 3145

Australia

Acknowledgements

I would like to thank Dr. Nectaria Tryfona and Professor Christian Jensen from Aalborg University, who hosted my visits to Denmark and whose active support in person during the visit and by email between visits inspired my work. Specifically, I would like to acknowledge the passionate and dedicated guidance, prompt and thorough examination of my work, and valuable feedback of Dr. Nectaria Tryfona and the meticulous attention to detail and inspired trouble-making of Professor Christian Jensen.

Thanks are due to Professor Bala Srinivasan for his feedback and perspicacious advice, especially regarding thesis structure, and his sincere interest in my welfare beyond purely academic concerns, including everything from permanent residency applications to pool maintenance.

I am thankful to Professor Kotagiri Ramamohanaro for his astute suggestions as to direction, invaluable advice, especially in moments of crisis, and consistent encouragement.

I am grateful to all my close friends in Melbourne for their support, caring, generosity, and for the richness and joy they added to my life, to Liz and Joe for sharing their home and warmth, Shonali for her interested discussions of my work, Maria for her survival tricks, Campbell, Murli, Megan, and Vicki, and to all the other rabble-rousers at Monash, Le, Peter G., Peter S., Krisztian, and Salah. I am grateful to my new family, the Pendharkars, for offering sustenance and entertainment while I was working non-stop on the thesis.

I thank my husband, Santosh, for the countless chores and long-distance errands he assumed and is still assuming on my behalf and for always being there when I called despite everything.

Finally, I thank my parents and dedicate this thesis to them as my foremost support through times of trouble. Their constant love, support, encouragement, and faith in me through difficult times enabled me to continue working through all the upheavals of these last few years.

Chapter 1

Thesis Overview

1.1 Introduction

Recent interest in spatiotemporal data modeling comes from *Multimedia Information Retrieval Systems* (MIRS) and *Geographic Information System* (GIS) applications, which manage data with spatial and/or temporal properties. Other applications potentially involving spatiotemporal modeling include mobile computing and telephony applications. The spatiotemporal data of interest to an application can include both temporal changes in spatial data and their inter-relationships, thematic (i.e. alphanumeric or non-spatial) data whose values are dependent on location and time, and composite data whose components vary over time and space. For example, the location of an object or spatial relations between objects depicted in a video frame can change in subsequent frames. Similarly, property boundaries recorded in a GIS cadastral system (i.e. land management system) or the location of resources and users in a mobile computing system may vary over time. Geographic applications related to weather forecasting may involve thematic properties such as temperature or humidity whose values depend on the location and time of measurement. Analogously, the components required in an automobile assembly are dictated by administrative regulations in effect for a specific jurisdiction and validity period.

Information systems for such applications must be able to capture and manage spatially and temporally referenced data in order to support efficient retrieval of data based on spatial and/or temporal semantics. For example, in order to satisfy a user request for video sequences that show a person walking across a street and colliding with a car, information regarding dynamic changes in an object's location and its spatial relationships with other objects depicted in the videos are required. Spatiotemporal data is also required for trend analysis to aid decision making and planning. For example, effective urban planning decisions regarding transport infrastructure depend on being able to predict future demographic shifts based on historical and current patterns. Models capable of representing the spatial and temporal semantics of a given application are required for data requirements and constraint specification in information systems development and for data manipulation and visualization in information systems operation. After providing an overview of the application areas driving the development of spatiotemporal models in Sections 1.2 through 1.4, this chapter reviews the motivations, objectives, scope, and contributions of the thesis in Sections 1.5 and 1.6 and describes the thesis structure in Section 1.7.

1.2 Multimedia Information Retrieval Systems

The term *information retrieval* traditionally referred to the retrieval of large, unstructured data objects—unstructured, that is, from the perspective of the retrieval system—in an extensive collection of such objects [Baez99, Price91, Salt83, Salt89]. For example, *Document Management Systems* traditionally treated each document in a document collection as a set of words with relative word frequencies (within the document versus in the collection as a whole) that could be mechanically matched against text query keywords without further consideration or knowledge of internal document structure or semantics. More advanced systems attempt to provide some level of interpretation based on word phrases, which can be considered the operative semantic unit for text [AlK99].

In *Image Retrieval Systems*, the non-sequential nature of images means that it is much more difficult to automatically distinguish discrete sub-units that can be meaningfully and easily compared with a user query. Visual semantics are dependent on the spatial properties and inter-relationships of objects, where spatial distributions of low-level features such as color, texture, and intensity are used to identify spatial objects within the image. In some cases, the entire image is required to understand the semantics of any part; i.e. the image itself can be considered the operative semantic unit. Therefore, images were traditionally retrieved by name or through the use of manually entered text annotations.

As the generation and collection of digital information increased, information retrieval systems expanded to include other types of multimedia data, such as image sequences, video, still and animated graphics, and audio. Of these multimedia data types, image sequence, video, and animation are particularly relevant in the context of spatiotemporal applications because of their visual and dynamic nature. The task of retrieving data from such multimedia databases is called *multimedia information retrieval (MIR)* and the systems responsible for this task are called *multimedia information retrieval systems* (*MIRS)* (also called *multimedia information systems*).

The distinguishing characteristics of MIRS, as opposed to database management systems (DBMS), is that (*i*) a single retrieved data object has quite complex and dense semantics whose interpretation and relevance varies considerably between different users and (*ii*) the semantics and internal structure of the data objects have not been extracted or made explicit to the system. Typically, a single collection may be used by a wide variety of users for quite different purposes. Therefore, the user is primarily responsible for interpreting retrieved data objects to determine the relevant semantic content in the context of a specific application need. In contrast, DBMS traditionally manage a collection of small alphanumeric data objects explicitly extracted and organized on the basis of the semantics relevant to a specific set of applications. For example, in a relational DBMS, employee personal details, salary, and job responsibilities are explicitly maintained as separate attribute columns and related to relevant department information through the foreign key column department number. Matching a user request to relevant data is then quite straightforward.

The basic challenge posed by MIRS—given the variable interpretations, size, and semantic complexity of the data—is finding ways to improve query effectiveness and efficiency through automated or semi-automated methods. Essentially, rather than depending on the use of manual text annotations for retrieval; the new generation of MIR research and systems exploits *content-based retrieval*, i.e. retrieval techniques based directly on the multimedia data contents. Content-based retrieval techniques vary considerably with respect to the semantic level of the content representation [AlK99, Gros97, Yosh99].

At the bottom end of the semantic scale, one approach utilizes low-level features such as image color and texture that can be automatically extracted from both indexed and query images. A sample image or graphic can then be used as a query in an attempt to retrieve *similar* images. However, there is a significant gap between low-level feature-based image descriptions and the semantic-based information needs of users.

More complicated automated or semi-automated image processing techniques can be used to distinguish spatial objects and their inter-relationships within an image or graphic. Domain-specific knowledge-based techniques, relevance feedback, and automatic or semi-automatic learning techniques can be used both (*i*) to augment image processing techniques to map low-level features to spatial objects and (*ii*) to map spatial objects and object inter-relationships into higher-level domain-specific semantics as described in [AlK99, Yosh99]. In fact, domain knowledge is essential for automated semantics extraction from images and from multimedia in general. For example, a constellation of spatial objects identified in an image or graphic may actually be regarded as a single composite entity semantically, e.g. details of a car or plane assembly. Domain knowledge is required to interpret the constellation of spatial objects as an integrated car or plane assembly. Analysis and subsequent understanding of the spatial relationships between the constituent parts may further aid in semantic interpretation of multimedia data, similarity matching, and subsequent retrieval. Mapping low-level image features to spatial objects and then to higher-level semantics is currrently an active area of research.

In the context of content-based retrieval of multimedia data with a temporal component (image sequences, animated graphics, and video), dynamic changes in spatial objects and their relationships or in higher-level semantics can be also used for cataloguing or querying multimedia databases. For instance, a video can be catalogued based on changing objects and object relationships within each scene of the video, where a scene itself represents a complex abstraction that must be derived from the video. Changes in low-level features are then used in image processing to extract semantics, e.g. as one way to detect a change in scene, rather than directly as a retrieval mechanism.

MIRS typically have management, retrieval, and storage functions, including editing and presentation tools [Chris95, Gros97] for creating and displaying multimedia presentations composed from different multimedia types (e.g. a combined image sequence, graphics animation, and audio track). Effective MIRS require the development of specialized image and graphics processing, indexing, modeling, editing, presentation, and storage techniques. For example, the composition and presentation of multimedia presentations requires time-based and space-based media models that can be used for scheduling and controlling synchronized playback of the different elements in the presentation. Rather than describing spatial and temporal properties of multimedia data contents (e.g. observed or recorded time-based changes in spatial or non-spatial data properties); such models (described in [Gibbs97]) are used to represent display and scheduling constraints in multimedia presentations.

In the context of this thesis, the relevant issue is spatiotemporal modeling for representation of multimedia data content. In particular, spatiotemporal semantics extracted from time-based visual multimedia data needs to be modeled at a level of abstraction suitable to facilitate semantic-based query and retrieval. Therefore, modeling techniques are required that can be used to effectively describe spatiotemporally referenced objects, object relationships, and properties. Such modeling techniques are also required for another class of spatiotemporal applications, Geographic Information Systems, that manage geographically referenced data, including both multimedia artifacts (e.g. satellite images) and extracted data values with explicitly recorded spatiotemporal semantics (e.g. soil acidity or humidity measurements).

1.3 Geographic Information Systems

GIS can be simply defined as computer-based information systems for geographically referenced data, i.e. spatial data whose position is described with respect to a geographic coordinate system. Alternatively, instead of describing GIS from a database perspective in terms of the types of data being managed, GIS can be defined in terms of functionality, application, components, or domain as illustrated by the various definitions in [Burr98, Haze91, Lang93, Worb95]. For example, a GIS can be defined as a comprehensive management system providing data capture, modeling, manipulation, retrieval, analysis, presentation, and quality control functions for spatial data. From an application perspective, GIS can be regarded primarily as an inventory, mapping, scheduling, simulation, or decision support system. Component-based descriptions refer to an integrated combination of hardware, software, data, people, and organizational procedures [Burr98, Haze91]. A domain-based perspective leads to distinctions between *spatial information systems* (SIS), *land information systems* (LIS), and *geographic information systems* (GIS).

Although usage of these terms vary, SIS applications are usually regarded as a super-set of GIS applications in that they include data with non-geographic spatial references. Examples include application data on molecular configurations in chemical applications or pathological anatomy in medical applications. Although such systems have basically the same functionality as GIS, the term *LIS* generally refers to cadastral systems (i.e. survey or land management systems) tracking land parcel information (e.g. ownership, dimensions) over a wide area and intended to be continuously used over a long period of time. In contrast, the term *GIS* is often used for systems dealing with environmental land management issues focused on a smaller geographic area and narrow topic of interest usually associated with a specific short-term project. In this thesis, the term *GIS* is used as a general term encompassing both cadastral and non-cadastral applications.

As compared to MIRS, GIS include data from a much wider variety of sources, including not only multimedia artifacts such as maps and aerial photographs, but also data entered as a result of field observations or sensors. Image processing and analysis techniques are relevant to both MIRS and GIS; however, additional data analyses techniques are required for GIS. These include:

- network analysis to determine optimal routing,
- terrain analysis to aid siting (i.e. positioning) decisions and understand location-based visibility,
- location analysis to determine neighborhood and proximity relations, and
- layer-based analysis to satisfy complex queries or constraints on multiple types of spatially referenced thematic attributes (e.g. vegetation type, slope, soil acidity).

Multimedia composition and editing is relevant to GIS as well as MIRS. For example, it is important in the context of GIS map production and data visualization.

A further distinction between multimedia databases and spatial or geographic databases is that the former traditionally consisted of uninterpreted multimedia artifacts whereas the latter included explicit semantic information relevant in a given application context. For example, explicit information on topological features and relationships between spatial objects can be extracted from multimedia artifacts and stored in a geographic database. However, advanced MIRS today may also incorporate semantic information describing the contents of the multimedia artifact from which it was extracted.

An understanding of the spatial relationships between components of a composite object is of even more importance in GIS than in traditional MIRS applications, since the analysis functions required and range of data types managed are typically much more extensive in GIS than MIRS. Management of composite spatial objects and their spatial sub-units is an important characteristic of many GIS applications. For example, GIS may be used to manage land use zones (e.g. residential, agricultural, industrial) or electoral districts within an administrative region or components of an electrical utility or transportation network. An understanding of topological, geometric, and orientation relationships between the components in such composite objects is often essential to their effective management.

As with MIRS, including a temporal dimension is important for retrieval of data based on spatiotemporal patterns. The temporal dimension is of further importance in GIS both to answer historical queries and to enable trend analysis, predictive modeling, and simulation of dynamic processes to support planning and decision making. The temporal dimension is particularly important to GIS in the context of managing complex spatial objects, since the identity of the components and their spatial relationships may vary over time. While spatiotemporal data modeling concerns in MIRS are mostly related to time-based changes in spatial data and their inter-relationships; GIS must also consider thematic attributes that vary over time or space.

The collection of multimedia artifacts in a GIS can be viewed as a multimedia database [AlK99, Yosh99]. However, the range of data sources, spatial data types, analyses, and associated semantic information typically handled by GIS is far more comprehensive than that traditionally considered in MIRS. To illustrate typical applications, GIS systems are important for urban and regional planners, civil engineers responsible for public utilities, epidemiologists studying disease distribution and spatiotemporally correlated factors, police and criminologists, military organizations, and tourism authorities and agents. Two new classes of applications with geographically referenced data, mobile computing and telephony, have become active areas of research on their own merit and are discussed next.

1.4 Other Spatiotemporal Applications

There has been dramatic increase in the use of mobile computing and mobile phones in recent years. In mobile telephony, users can use their mobile phone to access telephone services from any location within the broadcast range of a phone service cell, where an array of phone service cells provides coverage over a wide area. Similarly, using a combination of wireless and traditional wire networks, mobile computing allows users to have continuous access to networked services as they travel, using a lap-top or other mobile computing device. The fundamental characteristic differentiating mobile computing from traditional distributed computing systems is the frequent movement of computers between locations and concomitant changes in network connections to access resources in the new location. In addition, limitations of the mobile devices with respect to power and screen size require that mobile computing techniques be adapted to account for these restrictions. Frequent reconnection and mobile device limitations have implications for authorization, billing, addressing and communication, security, transaction management, concurrency control, backup, recovery, resource search and selection, and execution functions [Barb99, Form94, Imiel93].

Modeling mobile telephony and computing is fundamentally an exercise in spatiotemporal data modeling, since the locations of both the resources to be accessed (e.g. phone service cells or computing resources) and mobile users can change over space and time. Although the majority of current commercial and research work focuses primarily on current rather than historical data [Barb99]; companies concerned with trend analysis, performance, evaluation/tuning, and security issues are beginning to consider these applications in a wider context. In order to improve planing and decision-making, historical data regarding usage and movement patterns, security violations, and response time can be regarded as a valuable resource. However, exploitation of this resource requires effective conceptual modeling techniques to facilitate effective conceptualization and manipulation of spatiotemporal data.

1.5 Motivations and Objectives of the Thesis

Historically, early origins of current GIS can be traced to the science of cartography and the requirements of map making. In general, commercially available GIS still reflect the limitations of their paper map predecessors in that they are designed to handle two-dimensional and static views of discrete spatial entities. As noted in [Burr98], although GIS capable of handling three-dimensional geographic data are now available; the challenges posed by complex (i.e. composite spatial objects formed from multiple spatial sub-units), continuous, and temporally dependent spatial data remain problematic. For example, support for modeling both spatial and non-spatial properties of complex spatial objects and their topological constraints[1] is lacking in current GIS systems and proposed models. It is further noted that "It becomes obvious from current developments and reported research that there is still

[1] Since topology is an easily distinguished and relatively stable characteristic, it is of particular significance in modeling complex spatial objects.

considerable need for theoretical and technical developments in the modelling of geographical phenomena" [Burr98, p. 294].

As with GIS, commercial MIRS for image, graphic, and video data are based primarily on static features and semantics and do not support queries based on spatiotemporal properties. There has been relatively little research in the area of content-based retrieval of multimedia data based on spatiotemporal semantics. For example, Dimitrova [Dimit97] noted that existing models of video semantics depended either on manual text annotations, static semantics derived through image analysis on individual image frames, or iconic representations derived from automatic methods of detecting scene changes. In the last case, limited temporal information is used in the analysis process to derive representative icons for video scenes; however, there is no further provision for modeling or query based on temporal semantics. Another model, described in [Oomo97], associates a time interval and a manually entered text description with a video segment. Any video segment can then inherit the description from another video segment if the time interval associated with the first segment is contained in that of the second. Although this model makes some provision for the temporal dimension of video semantics, it does not model spatial or spatiotemporal semantics.

MIR research that does directly model spatiotemporal semantics [Arit97, Day95, DelB96, Dimit97, Dion98], described in the next chapter of the thesis, considers only temporal changes in spatial objects. Consideration of spatiotemporally dependent thematic attributes is the provenance of GIS rather than traditional MIR applications. Developing spatiotemporal models for multimedia data semantics, especially at a high level, is described as an area requiring further research in [AlK99, Yosh99]. Spatiotemporal data modeling of other applications with spatial and temporal dimensions, such as mobile computing and telephony, has yet to be addressed.

Although an extensive literature for spatial databases and for temporal databases is already available, efforts to integrate the two in a consistent framework, understand the semantics of spatiotemporal data, and develop spatiotemporal data models are of far more recent origin. Initial research in this area, primarily from the GIS community, focused on efficient representation and access to spatiotemporal data [AITa93, AlTa94, Lang89, Lang93]. More recent work focuses on higher-level data models [Arit97, Beck96, Brod00, Clar95, Day95, DelB96, Dimit97, Dion98, Erwig99, Faria98, Golsh94, Guti98, Paren99, Tryf99, Worb92], described in detail in the next chapter of the thesis. However, there is nothing comparable to the theoretical and practical support available for the development of business systems, i.e. conceptual data models such as the entity relationship or object-oriented data models for analysis and design; logical data models such as the relational model; implemented software tools such as case tools providing automatic translation between the conceptual and logical phases; and relational DBMS automating data management tasks.

This thesis addresses the problem, discussed above, of providing support for the requirements analysis and conceptual design phases of spatiotemporal application development. The research perspective adopted is that of GIS applications, since they generally involve a wider range of spatiotemporal data types and sources than MIR or mobile applications and have a longer history in terms of considering spatiotemporal representations. The goal is to develop modeling techniques that are flexible enough to support a wide range of spatiotemporal applications yet simple and intuitive enough to be used by both developers and clients in the early stages of application development. The focus is practical and utilitarian, i.e. developing techniques and modeling constructs that can serve as useful aids in spatiotemporal application analysis and design, rather than strictly theoretical exercises in conceptual modeling. However, the proposed techniques and constructs should be supported by formal definitions to allow for their unambiguous specification and provide a basis for automatically converting conceptual level schemas to implementable forms in later stages of development.

1.6 Scope and Contributions of the Thesis

This thesis first considers the general problem of providing a conceptual data model suitable for spatiotemporal analysis and design in the same way that the entity relationship and object oriented data models serve as modeling and communication tools in business applications. The specific problems of providing support for modeling complex spatial objects and their topological relationships is then considered in depth. Based on this overview, the main contributions of the thesis are as follows.

- A graphical modeling language is proposed that offers built-in support for capturing spatially referenced, time-varying information. The language is based on extending the well-known object-oriented standard, the Unified Modeling Language (UML), to capture the semantics of spatiotemporal data. The resulting extension, *SpatioTemporal Unified Modeling Language (STUML)*, maintains language clarity and simplicity by introducing only a small base set of fundamental modeling constructs: *spatial*, *temporal*, and *thematic*. These three constructs can then be combined and applied as required at attribute, association, or class levels of the object-oriented model based on a few simple rules. A formal functional specification of the fundamental semantic modeling constructs and their symbolic combinations is given, ensuring that the proposed language has a sound theoretical basis. Explicit provision is made for modeling common spatiotemporal properties in a group of related attributes, application-dependent existence dependencies, and different levels of abstraction based on the degree of specification detail required.

- A set of transformation rules is given for mapping STUML to UML schemas. This provides a theoretical basis for implementing STUML schemas using tools and products developed for UML. The conversion rules could also be used as a basis for implementing spatiotemporal extensions to existing UML case tools such as Rational Rose, which are used to automatically convert a conceptual into a logical (implementable) and then a physical (implemented) schema.
- A classification framework and modeling constructs are proposed to facilitate modeling of complex spatial objects, based on the concept of *spatial Part-Whole (PW) relationships* describing asymmetric relationships between spatial objects. The classification framework is based on spatial derivation and constraint relationships between the whole and its parts. This framework can then be used as a basis for defining specific constructs as needed for different applications. To illustrate, five different types of spatial PW relationships that are of general utility in spatial applications are identified based on their implied semantic constraints (both spatial and non-spatial) and formally defined using the classification framework. These are *spatial part*, *spatial membership*, *spatial inclusion*, *spatial cover*, and *spatial equal*. STUML is used to demonstrate the

feasibility of integrating the proposed constructs into an existing conceptual modeling language to provide support for modeling complex spatial objects.

- A comprehensive method of describing topological constraints on the spatial sub-units of a complex spatial object is proposed, which is suitable for high-level application modeling and considers the range of spatial data types required for geographic applications. A classification scheme for describing binary topological relationships is proposed based on the simple and intuitive concepts of intersection and difference. This can be used to describe topological relationships between the whole and its parts in spatial PW relationships. Set-based topological relations are then proposed to model topological relationships between *n* spatial objects, as for instance between the parts in a spatial PW relationship. As an illustration of the utility of this approach, it is used to extend the classification framework and modeling constructs for spatial PW relationships based on topological constraints.

1.7 Structure of the Thesis

The thesis is organized into seven chapters. Chapter 2 provides the background necessary for the rest of the thesis. First, the research issues relevant in the context of spatiotemporal data modeling are discussed in detail and a comparison of GIS and MIR perspectives and models are given based on these issues. Recent work from both research communities are reviewed and evaluated. Representative examples are selected to illustrate the range of different approaches found in the literature. Three graphical models developed contemporaneously with the work in this thesis for conceptual analysis and design are previewed. They are examined in greater depth later in the thesis (in Chapter 3) as a comparison to the graphical spatiotemporal modeling language proposed in this thesis. Chapter 2 concludes by summarizing recent trends and describing open research problems in spatiotemporal data modeling.

The graphical SpatioTemporal Unified Modeling Language (STUML) is described in Chapter 3. After considering the requirements of such a language, the problems inherent in using general graphical modeling languages to model spatiotemporal data semantics are demonstrated using UML. Alternative approaches to supporting spatiotemporal data modeling are evaluated and the rationale for adopting the approach used in this thesis, viz., extending UML, is explained. With this background, the proposed spatiotemporal extension to UML—STUML—is described.

An overview of STUML is followed by an informal description of the use and semantics of the base language elements, the *spatial, temporal, and thematic* constructs. Formal definitions for the constructs and their legal combinations at each model level (e.g. attribute, object, association) are given and illustrated by example. Additional modeling constructs are described that support modeling of common spatiotemporal properties in a group of related attributes, application-dependent existence dependencies between model elements, and application-dependent details of the space and time models assumed.

The advantages of using STUML for spatiotemporal applications are illustrated by comparing the UML and STUML schemas for the same application. Finally, we present a detailed comparison of STUML with the three other graphical modeling languages for spatiotemporal applications that were introduced in Chapter 2.

Chapter 4 provides a comprehensive set of transformation rules for mapping a STUML schema to a UML schema. The suggested conversion mechanisms are intended to provide a general solution that can be used as a basis for implementing STUML schemas using case tools developed for UML or for extending these case tools.

The use of spatial PW relationships to model complex spatial objects is discussed in Chapter 5. Since this work integrates efforts from the object-oriented and spatial research communities, relevant research from both communities is reviewed. After introducing basic terminology and definitions, a classification framework for spatial PW relationships is proposed. Five specific spatial PW relationship types of general utility in modeling spatiotemporal applications are identified, illustrated by example, and defined. Incorporation of the proposed modeling constructs for spatial PW relationships into a conceptual modeling language is then demonstrated with STUML.

In Chapter 6, one important characteristic of spatial PW relationships, topological constraints between spatial parts, is investigated in further depth. First, relevant research in the literature is reviewed and its limitations in the current context discussed. Application design techniques and modeling constructs suitable for modeling binary and set-based topological relationships are proposed and used to further illustrate and extend the work on spatial PW relationships.

The thesis is summarized in Chapter 7 and final conclusions presented. The contributions of the thesis are given and future research directions in the area of spatiotemporal modeling discussed. Finally, we note that much of the information contained in the literature survey presented in Chapter 2 and the research work described in Chapters 3, 5, and 6 have also been reported in the literature [Price99a, Price99b, Price99c, Price00a, Price00b, Price01a, Price01b].

Chapter 2

Spatiotemporal Data Models

2.1 Introduction

Early surveys of spatiotemporal research were biased towards representational and implementation concerns of geographic and cartographic applications [AlTa93, AlTa94, Lang89, Lang93], with the emphasis on change in thematic attribute values across time and space. More recent surveys consider a range of spatiotemporal data models from the *Geographic Information Systems* (GIS) community [Abra99, Pavl98]. These discuss the current implementation approach generally used by those few commercial GIS today that have any temporal support: successive snapshots of the database state at pre-determined time intervals. Spatiotemporal changes must then be derived by calculating differences between snapshots. Since this method is entirely impractical for answering spatiotemporal queries, the GIS research community is motivated to find data models suited to spatiotemporal data.

With the advent of image and video databases, the *Multimedia Information Retrieval* (MIR) community began to show interest in this area, as evidenced by reviews of MIR systems [AlK99, Asla99, Gros97, Yosh99]. Traditionally attention has been focused on image/video processing techniques and retrieval using text annotations (see [Price91]) or low-level image features such as color and texture (see [Asla99]). Recently, however, there has been increasing recognition in the MIR community of the need for higher-level conceptual data models to support content-based retrieval at a higher semantic level. Some of the proposed data models also consider the incorporation of knowledge bases to facilitate automated or semi-automated data classification from images and videos [AlK99, Day95, DelB95, Dion98, Yosh99]. A few of these higher-level and knowledge-based models incorporate spatiotemporal semantics to support queries based on spatial and/or temporal properties of objects represented in animations, image sequences or video.

Although both GIS and MIR are essentially examples of *multimedia information systems*, the two communities have remained fairly distinct and there are no major systematic and comprehensive reviews of spatiotemporal data research intended specifically to compare work from both research communities. However, these communities have overlapping concerns in the representation and modeling of spatiotemporal data.

In this chapter, we review recent efforts from both communities to develop data models for spatiotemporal data. The focus is on modeling spatiotemporal data. Questions of indexing and efficiency are generally outside the scope of this thesis, but are discussed extensively in [Lang93, Zani97]. Representation efficiency is considered only in qualitative terms to highlight differences between research approaches. Analysis and presentation of spatiotemporal data (i.e. user-interfaces) are other areas of enough importance to deserve separate treatment [MacD91, Mitas95, Sloc93], but are not considered here. Spatial, temporal, data modeling, and data representation issues are discussed insofar as they are relevant to provide a background for understanding current research in spatiotemporal data models. Therefore, the emphasis is on conceptual and logical models for spatiotemporal data (i.e. those which consider both spatial and temporal properties), particularly the structural and data manipulation components, as they have received the most attention in the literature to date.

The objectives of this chapter are to:

- highlight the fundamental research issues relevant to spatiotemporal data modeling, particularly at the logical and conceptual levels,
- review and compare current research efforts in this area from both GIS and MIR communities, evaluating the work with respect to the fundamental research issues, and
- assess the current state of spatiotemporal data modeling research in terms of recent trends and future priorities.

The organization of the chapter is as follows. Section 2.2 provides the background for understanding research in spatiotemporal data modeling by examining the relevant issues. Section 2.3 compares the perspective of the two research communities active in this area, viz., GIS and MIR. Recent work is reviewed in Section 2.4 using representative spatiotemporal data models, based on issues and perspectives discussed in Sections 2.2 and 2.3. Section 2.5 summarizes the recent trends demonstrated in the area of spatiotemporal data modeling research and the open problems that provide motivation for the work reported in this thesis and directions for future research. The chapter is summarized in Section 2.6.

2.2 Issues in Spatiotemporal Data Modeling

Although having in common a need to manage *spatial data* and their *changes over time*, spatiotemporal applications may vary considerably with respect to (*i*) the types of spatiotemporal data and operators required and (*ii*) the models for space, time, integrated space-time, data organization, and change processes assumed.

The term *spatiotemporal data* is used to refer both to *temporal changes in an object's spatial properties* and *variation of thematic (i.e. alphanumeric) attributes across time and space*. These represent different types of spatiotemporal data. An example of temporal change in an object's spatial properties is the revision of voting precinct boundaries or object position changes in a video. An example of temporal or spatial variation in thematic attributes is variation in soil acidity or air temperature based on the measurement time and location. In the context of this thesis, these two types of spatiotemporal data are referred to as *temporally dependent spatial objects* and *spatiotemporally dependent thematic attributes* respectively. The term *spatially dependent thematic attribute* is used to refer to changes in a thematic attribute's value over space and *temporally dependent thematic attribute* to changes in a thematic attribute's value over time.

Note that the term *spatially dependent thematic attribute* is adopted to describe spatial variation in thematic attribute values rather than employing terms often used for this concept in the GIS community, such as *spatial attribute* or *thematic layer*. The reason for this is (*i*) to ensure consistent naming conventions for both spatial and temporal variation in thematic attributes and (*ii*) to avoid any possible confusion as to whether an attribute has spatially dependent thematic values (i.e. a thematic domain) or a spatial value (i.e. a spatial domain). This is discussed further later in this section in the context of the specific data model adopted.

The distinction between spatial objects and spatially dependent thematic attributes generally corresponds to an object-based versus a field-based view of space, i.e. discrete, identifiable objects having spatial properties versus thematic attributes whose values vary continuously across a spatial field. Spatial objects are most often represented using vectors to approximate boundaries. Spatially dependent thematic attributes are most often represented with grids of spatial locations (e.g. points or regions) associated with thematic attribute values. In the context of spatiotemporal applications, the object-based versus field-based view of space generally corresponds to the distinction between *temporally dependent spatial objects* versus *spatiotemporally dependent thematic attributes*. A spatiotemporal application may be concerned with either or both data types; this, in turn, is likely to influence the underlying model of space and representations employed.

Another type of spatiotemporal data is *composite data whose components depend on time and/or location*. This means that particular components are associated with a composite only in conjunction with a specific time or location. That is, the identity of the components associated with a composite depends on time and location. For instance, a car model may be designed with different types of components for different localities, e.g. to accommodate the strict anti-polluting regulations in a given state such as California. Thus the car model is location dependent. A car accident, consisting of all the vehicles or other objects involved in the collision, is associated with a specific time and location. Although the components of a particular accident do not vary, they are only associated with the accident at the time and location associated with the accident. An example whose components vary temporally and spatially is the minimum combination of equipment and wards required in a certain category of hospital (e.g. general, maternity, psychiatric), where the relevant regulations determining the applicable base standards vary by locality and time period.

Note that the actual modeling construct used to represent spatial objects or composite data depends on the specific data model employed. In a data model such as the *object-oriented* (OO) model that allows composite attribute domains, an object's spatial properties could theoretically be modeled at the attribute level instead of the object level of the model. In this case, the term *spatial attribute* is used instead of *spatial object*. Or, if the spatial properties change over time, the term *temporally dependent spatial attribute* is used instead of *temporally dependent spatial object*. Composite data are represented as relationships in the *Entity-Relationship* (ER) data model, whereas they are represented as associations and/or class-valued attributes in the OO data model. For example, in this thesis, the terms *spatially*, *temporally*, or *spatiotemporally dependent association* are used to refer to composite data with spatial, temporal, or spatiotemporal dependencies respectively. Therefore, when comparing research proposals for spatiotemporal data models, consideration must be given to the basic *data model* (e.g. OO, ER, relational) used to organize and manage the data.

Consideration of higher-level data models such as the OO model further elucidates the earlier discussion on the naming conventions adopted in this thesis. In particular, a naming convention is required that distinguishes between (*i*) an attribute having a thematic domain, whose specific thematic values are dependent on the spatial location, called a *spatially dependent thematic attribute* in this thesis, and (*ii*) an attribute having a spatial domain, called a *spatial attribute* in this thesis.

A further distinction between spatiotemporal applications can be made based on the view of time and change processes. Depending on the application domain, time may be viewed as continuous or discrete. Continuous data may be understood in terms of either a functional mapping from time to a data value domain or an interpolation function between recorded data values. If discrete, data may be sampled and recorded at regular time intervals or only recorded irregularly when change events occur. Change processes are very much related to the integration of space and time, e.g. temporal changes in spatial data. In some cases, the spatiotemporal data to be recorded represents discrete, instantaneous changes or *events*. An example would be a revision to voting precinct boundaries or composite data whose components depend on time or location. However, other phenomena such as moving objects or spatiotemporally dependent thematic attributes such as soil acidity are better represented as a process of continuous change.

Another consideration for spatiotemporal applications is the types of operators and data manipulation required. This will be influenced by the choice of spatial data model, since spatial objects require different operators from those used for spatial fields. The integrity dimension of spatiotemporal applications is also important; however, there has been very little discussion of it in the literature to date.

The preceding discussion highlights the important issues relevant to the development of a spatiotemporal data model. Subsequent sections consider specific issues in greater depth, viz., data models in Section 2.2.1, space in Section 2.2.2, time in section 2.2.3, integration of space and time in Section 2.2.4, change processes in Section 2.2.5, and spatiotemporal data manipulation in Section 2.2.6.

2.2.1 Data Models

The term *data model* refers to the organizational basis or metaphor used for managing the data, e.g. relations for a *relational* data model; entities and their relationships for the *ER* and *extended ER* (EER) models; object classes encapsulating structural and behavioral properties for the *OO* model, or Abstract Data Types (ADT) (a precursor to object classes but typically without inheritance, polymorphism, or identity). Essentially, the *data model* provides a conceptual or logical framework for organizing, manipulating, and maintaining related data in a database. Besides the data models mentioned above, graph or network-based and logic-based frameworks have been used as the conceptual basis for organizing spatiotemporal data.

Data models and organizational frameworks may be at different levels of abstraction depending on whether they are closer to the user or computer perspective. Although it is sometimes difficult to draw a fixed line between different levels of data modeling, generally the assumption here is that the conceptual data model is concerned with semantic clarity and expressiveness and is independent of the particular implementation to be used. In contrast, the primary concern at the physical level is finding data representations and structures which can be efficiently implemented.

Higher-level *object-based models* such as the ER, EER, and OO models and ADT are more suitable for analysis and conceptual design. In some cases, database management systems (DBMS) based on the high-level model exist and therefore the conceptual design can be implemented without changing metaphors. An example is the OO model where OODBMS are available. In other cases (e.g. the ER model), the development process requires a translation into another lower-level model such as the relational or network model. Spatiotemporal database research that focuses on conceptual analysis and design generally uses the OO or ER models. [Guti98, Paren99, Tryf99]. Relational and network models have been used primarily at the logical levels [Clar95, Day95]. As noted by Worboys [Worb95], decomposing spatiotemporal information into tuples may have disadvantages in terms of analysis (i.e. pattern detection) and manipulation. Therefore, the use of a higher level model may have advantages in terms of usability.

Lower-level models operate at the physical level and are concerned primarily with efficient representations of spatiotemporal data. Research in this area has concentrated primarily on support for spatially dependent thematic attributes [Haze91, Lang88, Lang93, Pequ95]. Each different thematic attribute is represented in a separate structure. The structure consists of a complete base map to represent the initial state (i.e. the spatial distribution of thematic attribute values for a given attribute) and temporally ordered amendments.

Pequet [Pequ95] classifies these models based on their *primary level of organization* and *space representation* as follows.

- *Feature-based organization, Vector representation:*

  Approaches based on vector representation or its extension use geographical features as the primary level of organization [Haze91, Lang93]. The classic example of this is the proposal in [Lang93] using temporally-ordered amendment vectors to show boundary changes to spatial regions, where each spatial region corresponds to a single thematic attribute value.

- *Location-based organization, Grid representation:*

  Langran [Lang88] proposed that each grid location be associated with a linked list of temporally ordered and timestamped thematic attribute values for that location. This approach relies on spatial location as the primary level of organization.

- *Time-based organization, Grid representation:*

  The event-based spatiotemporal data model (ESTDM) is proposed in [Pequ95]. This model consists of a temporally ordered and timestamped linked-list of events. During each event, some locations acquire a new attribute value. For each distinct new attribute value for an event, the set of locations changing to that value is associated with the event. Therefore, we have a hierarchy of time→new attribute value→location (i.e. the set of locations assigned that new attribute value at that time).

Retrieval algorithms are presented in [Pequ95] to evaluate the relative efficiency of answering different types of spatiotemporal queries for each model. Clearly, these three alternative approaches will favor feature-based, location-based, and time-based queries respectively. The use of a base state with amendments in these models has advantages in terms of limiting redundant storage of high-volume spatial information. However, there are several problems with this approach, the most significant being the cumbersome need to reconstruct the state for a requested time period by incrementally assembling it from the base state and previous amendments.

Traditionally, a data model consists of three parts: structure, manipulation, and integrity. Extending a data model with spatiotemporal semantics has implications for all three parts of the data model. Some spatiotemporal research, especially that which explicitly intended to facilitate conceptual analysis and design of spatiotemporal applications, has focused mainly on the structural aspect [Brod00, Paren99, Tryf99]. Other conceptual-level and logical-level spatiotemporal models proposed have also considered data manipulation [Arit97, Beck96, Clar95, Day95, DelB96, Dimit97, Dion98, Faria98, Guti98, Worb92]. Because this requires understanding of the models and representations used, a detailed discussion of spatiotemporal data manipulation is deferred to Section 2.2.6

The integrity component has received the least attention to date in the spatiotemporal community. There has been some discussion of general integrity constraints associated with specific spatiotemporal models, especially temporal constraints between an object and its attributes or between related objects. For example, Parent [Paren99] and Tryfona [Tryf99] consider the representation of topological integrity constraints between related objects. Most integrity research to date has been restricted to the separate domains of time [Bohl94, Schw98] and space [Cock97, Hadz92]. Considerable work is still needed to understand the important categories of spatiotemporal constraints and develop techniques to provide automated integrity support for spatiotemporal data. In the rest of the chapter, we concentrate on specific conceptual or logical spatiotemporal models proposed that include both structural and manipulation components. These models are discussed in detail and compared in Section 2.4.

2.2.2 Space: Conceptual Models and Representation

Models of space discussed in the literature for spatial applications include a set-based model [Worb95] for hierarchical and/or containment spatial relationships, a topological model [Arms79, Gibl77, Suth75] for spatial properties which are invariant over a set of transformations, network spaces [Harar69] based on nodes and edges, and metric spaces [Suth75] having a standard distance function (i.e. obeying a specified set of conditions such as the condition of symmetry). However, a more specific and precise model, upon which all of the above mentioned models can be superimposed, is the Euclidean model. The Euclidean model of space, with a coordinate system allowing standardized metric measurements, is the most natural model of space in the modern world and is the basis of most spatial and spatiotemporal research. This will be the model assumed in the rest of the thesis.

Object and Field Based Models of Space

There are two alternative views of space even assuming a coordinatized, Euclidean model [Worb95]. One approach derives from an object-based view of space, i.e. that discrete, identifiable objects exist which have spatial (or spatiotemporal) properties and relations with other objects. However, some spatial (or spatiotemporal) phenomena are not fundamentally comprised of interacting objects with defined shapes and separate identities. For example, the science of oceanography or climatology involves the study of elements which have constantly shifting shapes without clearly demarcated boundaries and which exhibit continuous spatial variation in their properties, e.g. waves, clouds, air streams as discussed in [Jähne93]. A field-based view of space, with functions relating location to a given thematic property value, is more congruous with such phenomena. Field-based approaches to spatiotemporal modeling are discussed in [Lang93, Pequ95].

It should be noted that the two views do not necessarily imply use of a particular data model and are not necessarily mutually exclusive. An object-based model of space does not *necessarily* imply the use of an OO data model nor does a field-based model of space *necessarily* preclude the use of an OO model. It is important to distinguish between conceptual models of physical space as objects or fields and conceptual models of data organization using relations, objects, etc. as the organizing metaphor. A spatial object could be modeled as a relation in the relational data model and a spatial field as an object class in the OO model. In other cases, a single application may involve both spatial objects and spatial fields that are modeled using a single data model (e.g. both as relations or both as object classes). For instance, Worboys [Worb95] discusses example applications such as regional health systems that are most naturally modeled by a combination of objects for hospitals or health clinics and fields for disease epidemiology. If an OO data model is used to model the application, object classes can be used to model both spatial objects (e.g. having an attribute with a spatial domain) and fields (e.g. having a composite attribute with both thematic and spatial components). However, the choice of which model of space is adopted can have implications in terms of data representation and manipulation in the implementation phase of application development (see the discussion on *Representation* below).

Dimensionality

Another issue to consider when modeling space is the question of dimensionality, i.e. the number of spatial axes or coordinates. Both the dimensions of the underlying search space and the objects or fields being modeled within that space can be considered, e.g. zero-dimensional (0D) points, one-dimensional (1D) lines, two-dimensional (2D) polygons, or three-dimensional (3D) volumes in three-dimensional (3D) space. Most research integrating spatial and temporal data focuses on a 2D embedding space. This partly reflects the 2D nature of the artifacts that have been traditionally used to represent and record space, i.e. paper maps, images, and video. Technical and implementation difficulties associated with storing information about 3D space and the challenge of presenting and visualizing 3D space on 2D computer output devices present additional challenges and explain the common simplification of considering only 2D space. However, 2D models of space have limitations in their ability to accurately represent spatial objects and attributes in a 3D world. For instance, Hazelton [Haze91] points out the problems associated with recording multiple values for spatiotemporally dependent thematic attributes at the same time and 2D location but at different elevations. An example would be recording air temperatures at multiple elevations.

Representation

The dichotomy in field-based versus object-based models of space discussed earlier generally corresponds to a raster versus vector representation. More generally, field-based models employ some type of grid or spatial framework to partition a region. If the partitioning is regular and the sub-divisions correspond to pixels, the result is termed a raster data structure. Although a grid representation is usual for the field-based view of space, it is not always true that a spatial object will be represented by vectors, especially in image sequence or video applications. Assuming an object-based view of space, the basic unit could be interval projections of a 2D minimum bounding rectangle (MBR) or 3D minimum bounding cuboid (MBC) on the coordinate axes, a vector representation of a polygon, or even a raster-based representation of a region. Raster-based representations are sometimes parameterized with thematic data, so that the values of attributes across space can be maintained. Constraint databases have also been used as a higher-level representation of space. This is discussed in section 2.2.4.

2.2.3 Time: Conceptual Models and Representation

Models of time are generally classified as linear, branching, or cyclical. The simplest assumption is a linear and unidirectional model of time. In this model, time advances to the future with complete ordering and no branching. Many authors have noted the need for research into more complex time models, especially in spatiotemporal and GIS applications [Clar95, Lang89, Leban86, Zani97].

An alternative model of time allows branches, i.e. alternative time lines, in the past or future. A time line with past branching is required for applications which require hypotheses of possible causality relationships based on current data, i.e. to theorize as to sequences of previous events which could explain currently observed patterns. The need for retroactive data correction while maintaining the original, erroneous data for reference is another motivation for considering a time model which allows branching in the past. Future or hypothetical branches are required for exploring or recording possible alternative courses of action.

Another model of time formally incorporates the idea of periodicity. Many temporal phenomena are cyclic in nature, e.g. the seasons, animal mating. In fact, our everyday model of time is cyclical in nature and consists of nested cycles: e.g. 4-year cycles of years with a leap year each cycle and, within each year, months, weeks, days, hours, minutes, and seconds. For applications primarily concerned with cyclic phenomena or calendar time (e.g. appointment scheduling applications), a cyclic time model would clearly be advantageous. It would also be necessary to account for irregularities in these cycles; e.g. daylight savings time in calendar applications, El Ninjo effects on seasonal cycles in weather applications.

Finally, some applications may require a combination of these time models (i.e. involve alternative time lines and cycles) or even multiple, coexistent time models. For instance, applications recording and comparing events occurring in parts of the world located in different time zones and with different daylight savings practices may best be modeled by several different, asynchronous time lines. The multiple time lines represent concurrent rather than alternative time models.

Time Density

Another consideration is the density with which the time line is to be modeled. Intuitively, we regard time as continuous (i.e. isomorphic to the real numbers); however, this may not lead to the most suitable representation for implementation purposes and may be difficult to map into any physical system. Other possibilities are dense (i.e. isomorphic to the rational numbers) or discrete (i.e. isomorphic to the natural numbers) representations.

Dimensionality

Although less intuitively obvious than with space, a conceptual model of time may also need to be able to support multiple time dimensions, as highlighted by several authors [Barr91, Lang89, Lang93, Xiao89]. The 1D model of time corresponds to the time that some event occurs or fact is true. Although the terms *world* and *logical* time have also been used for this time dimension, the standard term now adopted by the temporal database community [Jens98] is *valid* time. Any reference to time is generally assumed to be valid time unless otherwise noted.

The most important additional time dimension is *transaction* time (previously referred to as *database, system,* or *physical* time in some papers), which corresponds to the time a fact or object is current in the database. Transaction time can be important for establishing accountability, auditing data modifications, and/or facilitating recovery processes in the event of database failure. Transaction time is well-researched in a purely temporal context with thematic data and has been addressed by several spatiotemporal researchers.

In the context of spatiotemporal, and in particular, GIS applications, other types of time may also need to be considered. *Existence* time of an object may be important for constraining the valid time of that object's attributes and relationships with other objects or for recording knowledge of pre-determined object lifespans. *Survey* time, *display* or *read* time, and *estimated* or *predicted* time are mentioned in literature evaluating temporal requirements of GIS systems [Barr91], although rarely considered in proposed systems to date. Survey-time is related to when measurements were made, which may not correspond to when the measured data was actually first valid. Display or read time shows the time that data was actually seen by users: this can be important in establishing accountability, especially in the context of dynamic systems. For example, checks could be made that users regularly scanned change and error notifications and, if not, they could be held legally accountable for any mistakes resulting from this obsolete understanding. Estimated or predicted times can be used for interpolations of missing data based on theoretical application models, especially in applications relying on imprecise or variable quality image data (e.g. satellite photographs).

For some MIR applications, time is important only in the context of individual multimedia artifacts; e.g. time is expressed relative to an individual video as a frame number. This does not match any of the dimensions described above and represents a time dimension we term *artificial* time.

Despite the possible utility of alternative models, spatiotemporal research work has, almost exclusively, relied on a discrete and linear model of time. This is due to the relative simplicity of this conceptual model and its ease of representation and implementation. Any model incorporating temporality considers either valid or artificial time. Object existence time has primarily been considered in research focusing on conceptual analysis and design [Paren99, Tryf99]; however, the concept has not been clearly defined in the literature. The only other alternative time dimension judged generally applicable enough to be incorporated in spatiotemporal models to date is transaction time.

Representation

The most common time representation is that of timestamping. Data can be associated with a unit of time, i.e. a timestamp, to indicate fact validity, event occurrence, recording time, etc. The standard representational units for time [Jens98] are *instants* (previously called *time points* in some sources), *intervals* or *periods* (i.e. the time between two instants), and *elements* (i.e. a finite union of intervals). In the context of SQL-92 (also called SQL2) [ANSI92, Conn02, Jens98], an *INTERVAL* type is a time duration of known length but with no fixed beginning or end instants. *DATE*, *TIME*, and *TIMESTAMP* data types are used to represent a date, time, and the combination of date plus time in SQL-92. Other units that can be found in the literature but are not listed in the standard temporal glossary [Jens98] include *time point sets* (i.e. sets of instants) and *time sequences* (i.e. ordered sequences of instants or intervals). Intervals can be represented as being open (end-points not included in the interval), closed (end-points included in the interval), bounded, or unbounded (without any end-points). This thesis adopts the temporal terminology and definitions of instants, intervals, and elements found in [Jens98].

The time interval is the most common time representation found in spatiotemporal research, partially because it has a natural symmetry with the interval projection representation often used for spatial data. As a primary means of representation and/or basis of temporal reasoning, instants have the disadvantage of leading to paradoxes and being less suitable for representing hierarchical time relations than intervals, as explained in [Allen83]. However, in some cases it is more natural to allow either an instant or interval to be associated with data. For this reason, instants are often represented by *chronons*, i.e. indivisible time intervals of a fixed, minimum duration determined by the application. Chronons are the smallest unit that can be represented in a discrete time model.

Regardless of the time unit employed, the granularity of timestamping is also an issue involving the usual trade-offs of precision and flexibility (at high-granularity) versus efficiency and simplicity (at low-granularity). Depending on the application, time dimension (type of time), and implementation, timestamps may be assigned to an attribute, entity (e.g. tuple, object), entity-aggregation (e.g. relation, object hierarchy), or even an entire schema. Attribute and entity timestamping are the most common. Timestamping entity-aggregations can result in unacceptable levels of redundancy unless only incremental representations are used, which would have implications for run-time efficiency as discussed in Section 2.2.1. Schema timestamping is used to maintain historical information regarding changes in database structure rather than contents: this has only been investigated in the temporal and not at all in the spatiotemporal context.

2.2.4 Integrating Space and Time

The majority of temporal research has been concerned only with thematic data; while the majority of spatial research has been ahistorical. A combined model of space and time may depend not only on the specific models of space and time adopted but also on the nature of the relationship between space and time being considered. For example, the relationship between space and time in a given application depends on the nature of the change processes (e.g. continuous, discrete) characterizing the application. Change processes are discussed later: here the focus is on different approaches to modeling integrated space-time data.

As discussed earlier, current spatiotemporal work assumes a 2D Euclidean space and unidirectional, linear model of time. Therefore, one spatiotemporal model would be an extension of Euclidean space with a temporal dimension, sometimes called a *space-time cube*. Researchers adopting this approach to spatiotemporal data modeling have usually assumed a 2D model of space and 1D model of time because of the difficulties of visualizing more than three dimensions. This approach has been adopted in [Arit97, Dimit97, Golsh94, Erwig99, Guti98, Worb95]. In particular, Erwig [Erwig99] and Güting [Guti98] explore the semantics of a generalized moving object using this approach. However, in general, the resulting semantics (e.g. types, data manipulation, integrity) of such a model are less well researched and understood than those of the component space and time models.

Another approach is a change-based model (sometimes called an event-based model) of space and time. In this model, spatial data and changes are not modeled continuously over time (i.e. along a time axis) but instead at discrete points corresponding to some type of transformation event (which may be instantaneous or have duration). Data is considered to be constant between events; therefore, only transformation events and their associated changes need to be modeled. This spatiotemporal model uses timestamps to indicate the time a change or event occurred, i.e. when a spatial data value is valid. In some cases, the changes may be modeled as separate temporally ordered sequences of events or versions for each data entity (e.g. image, object, attribute, etc.). This means that sequence information is maintained explicitly in the model either through forward links, backward links, or an ordered list. In this chapter, we will distinguish between the general change-based model with timestamped events or object versions [Beck96, Faria98] and the sequence-based model of change maintaining explicit ordering information [Clar95, Day95, Dion98] by using the terms *change-based* and *change-sequence* models respectively. A variation on the *change-sequence* model is one where the sequence of object versions is defined based on fixed time intervals rather than spatial transformation events. For multimedia applications involving video data, the spatial properties of an object can be recorded for each video frame regardless of whether they have actually changed between frames, as in [Day95]. Such models are called *time-sequence* models in this thesis.

Another variation on the conceptual spatiotemporal model represents an effort to address uncertainty or imprecision in spatiotemporal information through the use of relative temporal and/or spatial ordering of objects [DelB95, DelB96]. The models described previously assume a complete ordering of time and precise spatial locations. They allow an object, property, or event to be associated with a specific time and location. In some cases, it may be only possible or may be more appropriate to use a model of time and space which specifies only a relative spatiotemporal ordering, i.e. a space-time domain with direction and relative ordering but no specific measurements. A classic example would be in information retrieval functions, where similarity measures are designed to account for congruence in image or video data: metric measurement would tend to obscure rather than aid in the identification of similar patterns. We will refer to this space/time model using the term *relative ordering*.

Dimensionality

It is evident from the previous discussion that spatiotemporal systems are multidimensional. In fact, inclusion of more than one time dimension, three space dimensions, and/or possibly even different thematic attribute dimensions would result in more than three dimensions. Therefore, research in multi-dimensional systems (particularly with respect to indexing) can be directly applied to spatiotemporal systems.

However, space and time have unique properties, distinct from properties of thematic data, which require special consideration. In the same way that attributes are defined having a thematic domain, attributes can have spatial or temporal domains because their values have to do with space (e.g. property boundary) or time (e.g. appointment time) respectively. However, spatiotemporal data can also be a result of historical or location-based variation in thematic data. So, in addition to properties specific to space or time (e.g. space topology), it is important to note that thematic attributes will, in general, be dependent on time and, in some cases, may dependent on space as well. The value of any thematic attribute which is at all dynamic (i.e. subject to modification) will vary with time. The value of thematic attributes that are space related (e.g. geographical properties that vary with elevation or ground cover) will vary with space. Thus, thematic attributes can be represented as a function of time and possibly space. This effectively increases the dimensionality of fully integrated spatiotemporal-thematic systems.

Representation

Integrating time and space representations involves some of the same issues as with maintaining temporal records of thematic data, i.e. granularity and implementation issues related to the volume of data (e.g. storage, access efficiency, obsoletion policy). Implementation issues are outside the scope of this paper but are discussed in [Lang89, Lang93]. Granularity, however, relates to data modeling and representation. Temporal information can be associated with a whole spatial object and used to trace changes in an object's spatial properties over time. Alternatively, temporal information could be maintained for spatial sub-units of the object. The finest granularity would be an extension of the parameterization by thematic data described above for raster-based object representations, where the history of individual raster point values is maintained. If the parameterized thematic data is timestamped, then attribute values become a function of both time and spatial location. This is the approach followed in [Beck96], as described in Section 2.4.

A recent approach to representing spatial data in a very compact and space-efficient manner is using constraint databases, which represent a spatial extent as a conjunction of constraints on the coordinate values [Belu97]. For example, $2<=x<=5 \cap 3<=y<=6$ describes a rectangle with diagonal corner points (2,3) and (5,6). Spatial data is then queried using constraint manipulations. This work has been extended to spatiotemporal databases in [Chom97]. Actually, the constraint database approach could also be viewed as a conceptual model, i.e. a combination of the multi-dimensional *space-time cube* and *change-based* models described above. Erwig [Erwig99] compares constraint database approaches to the ADT model of spatiotemporal data. Constraint databases offer the advantages of dimension independence and consistent mathematical formalisms for data manipulation. However, two major disadvantages of constraint databases over ADT are described: (*i*) some important operations cannot be expressed because of the limitations of the formalism selected and (*ii*) this formalism results in a less natural (i.e. understandable) operational model from the user's perspective.

Another representation used to model sequential changes in a single object is called a *stream*. Streams are sequences of elements, where each element consists of a data value and associated timestamp. Data values may be composite and/or have spatial properties. In fact, Dionisio [Dion98] proposes a model that allows streams of streams. Streams have also been employed to represent information related to composition and presentation of multimedia artifacts such as video or multimedia documents. This is discussed further in [Dion98, Gros97].

2.2.5 Change Processes

Any temporal system can be viewed in terms of change processes. In the context of spatiotemporal systems, we must consider not only temporal changes in thematic data, but also changes in spatial data and composites. Spatiotemporal change processes can be categorized according to the unit (e.g. dimension and data type) and timing (e.g. discrete or continuous) of change.

For example, the unit of change may be an attribute, an object, or a composite (e.g. an event consisting of several objects in [Clar95] or a constellation of spatial sub-units in [Worb92]). The data type involved may be thematic and/or spatial, and if spatial, may be of different dimensions, e.g. 0D points or 2D regions.

The timing of change processes in a given application may be characterized as sudden and separately distinguishable transformations such as property divisions or gradual and incremental transformations such as shifts in soil acidity or air temperature. These two types of change processes are called *discrete* or *continuous* change respectively.

Discrete change events can be modeled as instantaneous, having duration, or having only a relative temporal ordering. Instantaneous changes are typically represented by associating each data version with a timestamp instant whereas discrete changes with duration have an associated timestamp interval (or timestamp instant with a specified duration). Relative temporal ordering can be represented by the use of version sequences or temporal logic expressions.

Continuous change is essentially a function from a time domain of instants to another domain, e.g. thematic values for a temporally dependent thematic attribute or spatial values (representing object dimensions) for a temporally dependent spatial object. For example, the metropolitan pollution index, a temporally dependent thematic attribute, can be considered a function from time to pollution index values. For a temporally dependent spatial object, the dimensions of the spatial object may be defined, e.g. a 0D point or 2D region. For spatially dependent thematic attributes, each domain value consists of a set of (spatial extent, thematic attribute value) pairs, where a spatial extent is a set of points in space (i.e. a spatial value). Therefore, a spatiotemporally dependent thematic attribute could be considered a function from a time domain to this composite domain. Alternatively, we could use a simpler function that maps an arbitrary time instant and spatial point to a thematic attribute value.

As pointed out in [Erwig99, Guti98], it is important to recognize that such abstract models of continuous change with arbitrary functional mappings are not practical to implement since computer representations must always be discrete. Other conceptual models that can be more easily translated into implementable designs may be required to approximate continuous change. One way to do this is to combine the discrete model of change with interpolation methods which can be used to return values between measured points.

Egenhofer [Egen92] and Yeh [Yeh92] consider possible interpolation methods for spatiotemporal data. Egenhofer [Egen92] analyzes the relative distance between different topological relationships of spatial objects based on the intermediate steps that would be required to transform one relationship to another. This relative distance is then used to predict the topological relationships likely to occur between two observed configurations. Yeh [Yeh92] discusses different types of interpolation useful for spatiotemporal data. For example, we could use a step function if the last measured value is considered to be valid; an average, minimum, or maximum of the neighboring points; linear or spline estimations; or specific functions based on the underlying mathematical model if known. Discrete semantics implies no interpolation, i.e. values can be returned only at measured points. However, finding appropriate interpolation techniques is not a trivial task, as pointed out in [Erwig99]. The difficulties can be illustrated by the challenge of interpolating between two recorded geometries of a moving region, which would not be easy to solve even for a specific application domain.

2.2.6 Data Manipulation for Spatiotemporal Data

Operators, queries, and query languages relevant to spatiotemporal systems are discussed in this section. Three types of operators (where operators include predicates) must be considered in addition to the traditional thematic operators: temporal, spatial, and spatiotemporal operators.

Temporal Operators

Temporal operators have been well researched in the context of temporal intervals and their binary relations [Allen83]. Thirteen basic interval relations have been defined and can be divided into the following three groups:

- before, meets, overlaps, during, starts, finishes,
- their inverses (after, met-by, overlapped-by, contains, started-by, finished-by), and
- equals.

In fact, these operators refer more generally to relations between any type of intervals and are also applied, for example, to interval projections of spatial objects' MBRs or MBCs on coordinate axes of Euclidean space. In fact, parallels can be drawn between Allen's relations and the topological relations (e.g. disjoint, touch, overlap) discussed in the next section. Both are concerned with relations between two objects. The difference lies in the dimensionality of the objects, i.e. 1D for Allen's relations and 2D or more for topological relations. Analogously, some metric operations (e.g. distance, duration) and set operations (e.g. union, intersection, difference) have applicability both for spatial data and temporal intervals. Additional temporal operators such as aggregation (summation over time), accumulation (averages over time), or interpolation (of intermediate or missing objects or attribute values) have also been mentioned in the literature [Lang89].

Spatial Operators

Spatial operators have been extensively studied for both fields and objects, especially in the context of GIS systems. Field-operations can be differentiated from object-based operations in that they exhibit the property of closure; i.e. both input and output are fields. Field operations are classified as local, focal, or zonal based on whether the output value at one location is dependent on input from the same location, nearby locations, or aggregated input from a larger set of locations (called a zone).

In contrast to field operations that always return fields, object operations can return different types of results including numbers, booleans, or sets of objects. They can be roughly categorized as being geometric, topological, set-based, or some combination thereof. Geometric operations involve a quantitative measurement. Some are dimension dependent, e.g. length for 1D objects, area for 2D objects, volume for 3D objects, and others are dimension independent, e.g. distance. Topological operations relate to the study of form or shape, including object and inter-object orientation. This includes (*i*) unary topological operators for a single object such as closed/open predicates (based on whether the object extent includes its boundary), boundary, interior, extreme points (for polygons and their 3D equivalents), and direction (i.e. orientation such as north, east, etc.) and (*ii*) binary topological relations between two objects such as disjoint, touch, inside, overlap, and equals. Finally, set-based operators include single-set operators (e.g. membership, cardinality, power-set), multi-set operators (e.g. union, intersection, difference, complement), and set-relation operators (e.g. product or function).

Spatiotemporal and Other Operators

Integration of spatial and temporal data requires an understanding of properties and operators that are not solely spatial or temporal, but are relevant only in the context of both types of data. There is a need for considerably more exploration in this area. Pequet [Pequ95] discusses grid-based spatiotemporal operators to retrieve locations having a given thematic value during an instant or period of time and to calculate the difference in extents over time for that value. Otherwise, most of the work in this area has been based on the object-based rather than the field-based view of space.

Spatiotemporal operators for a single object can be categorized as existential or transformational, and transformations further sub-divided into changes in shape, size, position, or some combination thereof. Existential operators include operators returning the value of an object or attribute at a given time and/or spatial location and related predicates related to object existence or attribute values.

Topological, metric (called geometric in the context of spatial data), orientation, or functional spatiotemporal relations between objects can also be considered. Topological changes in object relations over time include moving away, moving towards, or temporal sequences of object relations (e.g. first touched, then overlapped, then disjoint). Metric operators include such measures as the speed with which an object is moving. Güting [Guti98] systematically explores generic types and operations required to model spatial and temporal properties of data, especially those involving continuous movement of geometric entities.

Functional relationships represent one of the more interesting areas to be explored, as they can potentially incorporate a higher level of semantics than other spatiotemporal operators can. Claramunt [Clar95], in particular, considers the need to incorporate cause-and-effect relationships explicitly and considers operators representing processes such as production (i.e. generation of similar objects), transmission (i.e. transmission of properties to other objects), and re-structuring (e.g. union of two objects). The relevant functional operators would need to be considered in the context of a particular application domain and its semantics.

Other operators are dependent on the specific database model used, e.g. relational operators such as project and join, and may need to be extended for multidimensional queries involving time and/or space. For further discussion of the extension of relational operators see Langran [Lang89].

Queries and Query Languages

Query classes can be roughly divided into spatial, temporal, thematic, or combination [Arit97, Jain94]. Within these categories, queries can be further sub-divided based on whether they are point-based or range-based and by the specific operators required. Other classifications are possible. For instance, queries can be at different levels of granularity, e.g. object-based (during which periods did this object exist and in what shape?) or attribute value-based (what objects existed at this time in this region of space?).

Semantics, operations, and effective data structures for modeling and representation of integrated spatiotemporal data are current areas of active research and not yet well understood, so development of general or standard spatiotemporal query languages is largely an issue for the future. However, some initial efforts have been made to propose generic spatiotemporal extensions to use for a variety of query languages [Guti98], to extend SQL-based languages (e.g. TSQL2 or OQL) [Beck96], or to develop formal logic-based, algebraic-based, or constraint-based query languages as described in [DelB95], [Dimit97, Golsh94], and [Chom97] respectively.

2.3 MIR versus GIS Perspectives

Although there is considerable overlap and common interests between MIR and GIS research communities, some general distinctions can be made between their typical spatiotemporal research perspectives and interests. GIS are primarily intended to model the real world. While images, graphic maps, and other multimedia artifacts form an essential part of GIS, they are treated as means of discovering or recording actual geographic data. In addition, GIS systems contain spatiotemporal data that is directly entered without reference to any multimedia artifact, e.g. cadastral information related to changes in land deeds and their boundaries.

In contrast, the MIR community is primarily concerned with spatiotemporal data contained in multimedia artifacts and in the contents of the multimedia artifacts for their own sake, rather than just as a means of building an accurate model of the real world. The graphics and image sequences of a video or multimedia documents often represent the final product to be retrieved from the multimedia database by the information system. These artifacts are often deliberately constructed and their components re-assembled. Therefore, the MIR community has concerns, not relevant in the context of GIS, related to the synchronization and composition of the actual multimedia artifacts, e.g. the synchronization of audio and visual tracks in a video. These concerns are outside the scope of this thesis but are covered in [AlK99, Gros97]. However, the semantics of the constructed artifact resulting from the composition process is relevant in the context of this thesis. The construction represents a created semantics, e.g. video segments corresponding to meaningful scenes. Unless these semantics are explicitly associated with the artifact using manual annotations (as in [Zetts97]), semantic-based retrieval requires an extraction and mapping of the low-level features (e.g. color, texture) to higher-level objects.

Both communities share an interest in the spatiotemporal properties of the data represented (i.e. the contents) in multimedia artifacts; however, basic differences in the sources and intended uses of the data have implications for the preferred models, representations, and priorities of each community.

Researchers working in image and video retrieval generally assume that spatial data is input in terms of MBRs or MBCs and therefore rely on coordinate axes interval projection to represent spatial objects. The same technique can be applied to spatiotemporal objects if time is treated as another coordinate axis. Allen's interval relations can then be applied to both spatial and temporal intervals in a consistent manner and spatiotemporal operations decomposed into operations on constituent interval projections. This greatly simplifies queries compared to those typical of GIS and facilitates the development of relative order models based on temporal intervals, discussed in [AlK99].

In contrast, there are many GIS applications involving input data consisting of irregular spatial objects where the geometry and topology of the object is of specific interest. In some cases, researchers try to classify objects on the basis of pre-defined spatial classes using vector or raster representations. In other cases, it is assumed that the exact spatial extent of the objects is entered, e.g. for land deeds, where the exact dimensions and locations of property boundaries must be specified. Spatial and temporal data can no longer be treated uniformly as interval projections and the full complement of topological and geometric operations must be considered.

Another difference is that MIR spatiotemporal research is exclusively object-based, whereas both field-based and object-based approaches are required for GIS research. In addition, multimedia research specifically focuses on changes in spatial objects and their spatial inter-relationships over time; whereas GIS research is also concerned with changes in thematic data over space and time. There are differences in perspective even with respect to continuous changes in spatial objects, where MIR applications typically involve rapid and visible position change (i.e. real-time motion) and many—although not all—GIS applications model gradual changes in position, orientation, shape, and/or size. Therefore, there is generally more focus on motion trajectories in the MIR than the GIS research community.

MIR applications are primarily interested in retrieval based on some type of pattern matching, using low-level features such as color or texture derived automatically from image processing or higher-level semantics. Classifying image sequences or video in terms of their semantic content is generally manual or only semi-automated (e.g. with the help of knowledge bases in a limited domain). Queries are often based on examples. In the context of scene retrieval based on spatiotemporal properties, relative temporal order and spatial relations are generally more useful than quantitative criteria. This explains the importance of integrated space-time models and representations with explicit temporal ordering (e.g. sequence-based, stream-based, and graph-based representations) for MIR applications [Day95, DelB96, Dion98]. Another characteristic of MIR research is the use of multi-layer models and/or representations to facilitate the mapping of lower-level feature information to higher-level semantics; e.g. the two-level model described in [Day95]. In contrast, change-based models and representations with timestamps are more common in GIS research.

MIR time measures are often expressed in terms of frames or sampling intervals, especially in video research. For example, in [Dion98], time is expressed with reference to a time line internal to a specific object using time ticks. This time dimension does not fit into the categories usual for GIS applications, i.e. valid, transaction, and existence time. Rather than being concerned with multiple time dimensions and valid or transaction time, MIR applications are concerned with relative time ordering within the multimedia artifact itself. Essentially, an artificial time scale is used which expresses time relative to the multimedia artifact or a specific object. These characteristics of MIR multimedia information systems can be understood in terms of the importance of relative temporal order and the focus on multimedia artifacts discussed earlier. In contrast, the goal of GIS applications is to model the real world. Therefore, data is timestamped with a specific time value referencing a *universal* time line approximating real world time, i.e. a time line external to and shared by to all objects in the database. Specific application examples illustrating these differences are discussed further in the next section.

2.4 Recent Work in Spatiotemporal Databases

A comparison of recent research efforts will serve to illustrate current spatiotemporal research issues and directions. As is generally true of spatiotemporal research to date, all of the work discussed here is based on an Euclidean model of space and a linear model of time. The focus is on research proposals that consider both structure and manipulation data model components. The models described here are representative of the different research approaches that have been used in developing a high-level model for spatiotemporal applications in the context of MIRS and GIS.

Fundamental differences between the research projects can best be understood by comparing them with respect to the logical framework (conceptual data model) adopted and the intended application domain. As we will see, the application domain has clear implications for the space representation used, the time dimensions considered, and the integration of thematic data in the model. Whereas current approaches to spatiotemporal research can be classified based on the intended application domain and its underlying logical framework; other issues, such as data manipulation, are less related to fundamental differences in approach then to an individual researcher’s particular interests or focus.

Tables 2.1 and 2.2 summarize the characteristics of MIR and GIS models respectively, based on the issues discussed in Sections 2.2 and 2.3. An examination of the tables shows that the intended application domain is strongly related to the *space representation*, *time dimensions*, and *integration of thematic data* and related to the *space-time model* and *space-time representation*. All of the GIS applications shown except Güting [Guti98] offer a vector representation for spatial properties, all integrate thematic data, and some have more than one time dimension [Beck96, Clar95]. In contrast, image sequence and video applications generally use interval projections to represent spatial data [Arit97, Day95, DelB95], generally do not have any integration of thematic data (except [Dion98]), and all, except DelBimbo [DelB95], have a single time dimension based on artificial time. This is because the focus in image sequence and video processing is on identifying objects using MBRs or MBCs and tracing changes in the object's location or form over time with respect to multimedia artifacts. For the same reasons and as discussed in Section 2.3, space-time models and representations with explicit ordering are more common in MIR than GIS applications. Finally, the approach of integrating spatiotemporal data into current data models is common to both research communities. This will be discussed further with respect to individual projects below.

Table 2.1 Spatiotemporal Models for MIR

(Abbreviations: s. = spatial, t. = temporal, s.t. = spatiotemporal)

Author	Arit97	Day95	DelB96	Dimit97	Dion98
Application	image sequence	video	image sequence	video	multimedia
Data Model	none	OO	logic	algebraic	EER
Space-Time Model	space/time cube	time-sequence	relative ordering	space/time cube	change-sequence
Spatiotemporal Support	object	object	object	object	object & attribute
Thematic Data?	no	no	no	no	yes
Dimensions:					
time	1D	1D	1D	1D	1D
space	2D	3D	3D	2D	2D
Representation:					
space	interval projections	interval projections	interval projections	vector/raster combined	raster
time	interval	instant	relative order	interval	instant
integrated space-time (base s.t. unit)	composite x,y,t cuboid	directed graph segment node	frame/ sequence logic assertion	object motion trajectory	stream (sequence of entities or values)
Timestamp Granularity:					
artificial time	object	object	none	object	stream element
transaction-time	none	none	none	none	none
Query Language	no	no	logic / iconic example	algebraic/ trajectory example	logic
Operators					
metric	s.,t.	s., t., s.t.	no	yes	s.,t.,s.t.
topological (s. & s.t.)	s.	s.	s.: expressible	s.	s.
orientation (s. & s.t.)	no	no	no	s.	no
Allens (intervals)	s.,t.	s., t.	s., t.	t.	no
set	s., t.	s.	no	yes	yes
transformation	yes	no	no	no	no
functional	no	no	no	no	no
other	no	no	no	composition	stream

Table 2.2 Spatiotemporal Models for GIS

(Abbreviations: s. = spatial, t. = temporal, s.t. = spatiotemporal)

Author	Beck96	Clar95	Guti98	Faria98	Worb92
Application	GIS	GIS	GIS++	GIS	GIS
Data Model	OO	relational	any (ADTs)	OO	OO
Space-Time Model	change-based	change-sequence	space/time cube	change-based	space/time cube
Spatiotemporal Support	attribute & composite	object	object & attribute	object	object
Thematic data?	yes	yes	yes	yes	yes
Dimensions:					
time	2D	2D	1D	1D	1D
space	3D	2D	2D	2D	2D
Representation:					
space	vector/ raster	vector	functional/ set-based	vector	vector
time	interval or instant	interval or instant	instant	instant, interval, or element	interval
integrated space-time (base s.t. unit)	object class	linked relational tuples	abstract data type	object class	composite x,y,t right prism
Timestamp Granularity:					
valid-time	attribute (or raster pixel)	object & event	not applicable (continuous)	object	spatial unit (object component)
transaction-time	object	object	none	none	none
Query Language	OQL/ TSQL2	no	any (extend w/operators)	no	no
Operators					
metric	s.	s.	s.,t.,s.t.	s., s.t.	s.
topological (s. & s.t.)	s.	s.	s.	s., s.t.	s.
orientation (s. & s.t.)	no	no	no	s., s.t.	no
Allens (intervals)	t.	t.	no	t.	t.
set	s.	s.	no	no	s.
transformation	no	yes	no	no	no
functional	no	yes	no	no	no
other	no	no	no	no	no

An exception to the general trend of extending existing conceptual data models, Aritsugi [Arit97] considers only the representation of spatiotemporal data. Changes in an object's shape, size, position, or existence over time are modeled graphically as a composite MBC (cMBC) with each change represented by a separate MBC on the x, y, and time axes. Each MBC is equivalent to a single MBR extended along the time axis to represent the time period for which the object had that shape. Spatial, temporal, and spatiotemporal queries can then be answered by a sequence of operations on spatial or temporal interval projections. In addition to projection operators, Allen's interval relations, metric operators (start, finish, difference for intervals; area, gravity, difference for MBRs), and basic topological operators for MBRs (disjoint, meet, contain, overlap, equal, direction) are also provided. An additional operator is used to merge temporal interval projections from different spatial versions of an object to answer questions about object (as opposed to version) existence. This method of representation is clearly more optimal for object-based queries (e.g. when did a given object exist?) rather than time-based queries (e.g. what objects existed during a given time period?).

Day [Day95] has a similar approach to that described in [Arit97], i.e. storing a sequence of time and space coordinates for each object. However, computational efficiency is optimized at the cost of significantly increased storage by: (*i*) specifying spatial coordinates of all extant entities at sampled time intervals rather than only after a spatial transformation, (*ii*) pre-computing the total duration of each object, and (*iii*) pre-computing inter-object topological relations at sampled time intervals. Since topological relations and time durations are pre-computed, there is no need to rely on topological operators or an interval merge operator to answer existence queries during run-time. The organizational approach is more sophisticated, with a directed graph used to index objects (and object versions) by video segment, defined by having a consistent set of extant objects throughout. The appearance of a new object indicates the start of a new video segment. Again, the representation has implications for query efficiency, especially for queries that span multiple video segments. Day [Day95] describes operators similar to those in [Arit97], except for the omission of the merge operator and the addition of spatiotemporal metric operators (displacement and speed). A method of specifying temporal predicates is also provided to allow identification of a video sequence exhibiting a particular sequence of spatiotemporal relations between objects.

An important advantage of the approach described in [Day95] is the attempt to provide a conceptual as well as a representational data model and a mapping between the two. An OO model consisting of a user-defined abstraction hierarchy of logical objects and a schema mapping from logical to physical (i.e. graphed) objects is used to express application-specific semantic spatiotemporal concepts and manipulate data using these concepts. For example, *sinking a basketball* would be mapped into two physical objects (the net and the ball) and related by the topological operator *contain*.

Another MIR model based on change-sequences focuses on change in composite entities and continuous change processes rather than on changing topological relationships. Dionisio [Dion98] generalizes a time-based stream model for multimedia and simulation applications. An ER model extended with OO features such as complex attributes, aggregation, and methods is extended further with multimedia and stream types. Multimedia types include image, image sequence, video, free text, audio, and speech.

Stream types, which are either discrete or continuous, consist of timing information for the stream and data elements. The data elements in a discrete stream consist of finite sequence of ordered (i.e. indexed) and timestamped data values. Data values corresponding to intermediate times may be assumed to be stepwise constant (e.g. between frames of a video) or be determined by an interpolation function (e.g. for simulation). The data elements in a continuous stream consist of a function from a closed, bounded interval of indices to an infinite number of timestamped data values. Special stream operators are defined to return new stream(s) whose elements satisfy specified predicate(s).

Time is modeled relative to an individual stream, i.e. the time associated with a data value is in ticks or seconds elapsed since the beginning of the stream. The stream's timing information maps ticks to seconds (i.e. frequency of ticks per second). The data value can be atomic or composite, where each composite could be of multimedia or stream type. In this way, streams of streams can be defined. A multiple stream type for synchronization of a set of streams is also defined, consisting of synchronization information and the stream set.

Aritsugi [Arit97], Day [Day95], and Dionisio [Dion98] all store specific time values for object versions. In contrast, Del Bimbo [DelB95] records only relative temporal orderings and provides a model that easily copes with both relative and exact specifications of spatial data. Spatial object data is recorded using a variation of interval projections (expressed in terms of regions). A spatiotemporal logic is described using static assertions to represent object locations and spatial relationships within a frame which are then combined into temporal sequence assertions using temporal logic (i.e. booleans and temporal operators such as *until*, *eventually*, and *always*). Relative ordering of space or time can easily be expressed using this logic. Since the work is directed mainly towards content-based image retrieval, this relative ordering facilitates matching pictorial query to database sequences. This is especially important for the query by example facility, where graphic sketches of desired image sequences are automatically converted to spatiotemporal logic assertions and compared for similarity with existing assertions in the database. As discussed previously in Sections 2.2.4 and 2.3, the use of exact measures would tend to obscure rather than aid in identification of similar spatiotemporal patterns, especially considering the additional imprecision introduced by the use of graphic rather than image data. The major contribution of this work is providing a formal language for expressing spatiotemporal concepts. However, there are serious questions as to processing and storage efficiency of a logic-based approach to spatiotemporal data modeling, especially as regards algorithms used to match logic assertions. In addition, the relative ordering used by this model precludes the possibility of metric measurements for time.

In later work, Del Bimbo [DelB96] introduces spatial and temporal markers (a point on an object and a frame index respectively) to increase the expressiveness of his model by allowing qualitative expression of metric distance relations in space (e.g. near, far) and time (e.g. soon, not-soon). Spatial properties of objects are then represented by a set of representative points instead of interval projections. However, this is an awkward solution since representative points and markers must be explicitly specified and raises further questions of efficiency depending on the number of representative points used for an object. Furthermore, since no exact measurements are stored, only qualitative metric measurements are possible.

The most significant difference between the model proposed in [Dimit97, Golsh94] and other MIR models reviewed, is that spatiotemporal video semantics is captured in terms of object trajectories. The trajectory curve is interpolated and trajectory attributes (i.e. path length, velocity, and acceleration) estimated from point samples, assuming both discrete objects with rigid boundaries and continuous motion. The trajectory is calculated using the object centroid. Knowledge-based inferences of higher-level motion semantics are then used to associate object trajectories with domain-specific activities such as driving a car in a straight line. To do this, the activities of composite semantic entities consisting of multiple spatial object sub-parts are considered by combining information on the sub-parts' relative spatial positions, relative timing of their motion trajectories, and interactions. An algebraic query language is defined consisting of system defined object types and operators and user defined (domain specific) object types and operators, where a set of object instances is associated with each specified type. Formal algebraic expressions can then be composed based on a set constructor specifying the objects of interest and selection conditions, including spatial, temporal, and spatiotemporal selection criteria. An iconic visual query language based on the same algebraic framework allows users to specify queries based directly on an input (i.e. sketched) object trajectory.

Worboys' work [Worb92], although for GIS rather than MIR applications, is most similar to that proposed by Aritsugi [Arit97], in that he proposes that queries be decomposed into operations on spatial and temporal projections. Since the spatial data investigated consists of irregular polygons rather than MBRs, spatial properties are represented by vectors instead of interval projections and an object version is represented by a right prism rather than a cuboid. In addition, Worboys [Worb92] provides a more comprehensive model at a higher conceptual level which can model spatial and thematic aggregation (including collections) and inheritance. An OO model is used to represent a real world object consisting of multiple spatial sub-units and thematic attributes and to allow definition of new object-types using inheritance. He also gives an extensive classification of 0D-2D spatial object types and operations including a full range of topological, set-based, and geometric operators.

Since queries cannot be completely decomposed into operations on space and time interval projections in models intended for the GIS domain, other operations must be explicitly provided to describe spatial and spatiotemporal manipulations. Hence, the development of higher level operators is important for these applications. Whereas the previous models discuss only interval intersection, Worboys [Worb92] includes a more complete range of set-based operators (i.e. equality, membership, subset, intersection, union, difference, cardinality). Additional topological and geometric operators relevant to a vector-based rather than interval-based space representation, such as interior, boundary, extremes, inside and perimeter, are also discussed. For the same reason, Allen's interval relation operators are only provided for time and not space. Overall, Worboys [Worb92] clearly provides a more comprehensive conceptual model. However, a weakness of this model is the oversimplifying assumptions that thematic attributes are constant throughout spatial components and that only spatial components have temporal properties. No provision is made for modeling historical thematic data.

In [Faria98], spatial and temporal properties are added to an object class definition by associating it with pre-defined temporal and spatial object classes. As in [Worb92], this solution is not suitable for representing spatially or spatiotemporally dependent thematic attributes, as the timestamp and spatial locations are defined only at the object component level. Operators are defined for retrieving temporal properties of an object and spatial properties (including metric properties) of an object, possibly at a specific time. Temporal operators based on Allen's interval relations [Allen83] and metric, orientation, and topological spatial and spatiotemporal operators are defined for pairs of objects.

Becker [Beck96] proposes an OO model with more powerful spatiotemporal modeling than in [Faria98, Worb92]. It supports flexible association of temporal properties with spatial and thematic data. In [Worb92], a time interval can be associated only with a whole spatial unit (which may be a sub-unit of an object). Temporal concepts are generalized and formally incorporated into the OO model in [Beck96] by allowing timestamps to be associated (*i)* with any attribute, whether simple or complex (i.e. aggregated objects), spatial or thematic, or (*ii*) with any relationship between objects. Timestamps can be either intervals or instants. Furthermore, temporal, thematic, and temporal-thematic properties can be specified across spatial extents through parameterized raster representations of space. This is achieved by associating thematic data with each byte location of the raster array. Temporal variation is then modeled through timestamps associated with spatial and thematic data at each raster array location. This means that a single raster location can be associated with both historical spatial contents (e.g. indicating intensity and color at that location) and associated thematic information. Vector representations are used to model cases where thematic data is constant within each spatial version of an object.

An extensive hierarchy of base spatial classes are pre-defined for zero to three dimensions (0D-3D) and for both vector and raster representations in [Beck96]. Spatial application classes can then be defined as extensions of these classes through the use of inheritance and method redefinition. Both object-based and field-based spatial models can be accommodated through the use of geometric and parameterized thematic attributes. Basic topological and geometric, center, bounding-box, and intersection operations are extended for 3D objects (e.g. volume) and Allen's interval relations are incorporated for time intervals.

Erwig [Erwig99] and Güting [Guti98] systematically explore generic types and operations required to model spatiotemporal data, especially data involving continuous movement of geometric entities. Both spatiotemporally dependent attributes and temporally dependent spatial objects are supported. They define ADTs to be used to extend existing models with spatiotemporal semantics. Functions are used to define the semantics of operations on moving objects.

Claramunt [Clar95] uses the relational model to integrate thematic, temporal, and spatial data organized using the principal of an event-based model of change. Thematic, spatial, and temporal data are stored in separate tables with bi-directional domain links between these tables to represent objects and events. Temporality (i.e. temporal data) is represented through the combined use of object versions and events, stored in separate tables. Therefore, the thematic properties of objects, the spatial properties of objects, temporally ordered objects with timestamps, and temporally ordered events with timestamps are each represented in a separate relational table. Each object version is represented as a separate tuple in the object table with domain links to the spatial and thematic attributes associated with that version. Each event is represented as a separate tuple in the event table with bi-directional links to the object-versions involved in that event.

The event-based view of time assumes that every new object version is the result of some event, which may or may not involve multiple objects. Therefore, the event is adopted as the central organizing principal and timestamps are associated with both events and object versions. Descriptive information associated with each event serves to explicitly record the functional relationships underlying the creation of the resulting object versions. In this way, time is associated only indirectly with attributes through object-version links.

Past, present, and future data are stored in separate tables to ensure that the volume of historical data does not affect access to current data. This is the only one of the spatiotemporal models discussed to consider future time; however, it still assumes a linear model of time since it requires that different versions of an object be discrete.

Spatial and temporal operations are mentioned; however, the focus is on spatiotemporal operations including object transformations and inter-object functional relationships. Spatiotemporal transformation operators include single-object transformations and multiple-object restructuring. Single-object transformations are further subdivided into changes in shape (e.g. deformation), size (e.g. shrink, expand), position (e.g. move), or existence (e.g. appearance, disappearance). Restructuring includes union, split, and re-allocation. Functional operators are used to model the underlying semantics of inter-object relationships. A set of functional relationships classified as replacement or diffusion functions are enumerated specifically for describing event cause-and-effect semantics in GIS applications.

This spatiotemporal model is unique in its ability to model the underlying causes of change. However, it may be awkward for those application domains where certain attributes may vary continuously in time (or where the underlying functional relationships causing change are not well understood or extremely complex) to model each change as a separate event and object version.

Timestamp Granularity and Time Dimensions

Of all the projects using timestamps, only Becker [Beck96] and Dionisio [Dion98] are able to timestamp at the attribute level (equivalent to the stream element for [Dion98]). Becker [Beck96] uses attribute level timestamps for valid-time. These include timestamps for complex and multi-valued attributes (i.e. timestamping class relationship links and attribute collection members respectively). Claramunt [Clar95] timestamps at the object and event level and Worboys [Worb92] at the object component level. All others timestamp only at the object level, i.e. each different object version has a timestamp.

Only two papers [Beck96, Clar95] address the issue of multidimensional time. Becker [Beck96] distinguishes between suitable levels of timestamp granularity for valid (attribute level) versus transaction (object level) time. He recognizes, on the one hand, the need to minimize redundancy for versions involving change in just one or two attributes and, on the other, the need to facilitate rollback and recovery and take advantage of implemented OO versioning mechanisms to implement transaction timestamps. Obviously these justifications are somewhat contradictory. Application domains where transaction time is potentially queried (rather than just being used for recovery) might require a finer level of granularity for transaction timestamps as well.

Claramunt [Clar95] associates two different timestamps with each object version. However, only one sequence of object versions is defined by the version links, so there is no provision made for possible differences in ordering in valid versus transaction time. By assuming that the ordering in both time dimensions is identical, the value of the modeling technique for multiple time dimensions is effectively weakened.

Operators

With the exception of Del Bimbo [DelB95], all of the models provide some metric operators. Day [Day95] considers spatial, temporal, and spatiotemporal metric operators (e.g. displacement and speed). Basic topological operators and the set-based intersection operation are considered by most of the authors, although they would have to be expressed using logic assertions in [DelB95]. The most comprehensive discussion of geometric and topological operators can be found in [Worb92]. All models use Allen's relations for time. Research in multimedia application domains, which uses interval projections to represent space, also applies the relations to time.

The most extensive discussions of spatiotemporal operators, including higher level operators incorporating underlying spatiotemporal semantics, are provided in [Clar95, Faria98, Guti98]. The introduction of functional relationships in [Clar95] is based on the claim in [Pequ95] that timestamps alone are not sufficient for representation of underlying spatiotemporal semantics of GIS applications involving cause-and-effect data relationships. Instead, there must be explicit logical representation of the transformation events themselves including information on the entities involved, event duration, and the nature of the underlying transformation process. Pequet's work [Pequ95] is based on a field-based spatiotemporal model. Claramunt's primary contributions [Clar95] are (*i*) providing an analysis of which functional relationships were important for GIS and (*ii*) incorporating an explicit model of events into an existing object-based spatiotemporal model. Güting [Guti98] and Erwig [Erwig99] provide generic operators for moving objects. Faria [Faria98] focuses on retrieval of object geometries and metric measurements at a given time and predicates relating to the orientation or topological relationships of two objects at a given time.

Representation, Query and Query Languages

Representational organization of spatial, temporal, and, in some cases, thematic data affects not only the range of query types that can be answered, but may also affect the relative efficiency for different query types. A detailed discussion is beyond the scope of this thesis, but an example will serve as an illustration. For example, a representation that facilitates retrieval of object inter-relations at a particular point in time, as in [Day95], will be less suitable for tracing change in the spatial characteristics of a given object over time. The latter task might require search and retrieval across non-contiguous graph segments in [Day95], but would be contained in a single cMBC in [Arit97] or composite right prism in [Worb92].

Spatiotemporal query languages developed to date tend to reflect the application domain and functionalities of interest to the given researcher. The development of a formal logic-based spatiotemporal query language in [DelB95, DelB96] is very expressive and highly suitable for pattern-matching. However, it is not suitable as a user query language due to the complexity of query formulation and raises questions of efficiency even as a basis for automated icon-based description generation and similarity matching. The textual and visual algebraic query languages defined for the spatiotemporal model proposed in [Dimit97, Golsh94] are similar to those proposed in [DelB95, DelB96] in that their focus is pattern-based retrieval from a multimedia database, although the query language in [Dimit97, Golsh94] further provides for video composition and editing. The query languages described in [DelB95, DelB96, Dimit97, Golsh94] require some degree of manual user assistance to specify the spatiotemporal semantics of the data, which is problematic in the case of large multimedia databases. This user assistance involves drawing graphic sketches that can automatically be converted to spatiotemporal logic expressions for each catalogued or query image in [DelB95], selecting representative point markers in [DelB96], and specifying the user-defined data types and operators for a given domain in [Dimit97, Golsh94]. In addition, these query languages share the same problem typical of formal algebraic or logical query languages: users find it difficult to formulate or understand logic or algebraic query expressions as noted in [AlK99].

Another more intuitive query option is query by example, involving similarity matching between query and catalogued graphics or images. In the spatiotemporal context, a query example can consist either of several graphic sketches showing discrete changes in relative object positions and inter-relationships [DelB95] or sketched trajectories representing continuous object motion [Dimit97, Golsh94]. Other spatiotemporal models not reviewed here that allow trajectory-based retrieval from video are discussed in [AlK99, Yosh99]. Of these, the model proposed by Dagtas [Dagt98] is notable in that it allows specification of multiple object trajectories using petri-nets whose places represent individual object trajectories and hierarchical organization reflects the relations [Allen83] between their associated temporal intervals.

Although more intuitive for the user than the use of formal query languages, measuring similarity based on examples can be more difficult since the specific information needs of the user are not explicitly expressed. For example, if two objects move apart and change orientation in the example sketches, are both changes relevant to the specific user query?. Furthermore, *query-by-example* is suitable for only certain classes of spatiotemporal queries relating to changes in spatial objects and would not be suitable, for example, for queries based on changing values in spatially distributed thematic attributes.

In contrast, Becker [Beck96] proposes a textual query language, T/OOGQL, designed to facilitate formulation of spatiotemporal queries on both objects and attributes. A spatial query language based on ODMG's proposed OO standard Object Query Language (OQL) was extended in a manner analogous to TSQL2's temporal extensions to SQL, i.e. valid-clause for valid-time projections, select clause *snapshot* option for non-historical queries, and temporal predicates for use in the where clause. Temporal properties are incorporated in the data definition language component of T/OOGQL by adding instant and interval timestamp keywords. The ADTs in [Erwig99, Guti98] provide a formal basis for an extension to existing query languages.

Space/Time Integration with Thematic Data

Because GIS applications often involve changes in thematic attributes across space and time, logical frameworks incorporating thematic attributes with spatiotemporal data are found in all of the GIS models. Object aggregation (i.e. whole-part relationships between objects) is used to add temporal and/or spatial properties to traditional object classes with thematic data in [Faria98]. Similarly, Worboys [Worb92] includes only minimal incorporation of thematic data into the logical OO framework with no treatment of spatial or temporal variation of thematic data. Thematic data can be associated with an object; however, thematic attributes are assumed to be constant across spatial and temporal extents and across different object versions. Claramunt [Clar95] provides a higher degree of integration by associating thematic data with different object versions. This provides a mechanism for modeling temporal variation in thematic data at least at the level of object versions. Becker [Beck96] and Güting [Guti98] provide the highest degree of integration by providing for thematic data history at the attribute (rather than object) level and incorporating thematic variation across space as well as time in the OO model. Although Day [Day95] does not address the need to integrate thematic with spatiotemporal data, he does provide a logical framework intended to allow users to specify domain specific concepts and map them to spatiotemporal properties of video objects.

Data Models

Comparing the choice of conceptual data models used for spatiotemporal data, object-based models (OO, ER, EER, ADTs) are the most common, especially for GIS. The relational model does not offer the same range of modeling constructs, especially with respect to aggregation, collections, and inheritance. These features are of potential importance for describing the complex data characterizing spatiotemporal research. The logic model, while it offers important advantages for reasoning and for expressing imprecision or uncertainty, is cumbersome to use as a data model. Both the algebraic and logic models have limitations with respect to their practicality for large databases and usability for naive users.

2.5 Current State of Spatiotemporal Modeling Research

Considerable progress has been made in developing high-level conceptual models integrating spatial, temporal, and thematic data: object-based models have played a dominant role in this context [Beck96, Day95, Dion98, Faria98, Guti98, Worb92].

Several models provide a high level of integration and support both spatiotemporal objects and attributes [Beck96, Dion98, Guti98]. Support for continuous as well as discrete models of spatiotemporal change is much less common but has been addressed in [Dimit97, Dion98, Erwig99, Golsh94, Guti98]. Most research efforts address the issue of spatiotemporal modeling and data manipulation either from the GIS or the MIR perspective; however, Güting [Guti98] and Erwig [Erwig99] adopt a generic approach intended to identify fundamental spatiotemporal types and operations from any application domain.

Although there has been increasing focus on the development of high-level spatiotemporal data models, none of the models reviewed has as a priority support for the early stages of application development, i.e. requirements analysis and design. For example, consider a high-level conceptual framework such as those found in [Beck96, Erwig99, Guti98] that integrate a range of spatial, temporal, and thematic data and include a textual high-level query language capable of specifying spatiotemporal entity types. Although the data definition component of the query language has some potential for use in modeling spatiotemporal applications, the non-graphical query language of this model reduces its suitability for conceptual modeling.

In a clear indication that the need for a spatiotemporal graphical modeling language has been well-recognized, there have been several concurrent efforts to develop such a language recently reported in the literature. The MADS model [Paren99] extends a hybrid ER and object-based model with pre-defined hierarchies of spatial and temporal abstract data types represented by icons and special complex data types. A spatiotemporal ER model, STER, is proposed in [Tryf99] that adds temporal and spatial icons to entities, attributes, and relationships to support timestamped spatial objects and attributes. Peceptory [Brod00] is a spatiotemporal model and CASE tool aligned with both geographic and OO standards, based on adding spatial and temporal stereotypes to objects in UML. All four of the models—these three models and that proposed in this thesis—offer a graphical query language for spatiotemporal data modeling and concentrate on structural rather than data manipulation aspects. However, they differ with respect to the specific features supported and modeling mechanisms adopted. The differences will be discussed in detail in Chapter 3 as a comparison to the spatiotemporal graphical modeling language proposed in this thesis and described in that chapter.

With respect to the data manipulation component of modeling, more work is needed to come up with a comprehensive approach to spatiotemporal data manipulation and to synthesize or reconcile domain-specific and generic views of data manipulation. Although defined in the spatial database community, to our knowledge, field-operations such as those described in [Worb95] and discussed in Section 2.2.6 have yet to be incorporated in a spatiotemporal model. A related topic is support for continuous models of spatiotemporal change. Although some initial efforts have been made to provide support for continuous change using functional definitions for operations [Dion98, Erwig99, Guti98] and interpolation [Dion98, Yeh92], this is very much an open problem requiring further research.

Other issues have been raised for consideration [Barr91, Haze91, Jain94, Lang89], but have yet to be seriously addressed in the context of spatiotemporal systems, e.g. integrity, alternative time models, views. Integrity aspects of spatiotemporal data models have received minimal attention to date. Research into classes of spatiotemporal integrity constraints will be required before this lack can be remedied.

Finally, as both are essentially multimedia information systems, GIS and MIR systems have considerable overlap and could benefit from more exchange between the different research communities. It is true that different application perspectives may lead to different answers and priorities, as illustrated by the comparisons between spatiotemporal research in the MIR and GIS communities. However, reviewing research efforts across application domains represents an important part of the process contributing to a more complete understanding of spatiotemporal data modeling.

2.6 Summary

In this chapter, we reviewed the fundamental issues relevant in the context of spatiotemporal data modeling, compared and contrasted data models from GIS and MIR communities with respect to those issues, and identified current research trends and priorities. The concepts discussed in this chapter serve as a reference for the rest of the thesis and provide the rationale for the research focus of this thesis and the specific research approach adopted.

Specifically, we noted two notable themes in spatiotemporal data modeling research:

- the move towards integration of different data types and representations in one model to provide general support for spatiotemporal applications, and
- the increasing use of higher-level data models and the OO data model in particular.

It was further noted that open issues include the development of high-level graphical spatiotemporal languages, integrity, views, alternate time models, and data manipulation.

This thesis concentrates on the first two research areas mentioned above: graphical language development and—in the context of composite spatial objects—integrity constraint specification, with the goal to provide support for analysis and conceptual design of spatiotemporal applications. The specific approach adopted is motivated by the recognition of the advantages of the OO data model and by the requirement for an integrated model providing general support for a range of spatiotemporal applications. The proposed conceptual modeling techniques utilize the standard OO graphical modeling language UML and support a range of spatial, temporal, and spatiotemporal data types and representations. As will be discussed in Chapters 3, 5, and 6 respectively; the thesis proposes a spatiotemporal extension to UML, a framework and modeling constructs for representing common types of complex spatial object configurations differentiated by their implied semantic constraints; and a technique for specifying topological relationships (i.e. constraints) in such configurations.

Chapter 3

A Conceptual Modeling Language for Spatiotemporal Applications

3.1 Introduction

This chapter addresses the problem of developing a graphical data modeling language to support requirements analysis and conceptual design of spatiotemporal applications. From the discussion in the preceding chapter, it is clear that spatiotemporal applications vary considerably with respect to the types of spatiotemporal data managed and assumed models of space, time, and change. Therefore, in order to serve as a general tool for spatiotemporal application design, any such data modeling language should provide a clear, simple, and consistent notation to capture the semantics of different spatiotemporal data types and alternative models of space, time, and change. In particular, the language should be able to model the following spatiotemporal data types:

- temporal changes in an object’s spatial properties (*temporally dependent spatial objects* and *temporally dependent spatial attributes*),
- variation of thematic attributes across space and/or time (*spatially, temporally*, and *spatiotemporally dependent thematic attributes*),
- composite data whose components depend on space and/or time (*spatially, temporally*, and *spatiotemporally dependent associations*).

The language should also support instant-based and interval-based time semantics; object-based and field-based spatial models; and instantaneous, discrete, and continuous views of change processes. Multiple dimensions for time and space should also be supported.

In this chapter, we propose an extension of UML intended to address these goals. Extending the OMG standard for OO modeling [OMG99] was selected as the best approach given its high level of acceptance, tool support, understandability, and extensibility. Although the applicability of the proposed model is not necessarily limited to the GIS domain; the focus is primarily on GIS concerns and application examples in this thesis. We introduce a small base set of modeling constructs for spatiotemporal data that can be combined and applied to different levels of the object-oriented model in a consistent manner, guided by the same simple principles. The result is *SpatioTemporal UML*, abbreviated as *STUML*. A formal functional specification of the semantic modeling constructs and symbolic combinations are provided in the chapter.

The chapter is organized as follows. UML is used to illustrate the problems of using a general-purpose data modeling language for spatiotemporal applications. To this end, Section 3.2 gives a brief overview of UML. Section 3.3 illustrates the problems of modeling spatiotemporal data with UML, using a regional health application example; considers possible solutions; and justifies the approach adopted in this thesis. Sections 3.4 through 3.6 present the solution proposed in this thesis: a spatiotemporal extension to UML called STUML. Section 3.4 describes the syntax and semantics of the fundamental new constructs introduced—the *spatial*, *temporal*, and *thematic* symbols. Section 3.5 discusses three other symbols: the attribute group symbol (used to model common spatiotemporal properties), the existence-dependent symbol (used to model temporal constraints between an object and its attributes or relationships), and the specification box (used to specify details of the spatiotemporal semantics). Section 3.6 shows how the regional health application example described in Section 3.3 can be modeled using STUML and compares it to the UML schema from Section 3.3. Section 3.7 compares STUML to three other graphical spatiotemporal languages reported in the literature. The chapter is summarized in Section 3.8.

3.2 An Overview of UML

UML [Booch99, OMG99, Rumb99] consists of (*i*) nine types of diagrams specifying structure or behavior of a system and its data elements, (*ii*) notational conventions based on the OO paradigm, and (*iii*) built-in extension mechanisms for supplementing UML's core meta-model. Class diagrams and their specialization, Object diagrams, describe the structure of classes and object instances, respectively. Component diagrams describe implemented system software component structure and dependencies. Deployment diagrams describe the hardware and software configurations of delivered systems. Behaviors within the business process model are described by Use Case (system functionality), Sequence, and Collaboration diagrams. The latter two diagrams give a chronological versus procedural view of object interactions within a single system function. Class behavior is specified by State and Activity diagrams for external and internal events, respectively.

The diagram most relevant to the support of spatiotemporal data modeling is the Class diagram, since it captures the static structure of a database design. Those Class diagram components that are pertinent to this thesis are illustrated in Figure 3.1. Unless otherwise noted, the UML usage and notation employed in this thesis is based on Rumbaugh [Rumb99]: readers are referred to that source for further details of the Class diagram not shown in Figure 3.1.

The fundamental element of this diagram is the object *class description*, consisting of the class name and its properties, i.e. attribute descriptions and operation signatures. Attribute descriptions consist of the name, multiplicity (assumed to be *exactly one* if not specified), types[1], and default values (optional) for class attributes, where an attribute type can be an atomic or complex data type (whose values do not have a separate identity) or a class. In this thesis, we follow the default UML attribute string syntax specified in [Rumb99, p.168]. UML multiplicity and complex data type expressions from a standard programming language such as C or C++ can be used to specify attribute domains, as described in [Rumb99, p. 248]. However, we prefer to use informal descriptions of attribute domains in this thesis, for the sake of readability. Operation signatures consist of the name, parameters, and return types for class operations. Visibility (scope) can be indicated for attributes or operations.

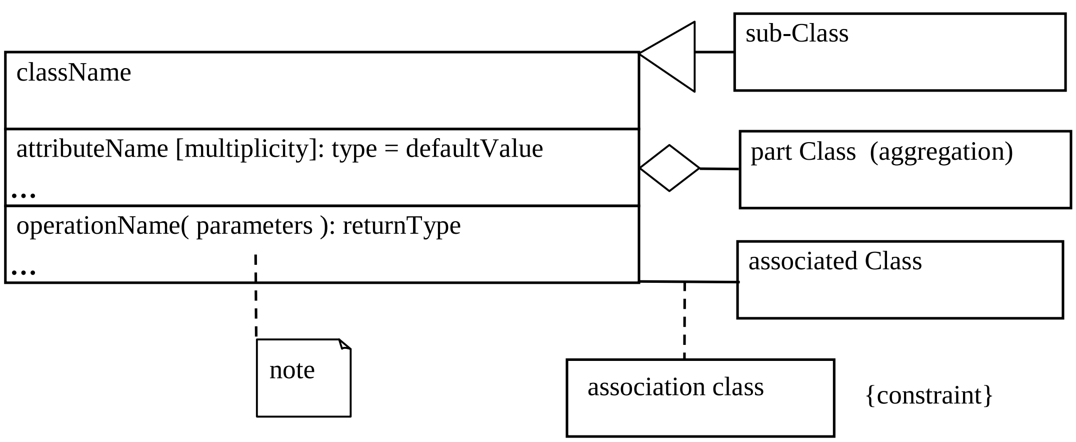


Figure 3.1 Class Diagram Legend

Classes can be connected by different types of standard OO links, including *generalization* (sub-classes defined based on a super-class) and *association* (semantic relationships between object classes). Generalization facilitates incremental design and extensibility through inheritance of common properties (i.e.

[1] The terms *type* and *domain* are treated as synonyms for the purposes of this thesis.

attributes and operations) from existing classes. In contrast, association represents semantic connections between instances of classes whose properties may be completely different. Associations, like classes, will be instantiated and their instances must be uniquely identifiable.

Associations are similar to attributes, except that the domain is instead modeled as a separately defined class and typically uses reference-based data semantics (i.e. identifiable and shared instances) rather than value-based data semantics (i.e. owned and copied instances). An association is usually represented by a line between classes. However, an association can also have its own properties, e.g. attributes. In this case, the association is *promoted* to a special *association class* that is then connected by a dotted line to the original association line. Specializations of associations with specific properties exist. *Aggregation* is used to describe an association relationship with part-whole semantics. *Composition*[2] is an aggregation relationship further constrained in having the part instance owned by only one composite object at a time, i.e., part instances cannot be shared. The representation for *composition* is identical to that of *aggregation* except that a black diamond is used instead of the white diamond shown for aggregation in Figure 3.1.

Constraints may be enclosed in curly braces. Notes may be included in an icon with a folded corner. The constraint or note icon is then placed next to the relevant model element as shown in Figure 3.1. Although UML provides its own notational language, the Object Constraint Language; the general philosophy is to allow the use of any textual language for constraints or explanatory notes. In this thesis, we use informal descriptions for constraints and the widely understood Backus-Naur Form (BNF) for specific explanations of syntax or terminology.

---

[2] *Composition* has a specific meaning in UML, whereas *composite* is a general term describing something with parts, e.g. an attribute with a complex data type, an object, an association.

3.3 Using UML for Spatiotemporal Data

In this section, we evaluate the core constructs and extension mechanisms defined in UML [Booch99, OMG99, Rumb99] in terms of their suitability for modeling spatiotemporal data and defining a UML extension to facilitate such modeling respectively. The UML usage and notation used is based on Rumbaugh [Rumb99], except that we use informal textual descriptions for complex attribute domains and constraints for the sake of readability. We use Backus-Naur Form (BNF) for specific explanations of syntax or terminology. Section 3.3.1 uses an application example to demonstrate some of the problems associated with modeling spatiotemporal data using only the core model of UML. We evaluate alternative approaches to providing support for spatiotemporal data modeling and explain the rationale for choosing to extend UML in Section 3.3.2. Finally, UML's extension mechanisms are evaluated for their potential utility in defining such an extension in Section 3.3.3.

3.3.1 Using UML: an Example

The following regional health application will be used to illustrate the use of UML to model spatiotemporal data. Assume an application measuring health statistics of different provinces, in terms of average lifespan, as related to the location (i.e. a point in 2D space), number of beds, accessibility (i.e. points in a half-hour travel zone), and surrounding population densities of a province's hospitals. A hospital is classified by category, where a given category is required to have a minimum number of beds in specific kinds of wards. However, category definitions may differ between regions due to local regulations.

For properties dependent on time and/or location, we want to record information about when (using time intervals unless otherwise specified) and/or where a given value is valid (i.e. valid time) or current (i.e. transaction time). For example, a province's population densities and average lifespans can vary and are recorded yearly at the same time instants (values are averaged between yearly measurements) and for the same regions. The number of hospital beds, the hospital's half-hour travel zone, the hospital's category, and the regional definition of hospital categories may change over time as well. We want to record existence and transaction time for hospitals, valid time and transaction time for a hospital's category, and valid time for all of the other time dependent properties. The time unit for the half-hour travel zone is not yet specified, demonstrating incremental design specification. Time elements are used to model hospital existence time since hospitals may sometimes be closed and later re-opened based on changes in local population density. Note that the number of beds, half-hour travel zone, and hospital category are only defined when the hospital is open.

Representation of spatiotemporal concepts using the core constructs of UML is not straightforward, as is illustrated using the regional health example in Figure 3.2. Figure 3.2 uses the following BNF definitions:

spatial-extent[3] := { point | line | region | volume }^{n}

timestamp := { instant | interval | element }

An object with spatial and/or temporal properties (e.g. hospital location or existence time) can be modeled using a separate attribute with a spatial or temporal domain (e.g. *location* or *hospital-existence-time* attributes, respectively). Object or association attributes with spatial and/or temporal properties can be modeled using composite attribute domains consisting of a set of tuples, where each tuple consists of a thematic value, spatial extent, and/or timestamp(s). For example, the half-hour

[3] Note that this definition does not limit the type of spatial extent, which can be discontinuous, have mixed dimensions, have holes or other irregularities, be unbounded, etc.

travel zone and number of hospital beds are modeled by the *halfHourZone* and *numBeds* attributes respectively. This is the most compact representation and is useful whenever the attribute values are likely to be unique across object instances.

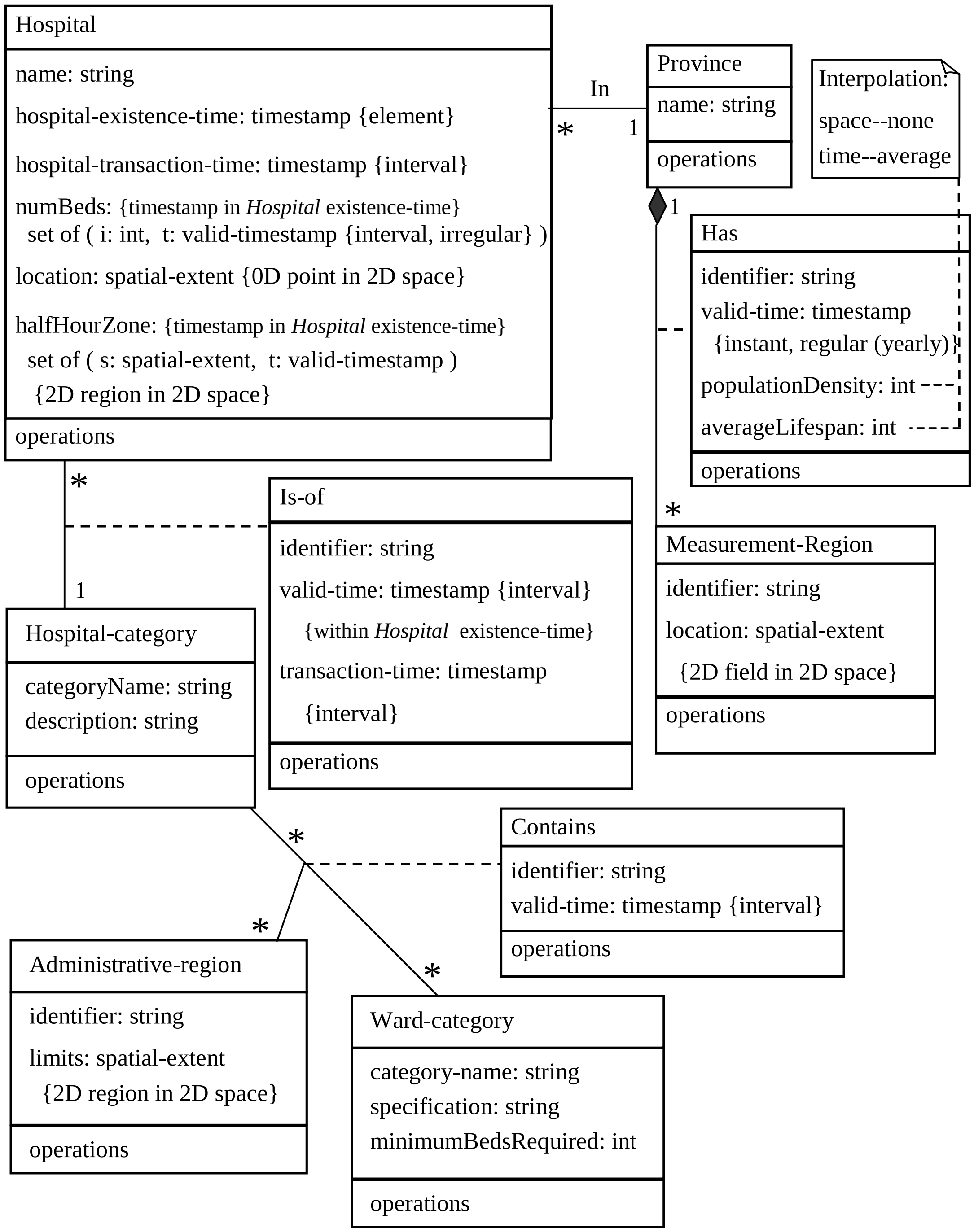


Figure 3.2 Regional Health Application Specification in UML

Alternatively, any attribute with spatial and/or temporal properties (e.g. population density or average lifespan in a province) could be promoted to a separate but associated class with the same information added to the new class. This approach could be useful, for example, whenever attribute domains were restricted to being atomic. Although not required by the semantics of the example application, we must also create an artificial *identifier* attribute for the new class because its instances must be uniquely identified [Rumb99, pp. 304, 307]. Of more concern, both of these approaches (composite attribute domains or a new associated class) could lead to redundancy whenever the same attribute value is repeated for different object instances, times, and/or spatial extents. This is especially significant for spatial data because of their size.

A more correct approach, in general, would be to promote the association to an association class[4] (e.g. *Has*) with spatial data in the *associated* class (e.g. *Measurement-Region*) and thematic and/or timestamp data (e.g. *populationDensity*, *averageLifespan*, and *valid-time*) in the *association* class. This still does not solve the problem of the artificial identifier or the extra complexity introduced for adding classes. However, this approach is preferred when (*i*) the same spatial extent is associated with different object, attribute, or timestamp instances or (*ii*) several different attribute types share the same timestamps or spatial extents. For example, the same measurement region may be associated with different population density or average lifespan values over time. Furthermore, these two thematic attributes are measured at the same time and locations (i.e. share the same timestamps and spatial extents).

Associations (e.g. *Is-of*) with temporal and/or spatial properties can be treated similarly by promoting the association to an association class and adding timestamp and/or spatial attributes to the new association class. In fact, standard UML notation

[4] More complex examples where an association class already exists are discussed in Section 4.4.

requires that any association with its own properties have a linked association class. So even without any considerations of redundancy, each association with spatial and/or temporal properties requires the creation of a new association class.

There could be some cases where it was advantageous to carry the decomposition process into separate object classes even further. Consider the case of an association with both spatial and temporal properties where different association instances are likely to have the same spatial attribute value. For example, the regional definition of hospital categories described previously implies that a single administrative region (e.g. *limits*) is associated with a set of regulations dictating the categories of wards required for each category of hospital during a given time period. Therefore, the value of the *limits* attribute could be the same for different hospital categories, ward-categories, or different timestamps. Creating a completely separate object class for the spatial attribute (i.e. distinct from the association class containing the association's temporal properties, the existing object class for hospital, and the existing object class for each ward category), allows the same spatial extent to be shared rather than duplicated in such cases. The new class would then be added to an existing binary aggregation between *Hospital-category* and *Ward-category*, resulting in the conversion of the binary aggregation to a ternary association. This approach is illustrated by the *Contains* association class and ternary association (between *Hospital-category, Ward-category*, and *Administrative-region*) shown in Figure 3.2. Note that this approach obscures the original part-whole semantics (modeled using aggregation in UML) between ward and hospital categories, which can no longer be represented in the new ternary association since an administrative region is not part of a hospital category.

Constraints are used to indicate the time units for timestamps, the time model, the dimensions of spatial extents, and the existence-dependencies described for the application example. Notes are used to show interpolation semantics. Association, rather than generalization, is used to represent a hospital's category since its definition varies regionally and does not affect the attributes defined for the hospital.

Figure 3.2 shows that modeling objects, attributes, and associations with spatial and/or temporal properties in UML results in the creation of new object and association classes, leading to the creation of a host of artificial constructs that significantly complicate the schema diagram. Furthermore, there is no single, easily visible notation representing spatiotemporal properties. This violates the requirement that the notation be simple, clear, and consistent. A better approach is to extend the fundamental characteristics of the existing UML elements to meet the spatiotemporal requirements. We consider several alternative methods of extending UML and discuss the advantages and disadvantages of each approach.

3.3.2 Alternative Approaches to Extending UML

Three different ways of extending a high-level data model to incorporate additional semantics are discussed in [Greg99]:

- implicit extension by redefining the semantics of existing notation,
- explicit extension by representing the additional semantics using existing constructs, i.e. essentially defining standard patterns in the style of Fowler [Fowl97b] for temporal data and Gordillo [Gord97, Gord98] for spatial data, or
- explicit extension by adding additional constructs to the modeling language.

Another explicit approach would be to define new data types (or ADT) incorporating the semantics that could then be used to describe attribute domains as needed, as in [Erwig99, Guti98].

While the implicit approach may require less initial training for users than the explicit approaches, it results in problems of incompatibility with the original model (i.e. pre-existing schemas now have different semantics) and lack of flexibility (i.e. since the new extended model is no longer suitable for applications not requiring the additional semantic support).

With respect to the explicit approaches, we can see that the pattern-based approach, illustrated in Figure 3.2, has the disadvantage of producing awkward and overloaded schemas; whereas the other two options add to the constructs or data types that must be learned by the user. Essentially, the additional complexity introduced by the new semantics is evident at the level of the schema for the pattern approach and at the level of the modeling language for the other two approaches. This essentially involves a trade-off between ease of initial use versus regular use.

It is our contention that (*i*) the priority should be for facilitating regular use (i.e. production of simple and clear schemas) and that (*ii*) new constructs can be designed to minimize learning time by taking advantage of orthogonality. Furthermore, conversion of new constructs that have equivalents in the original model can be automated for implementation or reference purposes. In the OO context, if the spatiotemporal additional semantics impact the object or association levels, then the definition of new data types for attribute domains is not sufficient: some new constructs will be required. Restricting modeling of spatiotemporal semantics to the attribute level would unnecessarily limit the flexibility of the modeling language;

therefore, we adopt the approach of defining new constructs. Next, we examine the potential use of existing UML extension mechanisms to define such constructs.

3.3.3 Using UML Extension Mechanisms

Stereotypes, tagged values, and constraints are advanced features of UML intended to support extensions to the UML meta-model; therefore, they provide a potential basis for defining a spatiotemporal extension. One problem with these mechanisms, as with some other aspects of UML such as aggregation and composition, is that they are inconsistently described in the main sources for UML [Booch99, OMG99, Rumb99]. A detailed discussion of these and other inconsistencies in UML can be found in [Hend99b].

Stereotypes are used to indicate a variation in usage or meaning for an existing UML model element. Tagged values and constraints can be attached to the stereotype to define its additional properties and semantics respectively. A set of standard stereotypes has been defined for UML [Booch99, pp. 442], but none is defined as applying both to attributes and composite model elements having identity (i.e. classes and associations versus composite attribute domains) or used for both spatial and temporal properties. Fowler [Fowl97a] suggests using a *history* stereotype to model historical associations between classes by adding a temporal subtype to one of the classes. But this seems to imply that a new stereotype should be added for each different level of granularity and does not account for spatial or spatiotemporal attribute variation.

Even if we introduce new stereotypes for spatiotemporal semantics, a strict adherence to the definition of UML extension mechanisms can be problematic. According to Rumbaugh [Rumb99, p. 450], a model element can have at most one stereotype. Instead of defining a model element with multiple stereotypes, a new composite stereotype should be defined using generalization and multiple inheritance, e.g. one for each meaningful combination of spatial, temporal, and thematic data semantics. However, this leads to a proliferation of modeling constructs and a less intuitive representation. Defining a small set of basic constructs that can be combined in a simple and semantically meaningful manner is a much more elegant way to add expressive power to a modeling language without sacrificing understandability or simplicity. Therefore, there are strong arguments for allowing a spatiotemporal extension to violate the strict definition of UML stereotypes by allowing model elements to have more than one stereotype.

Furthermore, Rumbaugh [Rumb99, pp. 449] states that “stereotypes may extend the semantics but not the structure of pre-existing metamodel classes”, with the exception that tagged values can be used to change the structure of a model element (but not its instantiations). Thus, they do not allow specification of types or domains (all tagged values are text strings) and are not intended for “serious semantic extensions to the modeling language itself” [Rumb99, pp. 469]. However, spatiotemporal semantics require a change in the structure of model elements and their instantiations to allow relevant time periods and/or spatial extents to be associated with the model element’s instances or values. Based on this discussion, it is clear that constructs added to extend UML with spatiotemporal semantics will necessarily go beyond the extension mechanisms defined for UML.

3.4 STUML: Fundamental Constructs

The proposed extension to UML, called *SpatioTemporal UML* (*STUML*), is based on the addition of five new symbols, illustrated in Figure 3.3, and a *specification box* describing the detailed semantics of the spatiotemporal data represented using the five symbols. The basic approach is to extend UML by adding a minimal set of constructs for spatial, temporal, and thematic data, represented respectively by *spatial*, *temporal*, and *thematic* symbols. These constructs can then be applied at different levels of the UML class diagram and in different combinations to add spatiotemporal semantics to a UML model element.[5] In addition, the *group* symbol is used to group attributes with common spatiotemporal properties or inter-attribute constraints and the *existence-dependent* symbol is used to describe attributes and associations dependent on object existence.

As discussed previously, although these new symbols can be roughly described as stereotypes; they do not adhere strictly to the UML definition. For improved readability, we use the alternative graphical notation for stereotypes described in [Rumb99, pp. 451]. These symbols can be annotated with a unique label used to reference the associated specification box. The first four symbols can optionally be used without the abbreviations shown in the figure (i.e. *S, T, Th,* and *G* respectively). The specific alphanumeric domain for a thematic attribute can be optionally indicated, e.g. *Th: int* for a thematic attribute having an integer domain.

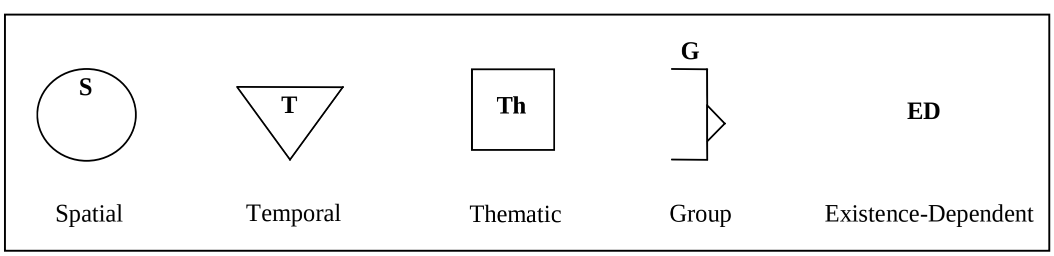


Figure 3.3 STUML Symbols

The *group* symbol, *existence-dependent* symbol and *specification box* are discussed in Section 3.5. The *spatial, temporal,* and *thematic* symbols are described

[5] In this way, expressive power is gained without sacrificing simplicity.

in this section. Section 3.4.1 provides a general overview of the meaning and use of these three symbols. Sections 3.4.2 through 3.4.4 explain the use and associated semantics of these symbols at the attribute (and attribute group), object class, and association levels respectively.

3.4.1 Spatial, Temporal, and Thematic Constructs

These constructs can be used to model spatial extents, object existence or transaction time, and the three different types of spatiotemporal data previously discussed (i.e. *temporal changes in spatial extents*; *changes in the values of thematic data across time or space*; and *composite data whose components vary depending on time or location*). To understand the use and semantics of the *spatial*, *temporal,* and *thematic* constructs, we first discuss the interpretation of each individual symbol separately.

The *spatial* symbol represents a spatial extent, which consists of an arbitrary set of points, lines, regions, or volumes. The spatial extent may be associated with thematic, temporal, or composite data or used to define an attribute domain. For example, Figure 3.4 (a) illustrates the use of the spatial symbol to define an attribute with a spatial domain (i.e. a *spatial attribute*). The *temporal* symbol represents a temporal extent, or timestamp, which may be associated with thematic, spatial, or composite data. For example, Figure 3.4 (b) illustrates the use of the temporal symbol to associate a timestamp with spatial data (i.e. a *temporally dependent spatial attribute*). Timestamps may represent existence time for objects; valid time for associations or attributes; and transaction time for objects, associations, or attributes. Timestamps can also represent valid or transaction time for an object's spatial extent (see Section 3.4.3). The *thematic* symbol represents thematic data.

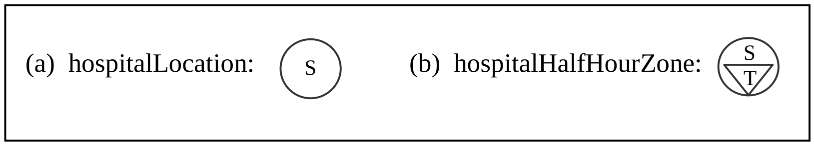


Figure 3.4 Spatial Attribute Examples

The *thematic* symbol can only be used at the attribute level and only in conjunction with one of the other two symbols to describe an attribute with temporal or spatial properties. A thematic attribute domain with no spatial or temporal properties uses standard UML notation, i.e. <*attribute-name*>: <*domain*>. When a thematic domain has additional spatial or temporal properties, either this notation can be used to indicate the specific thematic domain or it can be indicated inside the *thematic* symbol as shown in Figure 3.5.

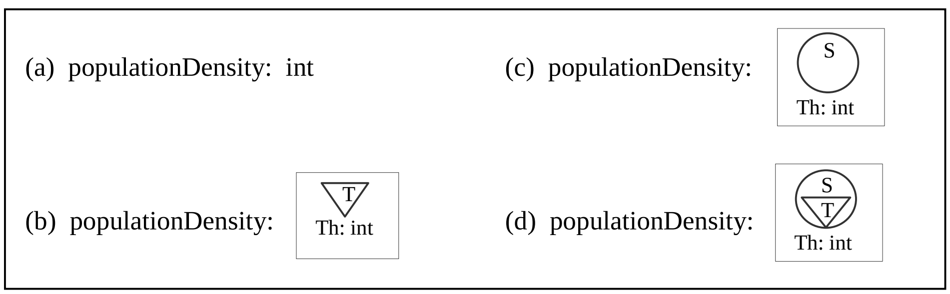


Figure 3.5 Thematic Attribute Examples

Figure 3.5 illustrates by example the four possible cases for a thematic attribute. Table 3.1 lists the terminology used (i.e. semantic attribute category) and the additional spatiotemporal properties for each thematic attribute case. Adjectives are used to describe the attribute domain (e.g. *thematic* attribute) and adverbs with the word *dependent* to describe additional attribute properties for composite attribute domains (e.g. *temporally dependent* thematic attribute). Note that case 3.5 (c) represents the field-based (as opposed to the object-based) view of space; therefore the terms *spatially dependent attribute* and *spatial field* are semantically equivalent.

Table 3.1 Explanation of Thematic Attribute Examples from Figure 3.5

Figure 3.5 example	Semantic attribute category (case)	Spatiotemporal property(s) added to thematic domain
(a)	thematic attribute	none
(b)	temporally dependent thematic attribute	temporal
(c)	spatially dependent thematic attribute	spatial
(d)	spatiotemporally dependent thematic attribute	temporal and spatial

The full expressive power of the three base constructs described above are realized by applying them in combination and at different levels of the object model. Therefore, the semantics of STUML depend on three factors: (*i*) the symbol used, (*ii*) the model element described by the symbol (i.e. object, association, or attribute), and (*iii*) whether the symbol is combined with other symbols. The general rules for combining symbols can be summarized as follows.

- *Nesting one symbol inside another* represents mathematically a function from the domain represented by the inner symbol to the domain represented by the outer symbol. Therefore, different orders of nesting symbols correspond to different functional expressions and represent different perspectives of the data.

  For example, Figure 3.5 (b) represents a function from the time to the integer domain for a given object or association instance. If we reverse the order of the symbol nesting, this would represent the inverse function from the integer to the time domain. However, from the conceptual design and schema perspective, both represent the same semantic modeling category and would result in the same conceptual and logical schema, i.e. a temporally dependent, thematic attribute.

Rather than arbitrarily restricting the representation of a semantic modeling category to one order of nesting, we prefer to allow the users to select the order of nesting that best matches their perspective of the application data. Although not explored in this thesis, the different orders of nesting could be exploited for a graphical query language or to indicate preferred clustering patterns to the database management system in generating the physical schema.

Note also that in Figure 3.5 (b), only one integer value is associated with each timestamp; however, several different timestamps may be associated with the same integer value. In Figure 3.5 (d), several integer values will be associated with each timestamp, one for each spatial location.

- *Placing one symbol next to another* symbol represents mathematically two separate functions, one for each symbol. The order in which the two symbols are written is not significant.

We now give the rule for which *symbolic combinations* are legal at each model level (i.e. attribute, object, and association), the *semantic modeling constructs* defined at each level with a textual description of each construct, and a *mapping* between the two. For a given semantic modeling construct, the textual and mathematical definitions are given for each possible symbol nesting that represents that construct.

Note that any reference to a timestamp, timestamps, a time point, or time validity in the definitions for a given symbol nesting could be for any time dimension. That includes transaction time and/or either valid time for attributes, associations, or an object's spatial extent (see Section 3.4.3) or existence time for objects. The first symbol nesting given for each semantic modeling construct is used in the examples.

We first summarize the *primitives* used in this chapter to denote various time, space, and model elements. Note the distinction between a domain and a power set (i.e. the set of all subsets) in that domain. The former is used to represent individual time instants (<T>) or points in space (<S>) and the latter is used to represent one or more timestamps (<2^T>) or spatial extents (<2^S>). Note also that an association identifier consists of the set of object identifiers for the objects participating in that association.

<T> ::=	domain of time instants
<2^T> ::=	domain of sets of time instants, where each set of time instants can be considered a timestamp or a set of timestamps
<S> ::=	domain of points in space
<2^S> ::=	domain of sets of points in space, where each set of points in space can be considered a spatial extent or a set of spatial extents
<oid> ::=	domain of object-identifiers
<aid> ::=	domain of association-instance identifiers, i.e. { <oid> }^{n}, where n>1
<id> ::=	domain of object and association identifiers, i.e. { <oid> | <aid> }
<D> ::=	thematic, i.e. alphanumeric, domain (e.g. integer, string)
<d> ::=	thematic attribute symbol
<t> ::=	temporal symbol
<s> ::=	spatial symbol
<s&t> ::=	any nested combination of a spatial and temporal symbol
<s&d> ::=	any nested combination of a spatial and thematic symbol
<t&d> ::=	any nested combination of a temporal and thematic symbol
<s&t&d> ::=	any nested combination of a spatial, temporal, and thematic symbol
<ED> ::=	existence-dependent symbol

3.4.2 The Attribute Level

At the attribute level, we can model *temporal changes in spatial extents*, where the spatial extent represents a property of an object (i.e. spatial attribute), and *changes in* *the value of thematic data across time and/or space* (i.e. spatially and/or temporally dependent thematic attributes or attribute groups[6]).

Legal combinations of symbols at the attribute level are any nested combination of a spatial symbol, a temporal symbol, and/or a thematic symbol. The only exception is that the temporal symbol cannot be used alone. An attribute with a temporal domain is treated as thematic data since temporal data types are pre-defined for popular standard query languages such as SQL (see Section 2.2.3). The attribute domain can optionally be followed by an *existence-dependent* symbol (discussed in Section 3.5). The rule for notation at this level can be defined using BNF notation and the primitives defined previously:

attribName: [<D> | <s&d> | <t&d> | <s&t&d> | <s> | <s&t>] [<ED>]

Six different attribute domains are possible, corresponding to six different semantic categories of attributes (i.e. modeling constructs). Reading the attribute domain symbols left to right in the above rule, we have:

- <D>: *thematic attributes*, illustrated in Figure 3.5 (a),
- <s&d>: *spatially dependent thematic attributes,* illustrated in Figure 3.5 (c),
- <t&d>: *temporally dependent thematic attributes,* illustrated in Figure 3.5(b),
- <s&t&d>: *spatiotemporally dependent thematic attributes,* illustrated in Figure 3.5 (d)
- <s>: *spatial attributes,* illustrated in Figure 3.4 (a), and
- <s&t>: *temporally dependent spatial attributes*, illustrated in Figure 3.4 (b).

Except for thematic attributes, these domains represent extensions for spatiotemporal data modeling.

[6] Attribute groups, including their spatiotemporal semantics and syntax, are described in Section 3.5.

For each of the semantic attribute categories listed above, a general textual description and all of the symbolic representations (i.e. symbol nestings) possible for that category are given in graphic form. For each symbolic representation possible, a mathematical definition and textual definition for that representation is given. For example, a spatiotemporally dependent thematic attribute involves a nested combination of three symbols (spatial, temporal, and thematic symbol represented by a circle, triangle, and square respectively). Therefore, there are six possible legal symbol nestings possible, each representing a different perspective of the same semantic attribute category.

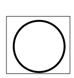
Note that *each one of the descriptions and definitions below apply to the identified object or association instance*; therefore, we do not state this explicitly for each case.

- *Thematic Attribute:* This is an attribute having thematic value(s) for the identified object or association (the phrase *for the identified object or association* is assumed for the other semantic attribute categories).

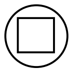
  <D>   f: <id>   <D>

  Returns the thematic attribute value for the identified object or association.

- *Spatially dependent Thematic Attribute:* This is a set of thematic attribute values, each associated with a spatial extent representing the location where that attribute value is valid. This implies that the attribute values may change over space and their changed values may be retained.

     f: <id>   (<S>   <D>)

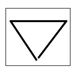
  Returns a set of spatial points, each with its associated thematic attribute value (valid for that spatial point).



f: <id> (<D> <2^S>)

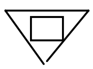
Returns a set of thematic attribute values, each with its associated spatial extent (where that thematic attribute value is valid).

- *Temporally dependent Thematic Attribute:* This is a set of thematic attribute values, each associated with one or more timestamps, representing the attribute value's valid and/or transaction time for a given object or association identifier. This implies that the attribute values may change over time and their changed values may be retained.



f: <id> (<T> <D>)

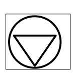
Returns a set of time points, each with its associated thematic attribute value (i.e. valid or current for that time point).



f: <id> (<D> <2^T>)

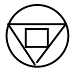
Returns a set of thematic attribute values, each with its associated timestamps (i.e. when that thematic attribute value is valid or current).

- *Spatiotemporally dependent Thematic Attribute:* This is a combination of spatially and temporally dependent thematic attributes as defined above, i.e. a set of thematic attribute values, each associated with a spatial extent and one or more timestamps.



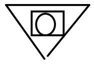
f: <id> (<T> (<S> <D>))

Returns a set of time points, each with its associated set of spatial points, and, for each spatial point, its associated thematic attribute value (i.e. valid or current for that time and spatial point).



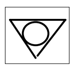
f: <id> (<D> (<T> <2^S>))

Returns a set of thematic attribute values, each with its associated set of time points, and, for each time point, its associated spatial extents (i.e. where that thematic value is valid or current for that time point).



f: <id> (<S> (<D> <2^T>))

Returns a set of spatial points, each with its associated set of thematic attribute values, and, for each thematic attribute value, its associated timestamps (i.e. when that thematic attribute value is valid or current for that spatial point).

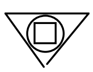


f: <id> (<S> (<T> <D>))

Returns a set of spatial points, each with its associated set of time points, and, for each time point, its associated thematic attribute value (i.e. valid or current for that spatial and time point).

f: <id> (<T> (<D> <2^S>))

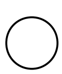
Returns a set of time points, each with its associated set of thematic attribute values, and, for each thematic attribute value, its associated spatial extents (i.e. where that thematic attribute value is valid or current for that time point).



f: <id> (<D> (<S> <2^T>))

Returns a set of thematic attribute values, each with its associated set of spatial points, and, for each spatial point, its associated timestamps (i.e. when that thematic attribute value is valid or current for that spatial point).

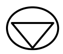
- *Spatial Attribute:* This is an attribute with a spatial domain, i.e. the attribute value is a spatial extent.



f: <id> <2^S>

Returns the spatial attribute value.

- *Temporally dependent Spatial Attribute:* A spatial attribute is associated with one or more timestamps, representing the spatial extent's valid and/or transaction time.

   f: <id> (<T> <2^S>)

  Returns a set of time points, each with its associated spatial attribute value (i.e. spatial extent).

   f: <id> (<S> <2^T>)

  Returns a set of spatial points, each with its associated timestamps (i.e. when the spatial attribute value, i.e. spatial extent, intersects that spatial point).

The use of these symbols at the attribute level is illustrated in Figure 3.6 based on extracts from the regional health application described in Section 3.3.1, with the addition of the *rank* attribute to indicate the performance ranking of a hospital during a specific time period.

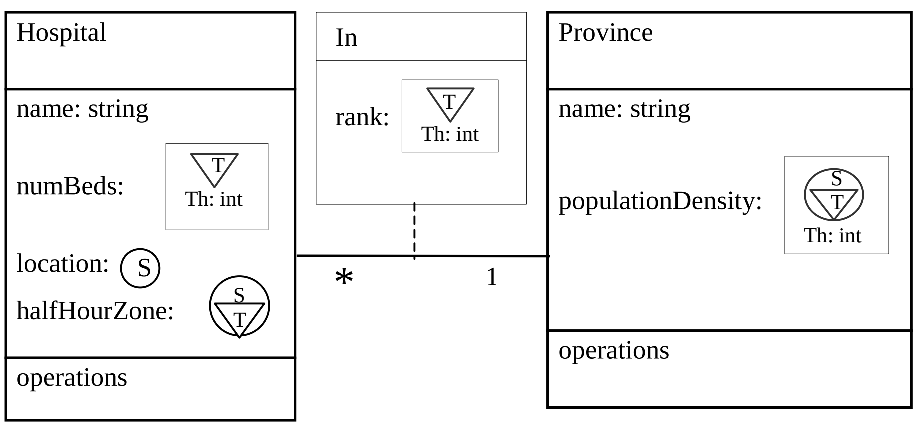


Figure 3.6 Using STUML at the Attribute Level

The correspondence between a specific semantic attribute category and its attribute example in Figure 3.6 is shown in Table 3.2. All of the examples are object attributes except the *rank* attribute from the *In* association, which illustrates the use of STUML constructs for association attributes.

Table 3.2 Examples of Semantic Attribute Categories in Figure 3.6

Semantic Attribute Category	Attribute Example in Figure 3.6
thematic attribute	*name* (for *Hospital* and *Province*),
temporally dependent thematic attribute	*numBeds*, *rank* (association attribute)
spatiotemporally dependent thematic attribute	*populationDensity*
spatial attribute	*location*
temporally dependent spatial attribute	*halfHourZone*

A thematic attribute domain is indicated as a string after the attribute (as in the *name* attribute) or—if that attribute also has temporal or spatial properties (as with the *numBeds*, *rank*, and *populationDensity* attributes)—by the use of a thematic symbol (a square). If no thematic domain is explicitly specified for an attribute, then the use of the spatial symbol (a circle) indicates that the attribute has a spatial domain. Thus, the *Hospital location* and *halfHourZone* attributes represent spatial data. The nested temporal symbol (a triangle) used for *halfHourZone* indicates that the spatial extent associated with this attribute may change over time and thus should be timestamped. Therefore, an attribute marked by a spatiotemporal symbol (and no thematic symbol) represents a spatial extent that changes over time. In this case, as transport networks change, the geometry of the half-hour travel zone must be updated.

In contrast, an attribute that has a thematic domain with spatial and/or temporal properties represents a spatially and/or temporally dependent thematic attribute. This is indicated graphically by using the thematic symbol with the spatial and/or temporal symbols. Thus the thematic symbol is used to differentiate between two different types of spatiotemporal data: *temporal changes in spatial extents* (without the thematic symbol) and *changes in the value of thematic data across time and* *space* (with the thematic symbol). The fact that *numBeds* has an integer domain and thematic symbol associated with a temporal symbol indicates that the integer value of *numBeds* may change over time and should be timestamped. Analogously, the integer value of *populationDensity* may change over time or space and thus each value is associated with a timestamp and spatial extent.

Finally, the thematic integer attribute *rank* (added to the original *many*:*one* association *In* between Hospital and Province from Figure 3.2) illustrates the use of STUML symbols with association attributes. The attribute *rank* indicates the performance ranking of a specific hospital in a province during a specific time period. In order to add attributes to the association *In*, the association must first be raised to an association class, following standard UML notation.

3.4.3 The Object Class Level

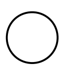
At the object class level, we can model *temporal changes in spatial extents*, where the spatial extent is associated with an object instance. We can also model the time an object exists in the real world (i.e. existence time) or is part of the current database state (i.e. transaction time).

An object class can be marked by a temporal symbol, a spatial symbol, or any nested combination of these. In addition, this is the only level where the symbols can be paired; i.e. a temporal symbol can be paired with either a spatial symbol or a nested combination of the two symbols. The separate temporal symbol represents the existence and/or transaction time of the object. The spatial symbol represents the spatial extent associated with that object. If the spatial symbol is combined with a nested temporal symbol, then the spatial extent is timestamped to show the valid or transaction time of the spatial extent. Since the object can exist or be current even when it is not associated with a spatial extent, separate timestamps are required for the object instance and for the object instance's spatial extent. The rule for object level notation can be given using BNF notation and the primitives defined previously as follows:

className [<s> | <s&t>] [<t>]

Corresponding to the five possible instantiations of the rule, <s>, <s&t>, <t>, <s><t>, and <s&t><t>; there are five different semantic categories of object classes. They are *spatial object*, *temporally dependent spatial object*, *temporal object*, *spatiotemporal object*, and *temporally dependent spatiotemporal object* respectively. A general textual description, symbolic representations, mathematical definitions, and textual definitions are given below for each of these semantic categories.

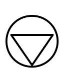
- *Spatial Object (Class):* An object is associated with a spatial extent. This is equivalent to an object having a single spatial attribute except that there is no separate identifier for the spatial extent.

  f: <oid>   <2^S>

  Returns the spatial extent of the identified object.

- *Temporally dependent Spatial Object (Class):* The spatial extent associated with a spatial object is also associated with one or more timestamps, representing the spatial extent's valid and/or transaction time.

  f: <oid>   (<T>   <2^S>)

  Returns a set of time points, each associated with the spatial extent of the identified object at that time point.

  f: <oid>   (<S>   <2^T>)

  Returns a set of spatial points, each with its associated timestamps (i.e. when the object's spatial extent intersects that spatial point), for the identified object.

- *Temporal Object (Class):* An object is associated with one or more timestamps, representing the object’s existence and/or transaction time.

   f: <oid> <2^T>

  Returns the timestamp of the identified object.

- *Spatiotemporal Object (Class):* This is a combination of a spatial and temporal object as defined above, i.e. each object instance is associated with a spatial extent and one or more timestamps representing the object’s existence and/or transaction time.

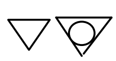
   f: <oid> <2^T> and f: <oid> <2^S>

  Returns the timestamp and the spatial extent of the identified object.

- *Temporally dependent Spatiotemporal Object (Class):* This is a combination of a temporally dependent spatial object and a temporal object as defined above, i.e. an object is associated with a spatial extent, one or more timestamps representing the spatial extent’s valid and/or transaction time, and one or more timestamps representing the object’s existence and/or transaction time.

   f: <oid> <2^T> and f: <oid> (<T> <2^S>)

  Returns the timestamp of the identified object and a set of time points, each with its associated spatial extent (i.e. valid at that time point), for the identified object.

   f: <oid> <2^T> and f: <oid> (<S> <2^T>)

  Returns the timestamp of the identified object and a set of spatial points, each with its associated timestamps (i.e. when the object’s spatial extent intersects that spatial point), for the identified object.

The use of symbols at the object class level is illustrated in Figure 3.7. In Figure 3.7 (a), the temporal symbol at the *Hospital* object level represents a temporal object class with existence and transaction time (see Section 3.5). In Figure 3.7 (b), we give an example of a temporally dependent spatial object. This example assumes that there is no need to represent hospital location separately from the half-hour travel zone. Instead, a hospital object is treated as a spatial object with a single associated spatial extent, showing the half-hour travel zone around that hospital. The temporal symbol indicates that the spatial extent should be timestamped, since the half-hour travel zone can change over time. Finally, Figure 3.7 (c) combines 3.7 (a) and 3.7 (b), illustrating a temporally dependent spatiotemporal object. The object is *spatiotemporal* because it has timestamp(s) and a spatial extent; and it is *temporally dependent* because the spatial extent also has timestamp(s). In this case, the value of the spatial extent is dependent on its timestamp(s) but not on the value of the object's timestamp(s) (see Section 3.5 for further explanation).

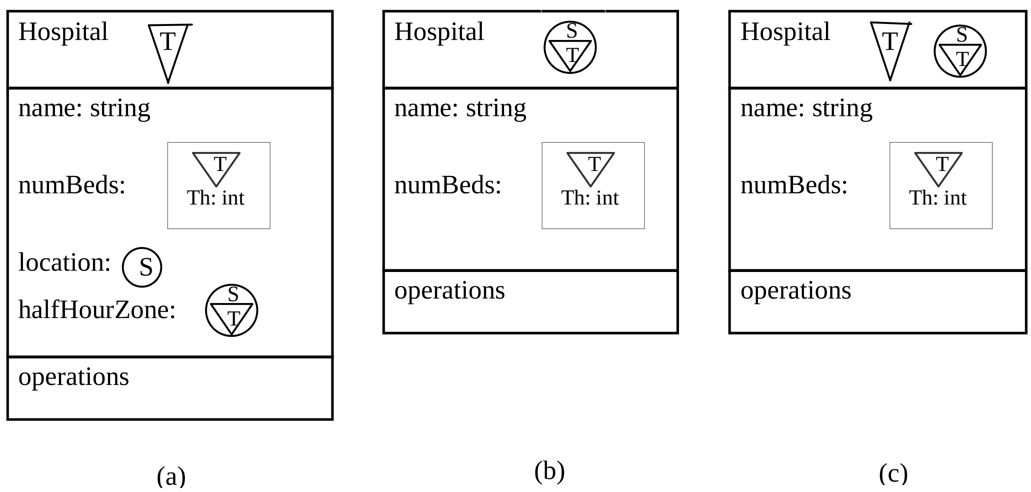


Figure 3.7 Using STUML at the Object Class Level

3.4.4 The Association Level

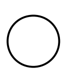
At the association level, we can model *temporal changes in spatial extents*, where the spatial extent is associated with a relationship between object instances (i.e. spatiotemporal association), and *composite data whose components vary depending on time or location* (i.e. spatiotemporal aggregation or composition). The following discussion applies to any type of association, including aggregation and composition.

At the association level, any nested combination of a spatial and/or a temporal symbol represents a legal combination describing spatiotemporal properties of the association. Except for the omission of the *thematic* symbol, the association level is similar to the attribute level. The association spatiotemporal properties can optionally be followed by an *existence-dependent* symbol (discussed in Section 3.5). The rule for the association level notation can be given in BNF as follows:

assoc-line [<s> | <t> | <s&t>] [<ED>]

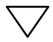
Reading the rule from left to right, the three possible spatiotemporal symbol combinations <s>, <t>, or <s&t> correspond to the three different semantic categories of associations. These are *spatially dependent associations*, *temporally dependent associations*, and *spatiotemporally dependent associations* respectively. A general textual description, symbolic representations, mathematical definitions, and textual definitions are given below for each of these semantic categories.

- *Spatially dependent Association:* An association instance is associated with a spatial extent representing the location where the association instance is valid. This implies that the association instances may change over space and their changed instances may be retained.

   f: <aid> <2^S >

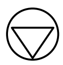
  Returns the spatial extent of the identified association.

- *Temporally dependent Association:* An association instance is associated with one or more timestamps, representing the association's valid and/or transaction time. This implies that association instances may change over time and the changed instances may be retained.

   f: <aid> $<2^T>$

Returns the timestamp(s) of the identified association.

- *Spatiotemporally dependent Association:* This is a combination of spatially and temporally dependent associations as defined above, i.e. an association is associated with a spatial extent and one or more timestamps.

   f: <aid>   (<T> $<2^S>$)

Returns a set of time points, each with the associated spatial extent for the identified association at that time point.

   f: <aid>   (<S> $<2^T>$)

Returns a set of spatial points, each with its associated timestamp(s) (i.e. when the association instance's spatial extent intersects that spatial point), for the identified association.

The use of these symbols at the association level is shown in Figure 3.8.

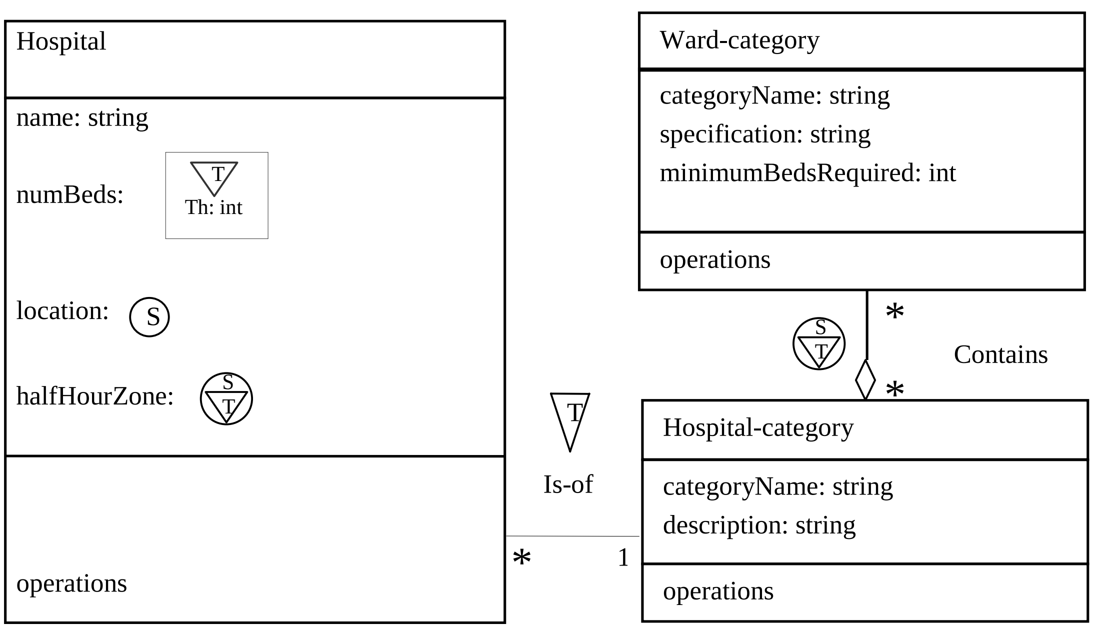


Figure 3.8 Using STUML at the Association Level

Marking the *Is-of* association with a temporal symbol signifies that the category of a hospital may change over time, as local health needs change and wards are opened or closed. Therefore, association instances should be timestamped.

A spatially dependent association is one where an association instance is associated with a spatial extent to show where that instance is valid. For example, the same category of hospital may require different categories of wards in different areas depending on local regulations. Therefore, the *Contains* aggregation association must be spatially dependent. In fact, since the local categories may also change over time, the *Contains* aggregation association is actually spatiotemporally dependent. In this case, both of the associated object classes are purely conceptual. An association between two physical object classes can also be spatiotemporally dependent; e.g. a consultation of a ward doctor with a specialist is scheduled for a specific location and period of time in the hospital.

It is important to consider whether any constraints should be implicitly assumed between the timestamps or spatial extents of participating objects with those of a temporally and/or spatially dependent association, especially in the case of aggregation and composition. The specific constraints that are appropriate depend on the semantics of the particular association. As an illustration, consider a grandparent/grandchild association between two people. Such an association can be defined even outside the existence time of the grandparent. Similarly, the *aggregation* of particular types of car components (i.e. *component class*) in the design of a given car model (i.e. *model class*) may vary regionally. For example, special anti-polluting devices may need to be added to the car model in certain states. However, there are no implied constraints between the spatial extents of the classes and the aggregation.

These examples show that such constraints are application dependent; therefore, they should not be incorporated as implicit defaults in the modeling language but should be specified explicitly. This can be done either on an ad hoc basis as required using UML constraints or by defining explicit modeling constructs for commonly used constraint patterns. The latter approach is illustrated by the introduction of the existence-dependent construct in Section 3.5 to support the semantics of temporal dependency between associations and their participating objects or between objects or associations and their attributes. Explicit modeling constructs for spatial object associations involving implied spatial constraints (called *spatial Part-Whole relationships)* are described in Chapter 5 and for binary and n-ary topological constraints in Chapter 6.

3.5 STUML: Specification Box, Existence Time, Groups

Section 3.4 described the different types of timestamps that can be associated with an attribute, association, or object class: but where do we specify which types are required for a given application? Detailed spatiotemporal semantics are specified in a *specification box,* which can be associated with any of the icons or combinations using a unique naming convention (used in this thesis) or label. A specification box was adopted instead of standard UML mechanisms such as tagged values or constraints for the reasons discussed in Section 3.3.3. Specification boxes can be inherited from parent classes as with any other class property. The *specification box* syntax is illustrated in Figure 3.9.

Although the exact specifications required may vary between individual applications, it is possible to suggest general guidelines for the common types of information required for spatiotemporal applications based on the issues of time and space models and representation discussed in Chapter 2. In this context, the specification box construct is intended to serve as a general guide and standardized communications medium for spatiotemporal application developers. The rest of this section examines in detail the different types of information contained in the specification box and two other modeling constructs, the *existence-dependent* and *group* construct, associated with that information.

```
SPECIFICATION BOX  <Identifier>:
TimeDimen. ::=  [existence | valid ] [ transaction ]
TimeInterpolation ::= discrete | step | min | max | avg | linear | spline | <user-defined>
TimeModel ::= irregular | ( regular {<frequency> [ ,<beginning>,<end> ]} )
TimeUnit [ (<TimeDimen.>) ] ::= instant | interval | element
SpaceInterpolation ::= <same as TimeInterpolation>
SpaceModel ::= '('<max object/field dim>,<max search space dim>')': object | field
Group ::= independent | (dependent  (formula )* )
```

Figure 3.9 Specification Box Syntax in STUML

The specification box includes information on the time units and the time and space dimensions, models, and interpolation. Users can specify regular (recorded at regular intervals) or irregular time models and object-based or field-based space models. Interpolation functions can be specified to derive values between recorded spatial locations or timestamps for spatially and/or temporally dependent thematic attributes. The time units (i.e. instant, interval, element) used are defined in [Jens98].

Time dimensions include existence time (for objects), valid time (for attributes, associations, and a spatial object's spatial extent), and transaction time (for objects, attributes, associations, or a spatial object's spatial extent), as defined in [Jens98]. However, object existence time is more precisely defined as the time during which existence-dependent attributes and associations can be defined (i.e. have legal values) for that object. In other words, existence-dependent attributes and associations are those that are defined only when the related object(s) exist. This implies that attributes and associations that are not existence-dependent (e.g. an employee's social-security number) may be defined even when the related object(s) no longer exist. Other attributes, e.g. work-phone number, are defined only while the related object(s) exist (e.g. the employee works at the company) and are therefore existence-dependent.

Note that existence time is not necessarily equivalent to the biological or physical lifespan of an object. For example, existence time may be used to model the time that a given hospital is open; whereas the hospital may be associated with grounds and physical facilities even when it is not open. For example, this could be true before the initial opening, during a period of temporary closure, or after permanent closure but before being demolished. Thus the existence time of the hospital is not equivalent to the physical lifespan of the hospital. The exact meaning of the term *existence time* will be dependent on the application; therefore, individual applications define which attributes and associations are existence-dependent. Object identifiers are never existence-dependent, as they can be used to refer to historical objects. Any other attribute or association can be defined as being existence-dependent.

If existence time is associated with a given object, the existence-dependent attributes and associations for that object class must be explicitly marked as such by adding the superscript *ED* (the *existence-dependent* construct) to the attribute or association name. Conversely, existence-dependent attributes and associations can only be defined for objects having existence time specified. In the case of an existence-dependent association, existence time must be defined for at least one of the participating objects.

If an existence-dependent attribute of an object is temporally dependent, then every valid-time timestamp for the attribute's instance data must be included within the existence time of the corresponding object instance. If an existence-dependent association (or an existence-dependent attribute of an association) is temporally dependent, then every valid-time timestamp for the association's instance data (or for the attribute's instance data) must be included in the intersection of the existence times for those participating object instances that have existence time defined. An existence-dependent object attribute is undefined outside the existence time of the corresponding object instance. Similarly, an existence-dependent association (or association attribute) is undefined outside the intersection of the existence times for those participating object instances that have existence time defined.

In the case of a temporally dependent spatiotemporal object having existence time for the object and valid time for its associated spatial extent, it is assumed that the spatial extent is *not* existence dependent on the object. Otherwise, the object's associated spatial extent must be modeled as a separate existence-dependent attribute of the object. Furthermore, the time specifications for the object's spatial extent are given separately from those of the object; although both are located in the same object specification box. The time specifications for the object itself (i.e. existence time and/or transaction time) precede the space specifications that describe the object's spatial extent. The time specifications for the object's spatial extent (i.e. valid time and/or transaction time) are listed after the space specifications.

Note that the time model and interpolation specification apply only to valid time, whereas the time unit specification is used both for valid or existence time and transaction time. Therefore, the dimension must be specified for time unit whenever a model element is associated with both valid or existence time and transaction time. In addition, time interpolation is normally used for temporally dependent thematic attributes. Time interpolation of spatial attributes (i.e. spatial extents) must be discrete (i.e. no interpolation) or user defined. If not specified, the default assumption is an irregular time model and discrete interpolation (i.e. no interpolation), i.e. the first option listed for each of these categories in the specification box syntax from Figure 3.9.

Space dimensions include the dimensions of the spatial extent(s) being specified, followed by the dimensions of the underlying search space. The object-based spatial model is used for a spatial object or spatial attribute. This indicates that a single spatial extent is recorded for a given object instance or its attribute instance respectively. The field-based spatial model is used for a spatially dependent thematic attribute; where a single object instance has a set of thematic values for that attribute, each associated with a different spatial extent. Space interpolation applies only to spatially dependent thematic attributes using the field-based spatial model. As with time, the default assumption is discrete (i.e. no) space interpolation if not otherwise specified.

The specification box can also be used to specify spatiotemporal constraints within an attribute group. Otherwise, the default assumption is that the attributes in the group are independent. The *group* construct and symbol is used to group attributes sharing the same timestamps or spatial extents (i.e. measured at the same times and locations), which then only need to be specified once for the group. Thus, the group symbol graphically illustrates associated sets of attributes and avoids the possibility of redundantly specifying the same spatial extents and timestamps. Note that a group's attributes never share thematic values, even if the thematic symbol is used in the group specification. If the group's attributes have different thematic domains, then these can be indicated next to each attribute using standard UML text notation. Otherwise, the spatiotemporal semantics and syntax of attribute groups is identical to that of spatially and/or temporally dependent thematic attributes (described in Section 3.4.2).

Finally, any additional spatiotemporal constraints beyond those explicitly supported by the specification box syntax can be specified for the associated construct. For example, n-ary topological constraints on an association between a set of spatial objects (called a *spatial Part-Whole relationship* and described in Chapter 5) can be specified in the specification box for that association. This is described in Section 6.6.2, using the explicit modeling constructs proposed in Section 6.5 for specification of n-ary topological constraints.

3.6 Using STUML: the Regional Health Care Example

Figure 3.10 shows the full regional health application described in Section 3.3.1 as it would be represented using the proposed UML extension, STUML. The use of spatial, temporal, and thematic STUML symbols was illustrated in Section 3.4. Figure 3.10 shows how the specification box, group, and existence-dependent constructs are used with these symbols to specify detailed spatiotemporal semantics.

Following UML convention, another compartment, called the *specification* compartment, is added to each object class to accommodate the *specification boxes* for that class. The *specification* compartment can be used to specify spatiotemporal semantics for the object, the attributes of the object class, and any associations in which the object class participates. Alternatively, a *specification* compartment can be added to an association class to specify spatiotemporal semantics for that association and its attributes.

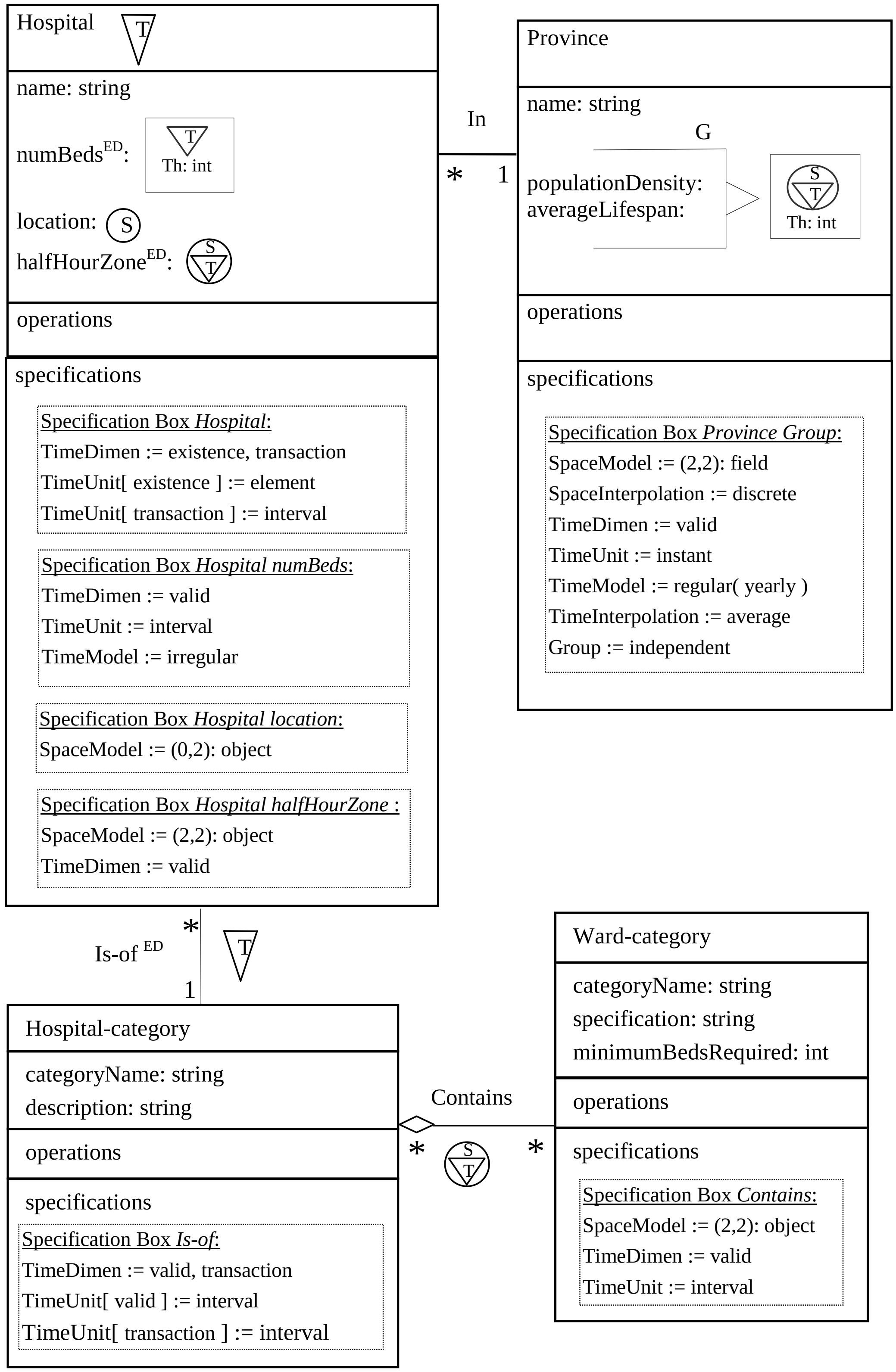


Figure 3.10 Regional Health Application Specification in STUML

In Figure 3.10, each of the object classes has a specification compartment. Four specification boxes are added to the specification compartment of the *Hospital* object class to describe the spatiotemporal semantics of the *Hospital* object, *numBeds* attribute, *location* attribute, and *halfHourZone* attribute respectively. No specification box is needed for the *name* attribute since it is a standard thematic attribute without spatiotemporal properties.

Specification boxes for those associations with spatiotemporal semantics, *Is-of* and *Contains*, are located in the specification compartments of their participating object classes, *Hospital-category* and *Ward-category* respectively. Note that the specification box for an association can be placed in the specification compartment of any participating object class. Therefore, the specification box for *Is-of* could have been located in the *Hospital* instead of the *Hospital-category* object class. Similarly, the specification box for *Contains* could have been located in the *Hospital-category* instead of the *Ward-category* object class.

Finally, the specification box giving spatiotemporal details for the attribute group in the *Province* object class is located in the class's specification compartment. As with *Hospital name*, no specification box is needed for *Province's name* attribute since it does not have any spatiotemporal properties. We then proceed to look at the spatiotemporal semantics specified for individual attributes, attribute groups, objects, and associations in more detail to illustrate the expressive power of STUML.

Hospital location is specified as a single point in 2D space. *Hospital halfHourZone* and *Contains* are specified as a region in 2D space. In contrast, the *Province populationDensity* and *averageLifespan* group is associated with a 2D field in 2D space. This means that, for a single object instance, the two attributes in the group are associated with a set of regions and have a separate attribute value for each region for a given point in time. Since these two attributes share common timestamps and spatial extents, they are grouped. Since both attributes are integers, we can specify the thematic domain in the group symbol. If the attributes had different thematic domains, then we would specify them for each attribute rather than for the group.

The group is then associated with a single symbol and *specification box*. Here we specify that any attribute in the group uses average interpolation in time and no interpolation in space, has a valid time dimension using *instant* as the time unit, and is measured yearly (i.e. a new set of values is recorded for the attribute each year). This means that the population density and average lifespan between recorded time instants is assumed to be the average of the values at the two nearest time instants and undefined outside of recorded spatial regions. No inter-attribute constraints are defined for the group, as shown by the keyword *independent*.

The temporal symbol at the *Hospital* object level is used to indicate existence time and transaction time. Existence time is used to model the periods when the hospital is open, i.e. when the existence-dependent attributes *numBeds* and *halfHourZone* and the existence-dependent association *Is-of* are defined. Since these model elements are temporally dependent, the valid timestamps of all their instances must be included within the *Hospital* existence time. As explained in Section 3.3.1, a time element is used to model hospital existence time since a hospital can close temporarily and re-open later.

Attribute *numBeds* is specified as irregular because this attribute is not recorded periodically: whenever it changes the new value is recorded. As explained in Section 3.5, this is the default assumption for any model element with temporal properties unless specified otherwise. Similarly, discrete time and space interpolation are assumed except where specified otherwise (e.g. as with the attribute group in Figure 3.10). In contrast, there are no default assumptions for time dimensions, time units, or space model. When these specifications are missing, as with the time units for the *halfHourZone* hospital attribute, this indicates that those specifications have not yet been decided. This is an example of incremental design.

As previously noted, the specification box for an association (e.g. *Is-of*) can be placed in the specification compartment of either of its participating object classes (e.g. *Hospital* or *Hospital-category*). Note that since *Hospital-category* is not temporal and therefore does not have existence time defined, the only constraint on the valid-time timestamps of the *Is-of* association comes from the *Hospital class* existence time. Therefore, it may be preferable to include the *Is-of* specification box in the *Hospital* class. Analogously, a convention of placing the specification box for an aggregation or composition association (e.g. *Contains*) in the *whole* object class (e.g. *Hospital-category*) would be more consistent with the rules for spatial Part-Whole relationships described in Chapter 5.

Comparing the schemas of the regional health application from Figure 3.10 and Figure 3.2, the schema that uses STUML specifications is much simpler than the corresponding UML schema. As discussed in Section 3.3.1, the UML schema required the creation of additional classes for each association or attribute group with spatial and/or temporal properties. If attribute domains were further restricted to being atomic rather than composite, then even more classes would have to be created for each thematic attribute with spatial and/or temporal properties and for each spatial attribute with temporal properties. Therefore, the use of UML in Figure 3.2 results in the creation of a host of artificial constructs to represent spatiotemporal semantics, obscuring the schema design. We can see that far fewer object classes are required in Figure 3.10 to describe the same application example. Note also that the part-whole semantics between hospital and ward categories can be modeled (using aggregation) in the STUML schema but cannot be represented in the equivalent UML schema for the reasons given in Section 3.3.1.

Modeling representative excerpts of actual spatiotemporal applications, e.g. a cadastral application [Tryf99] and medical multimedia application [Dion98], showed a similar pattern. In particular, fewer object classes were required to model spatiotemporal associations or attribute groups. Fewer attributes were required, since graphical symbols and specification boxes were used instead of extra attributes (e.g. for time dimensions or identification) to provide a compact, distinct, and consistent representation of spatiotemporal properties.

By incorporating spatiotemporal semantics in the modeling language itself, STUML reduces the complexity of the resulting schemas. The level of detail is reduced without sacrificing understandability. This allows the application developer to concentrate on the characteristics of the specific application domain of interest. The modular specification of spatiotemporal properties also facilitates schema reuse and extension.

For example, if we want to reuse the schema from Figure 3.10 for the same application but without historical records; we can simply delete all of the temporal symbols and specifications. Similarly, if hospital definitions do not vary regionally, one need only remove the spatial symbol from the *Contains* icon and specification box. In contrast, the modifications required to reuse the schema from Figure 3.2 are not nearly so obvious or modular. Each schema element would have to be examined to determine which model elements would need to be modified or deleted.

If, on the other hand, we want to extend the Province class from the existing application with another group of spatiotemporally dependent thematic attributes with shared properties; we simply add another group to the definition of the *Province* object class (or alternatively define a sub-class of *Province* containing this additional group). The same extension would be much more complicated in the schema from Figure 3.2, involving the creation of a new object class, association, and association class for the additional spatial extents and their associated timestamped thematic attributes (similar to the UML representation of the existing attribute group in Figure 3.2). This process substantially complicates the extended schema and reduces its readability.

The specification box aids readability by providing a clear and consistent framework for the detailed specification of spatiotemporal semantics. As illustrated in Figure 3.2, these semantics are represented in UML using constraints and notes. Such ad hoc notation is unlikely to be standardized among users, making the diagram more difficult to read. The specification box can serve as a guideline for application developers, highlighting generally relevant semantics to be considered when modeling spatiotemporal data. This facilitates effective communication and consistent design documentation.

3.7 Comparison to Other Modeling Languages

Contemporaneous to the research described in this thesis, several papers have been published that specifically address the need for a graphical modeling language to support conceptual design of spatiotemporal applications. In the following comparison of these modeling languages—MADS [Paren99], STER [Tryf99], and Perceptory [Brod00]—to STUML, we adopt the terminology used for STUML rather than that of the individual authors in order to avoid confusion. Furthermore, except where explicitly indicated otherwise, the following discussions apply equally to the ER or OO models even though OO terminology (attribute, object, and association rather than attribute, entity, and relationship) is used for convenience. Table 3.3 compares the spatiotemporal semantics of the four languages.

Table 3.3 Comparison of Spatiotemporal Semantics

(Abbreviations: s. = spatial, t. = temporal, s.t. = spatiotemporal)

	Language:			
Feature:	**STER**	**MADS**	**Perceptory**	**STUML**
Model	ER	Hybrid ER/OO	OO (UML)	OO (UML)
Level of spatial property support	Entity	Entity, Attribute, Relationship	Object	Attribute, Attribute group, Object, Association
Support for dependent spatio-temporal properties	Yes	No	No	Yes
Support for thematic attribute variation	s., t., s.t.	s., t., s.t.	none	s., t., s.t.
Support for multiple granularity	Explicit	Explicit	Explicit (& alternate)	Implicit (add extra spatial attributes)
Support for time dimensions	Valid, Transaction, Existence	Valid, Transaction, Existence	Valid	Valid, Transaction, Existence* (* defined)
Support for spatial constraints	Topological	Spatial	None	Topological, Spatial Part-Whole relationships
Other	Textual language equivalent	CASE tool	CASE tool, Standards-aligned	Attribute groups

A language that supports spatiotemporal properties at any level of the model allows users the flexibility to select the most appropriate modeling construct for their application. For example, the centerline in a road can be more naturally modeled as an attribute (with a spatial domain) of the spatial road object than as a separate spatial object with distinct identity. However, the road itself is more naturally modeled as a separate spatial object since it is has its own name. This shows that some spatial extents in an application may be separately identified while others are not; however, each of these two cases can be directly modeled only in languages such as STUML and MADS that can model spatial properties at both object and attribute levels.

Neither Perceptory nor STER have provision for attributes having a spatial domain. (Instead, the use of the spatial icon at the attribute level in STER indicates spatial variation in thematic attributes.) This reduces the flexibility of the resulting models since any data element associated directly with several different spatial extents must be modeled as an association of spatial objects (requiring the use of artificial identifiers for each new object) rather than a single object with several spatial attributes.

Furthermore, neither STER nor Perceptory consider associations having spatial extents (i.e. composite data that are associated with a spatial extent or whose components vary over space) and Perceptory does not support temporal associations. Instead, the association must first be promoted to an association class before adding spatial or temporal properties, introducing an artificial identifier for the new class and increasing the complexity of the resulting schema.

To illustrate the importance of spatiotemporal dependencies, consider a spatial object that is associated with an existence time separate from and independent of the valid time of its associated spatial extent (see Section 3.5). For example, assuming that existence time is used to model the time when a given hospital is open (see Section 3.5), a hospital may be associated with a spatial extent (its grounds) even when it is not open (before initial opening or during a period of closure). This can only be modeled in languages—such as STER and STUML—that provide separate constructs for independent and dependent spatiotemporal properties. Of the four languages, only STUML supports modeling of spatiotemporal semantics and dependency at all levels of the object-oriented model.

The previous discussion essentially relates to an object-based view of space. Support for a field-based view of space involves modeling variation in thematic attribute values over space and/or time. STER, MADS, and STUML provide explicit support for modeling spatial, temporal, and spatiotemporal variation in thematic attributes. In contrast, the emphasis in Peceptory is on supporting spatial objects rather than fields, as there is no explicit support for modeling spatiotemporal variation in thematic attributes.

One advantage of the other three models as compared to STUML is their explicit support (proposed in STER and MADS, implemented in Perceptory) for representing multiple spatial granularities for a single spatial object. Multiple granularities can always be represented in OO (or ER) models by defining a separate (*i*) sub-class, (*ii*) sub-type (as illustrated in [Paren99]), or (*iii*) spatial attribute (as discussed in [Brod00]) for each possible granularity in which a spatial object can be represented. However, these solutions lead to the artificial creation of extra object classes or attributes to represent the different granularities. Therefore, the most natural and flexible modeling option would be to allow multiple spatial granularities to be modeled directly in a single modeling construct (e.g. attribute, object, association). This is supported by Perceptory, MADS, and STER; represented by adding additional icons, each representing the different geometries, to the modeling construct's icon list. Perceptory further differentiates between multiple granularities, where a single object has more than one geometric representation, and alternative granularities, where a single object has a single representation with alternative options for the geometry of that representation explicitly specified. The latter case is represented graphically by physically joining geometry icons rather than listing them separately.

With respect to time dimensions for temporal properties, only valid time is supported by Perceptory; whereas existence, valid, and transaction time are supported in the other three models. Precise definitions for valid and transaction time are given in [Jens98]. However, there is no precise definition of existence time and its semantics in either MADS, STER, or in the general literature on temporal databases. The ambiguities and contradictions inherent in the common usage of existence time as lifespan are discussed in [Paren99]. To remedy this problem, a precise definition of existence dependence is given in STUML in terms of application defined dependencies and a construct, the existence-dependent construct, proposed to model such dependencies.

All of the models except Perceptory provide some degree of explicit support for expressing spatial constraints. Topological constraints on associated (or related) objects are expressed in STER and MADS by appending icons to the association. In the case of MADS a limited set of pre-defined icons is provided for the purpose. In STER, a general spatial icon on a relationship indicates the presence of a topological constraint that must then be specified textually. In STUML, as mentioned in Section 3.4.4, specific modeling constructs, called *spatial Part-Whole relationships*, are defined for associations of spatial objects with implied spatial constraints. These include a set of pre-defined modeling constructs for common spatial derivation and topological constraints between whole and part spatial objects (discussed in Chapter 5). Other topological constraints between the whole and the parts, or between parts, can be specified using the general modeling constructs defined for binary and n-ary topological relationships (discussed in Chapter 6).

Additional features provided include an equivalent textual notation for the modeling language in STER, an implemented CASE tool for MADS and Perceptory, and an attribute group construct in STUML to model a group of related thematic properties measured at the same times and locations. By explicitly modeling and graphically denoting related groups of attributes, the attribute group construct facilitates understanding of dependency semantics between attributes and provides a mechanism whereby automatic translation tools can optimize the physical design based on these dependencies. For example, consider the case where a schema with an attribute group is to be implemented. The resulting implementation need only record associated timestamps and spatial extents once for the group rather than separately for each attribute.

Moving from a discussion of semantics to their representative syntax, the graphical notation of the four models is compared in Table 3.4. Graphical icons are used to represent spatial and temporal properties and to add spatial and/or temporal semantics to an existing modeling construct in all four models; however, their application is significantly different in STER, MADS, and Perceptory than in STUML.

Table 3.4 Comparison of Spatiotemporal Syntax

(Abbreviations: s. = spatial, t. = temporal, s.t. = spatiotemporal)

Language:	**Different graphical icons used to distinguish:**	**Graphical notation used to combine icons:**
STER	Between s., t., & s.t. properties, Geometries, Multiple granularities, Time dimensions, Topological constraints, Dependent vs. independent s.t. properties	Icon listing: for all purposes
MADS	Between s. & t. properties, Geometries, Topological constraints, Spatial object versus field	Icon listing: for all purposes
Perceptory	Between s. & t. properties, Geometries (including aggregate), Multiple & alternate granularities, Time units, Attribute derivation	Icon listing: for all purposes except alternate spatial representations. Joined icon listing: alternate spatial representations
STUML	Between s. & t. properties Spatial Part-Whole relationships (Specification box used to describe all other types of properties and constraints)	Icon listing: independent s.t. properties, Icon nesting: dependent s.t. properties, Separate association icon: spatial Part-Whole relationships

In MADS, STER, and Perceptory, separate icons are used to represent different spatial geometries (e.g. point, line, region, set of geometries, aggregate geometries[7]) and multiple granularities (e.g. a land parcel represented both as a point and a region). Additional icons are used to represent different time dimensions (e.g. existence, valid, transaction, and bitemporal time), topological constraints, and

[7] A variation of icon listing, without icon borders, is used to represent aggregate geometry (e.g. a composite formed from a region and line) and distinguish it from multiple granularities.

dependent spatiotemporal semantics in STER; time units (e.g. instant and interval) in Perceptory; and spatial data types or topologies in MADS (e.g. spatial objects versus spatial fields, contains versus inclusion). As many icons as appropriate to model the relevant semantics can then be added to a given modeling construct, which we will call an *icon list* for the purposes of this discussion.

The result of this approach is a proliferation of different icons at the language definition level, i.e. defined in the model, and at the language application level, i.e. added to individual modeling constructs. This increases the complexity of both the model and the resulting schemas created using the model. Users must be able to remember all of the possible icons, squeeze them into the schema diagram when creating the schema, and read all of the resulting detail when interpreting an existing schema. For instance, four separate icons would need to be added in STER to an entity that can be represented as either a point or a region and for which both existence and transaction time is recorded. That is, the icon list for the one entity consists of existence, transaction, point, and region icons. Furthermore, some of the icons are cumbersome to draw and reproduce consistently in MADS and Perceptory. Therefore, for all practical purposes, users are limited to using the associated graphical CASE tool to draw schemas.

STUML does not have the same problems of icon proliferation at language definition and application (i.e. schema diagram) levels or graphically complex icons. To maintain the graphic simplicity of the modeling language and schemas, spatiotemporal details such as geometry, time dimension, or general topological constraints are represented textually in the specification box rather than graphically through separate icons. This corresponds well to the common cognitive process of hierarchically structuring levels of abstraction, as described in [Timpf99] for maps, so that the level of detail considered is appropriate to the current requirements. Further support for this approach comes from Bedard's experimental observation [Bedar99] that spatial database developers include minimal spatial information in the graphical schema, relying on accompanying textual specifications (e.g. in the data dictionary) to provide detailed information.

The use of spatial and temporal icons to describe both properties and constraints in STER and MADS also means that a single graphical icon or notation (i.e. method of combining icons) relates in some cases to data structure and in others to data integrity rules. For example, in STER, entities annotated with spatial icons or relationships annotated with temporal icons describe data structure (i.e. spatial extent for the entity or timestamps for the relationship), whereas relationships annotated with spatial icons describe data integrity (i.e. spatial constraints between the related spatial objects). In contrast, STUML consistently uses spatial and temporal icon annotations of an existing modeling construct to describe data structure at every level of the model, therefore ensuring consistent semantic interpretation. Note that although STUML includes additional graphical icons to model spatial Part-Whole relationships (see Chapter 5), these icons cannot be added to existing UML symbols but instead are used in place of standard UML association symbols. Therefore, the semantic difference between data structure and data integrity is clearly reflected by a completely distinct style of graphical icons and notation.

All of the languages generally share with STUML the philosophy of combining spatial and temporal constructs in an orthogonal manner to represent spatiotemporal semantics rather than adding completely new constructs and icons, thus increasing expressive power without sacrificing simplicity. However, this approach is exploited more fully in STUML by including two different but simple methods for combining the icons to represent two fundamentally different semantic concepts. Icon nesting is used to represent dependent relations between value domains (e.g. a temporally dependent spatial object) whereas icon pairing (where pairing is equivalent to an icon list with exactly two icons) is used to represent independent relations between value domains (e.g. a spatiotemporal object). Nesting spatial and temporal icons represents a functional composition from one domain to the other; whereas, pairing the same icons represents two independent functions.

To illustrate, consider the example of hospital in Figure 3.7 (c). The pairing of symbols shows that independent information is recorded for the temporal and spatial properties of an object, for example, the time dimensions and associated spatial extent of an object respectively. The symbol nesting shows that the spatial extent associated with the object is dependent on the time it is recorded, i.e. can vary over time.

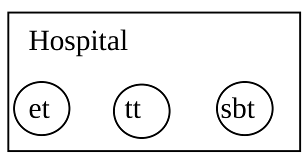


Figure 3.11 Representation of Hospital in STER

Figure 3.11 shows how the same hospital would be represented in STER, assuming that both existence and transaction time is recorded for the object and that both valid and transaction time is recorded for the object's associated spatial extent. Object existence time and transaction time is represented by the *et* and *tt* icons respectively; whereas the spatial extent of the object is represented by the *sbt* icon. In this case, the same graphical technique—icon listing—is used to represent two different semantic concepts—that of multiple time dimensions and that of combining spatial and temporal properties to represent independent spatiotemporal data properties. (Note that dependent spatiotemporal properties are represented in STER by creating a new icon, *sbt*. There is no provision representing this concept or distinguishing between dependent and independent spatiotemporal semantics in MADS or Perceptory, so a spatial icon would be used instead.) The same type of ambiguity is found in MADS and Perceptory. In contrast, STUML represents base spatiotemporal semantics by combining spatial and temporal icons (using two different graphical notations to represent dependent and independent semantics) and details such as time dimensions, time units, or spatial geometry by standard textual notation in the associated specification box.

Finally, the introduction of the thematic symbol in STUML aids consistency by permitting consistent representation of functional composition semantics, regardless of whether thematic, spatial, and/or temporal domains are involved. It also makes graphically explicit the fundamental semantic distinction between attributes having a spatial domain and thematic attributes whose values depend on space.

3.8 Summary

In summary, this chapter proposed a UML extension to support applications requiring a range of spatiotemporal models and types. A clean technique was introduced for modeling *composite data whose components vary depending on time or location, temporal changes in spatial extents,* and *changes in the value of thematic data across time and space.* The proposed extension supports alternative models of time and change processes, as well as valid, transaction, and existence time dimensions. By introducing a small base set of modeling constructs that can be combined and applied at different levels of the UML model (including attribute groups), language clarity and simplicity is maintained without sacrificing expressive power or flexibility.

The introduction of a thematic symbol and formal rules for combining spatial, temporal, and thematic symbols provides a consistent and well-defined notation for representing spatiotemporal semantics. Temporal and spatial associations are treated in a parallel manner, i.e. to describe model structure. Attribute groups are introduced to explicitly model related groups of thematic attributes with common spatial and/or temporal properties.

In addition, existence time has been precisely defined based on application-defined dependencies of individual object properties and modeling constructs introduced to reflect these semantics. This allows users to differentiate between those properties that *are* still defined when the object does not exist (e.g. employee social security number) and other properties that *are not* (e.g. work phone number).

STUML is clearly distinguished from other graphical modeling languages proposed for spatiotemporal applications. The distinctive characteristics of STUML include:

- support for representation of spatiotemporal semantics and dependency at all levels of the object-oriented model; and
- a consistent and simple mapping between graphical representation and semantics with a two-layer representation of spatiotemporal semantics (using graphical icons and the specification box) at different levels of abstraction,
- the definition of an attribute group construct to explicitly model a related group of thematic attributes with common spatial and/or temporal properties, and

- the precise definition of existence time and its associated modeling construct in terms of application-defined dependencies.

In Chapter 4, formal rules for mapping a STUML schema to an equivalent UML schema are given. These transformation rules provide a theoretical basis for implementing STUML schemas using tools and products developed for UML. They could also be used as a basis for implementing spatiotemporal extensions to existing UML case tools such as Rational Rose [Quat98, Ratio98a, Ratio98b], which are used to automatically convert a conceptual into a logical (implementable) and then a physical (implemented) schema.

Chapter 4

Mapping STUML to UML

4.1 Introduction

In this chapter, we present a comprehensive set of transformation rules for mapping a schema expressed in STUML to an equivalent schema in UML. The discussion of alternative UML representations of spatiotemporal semantics in Section 3.3.1 provides a foundation for developing such rules, specifically formulated to support conversion from STUML to UML at the conceptual level and facilitate the potential exploitation of existing UML development tools for use with STUML.

As discussed in [Bedar99, Booch99, Rumb99], there may be cases where a purely conceptual representation in STUML or UML may be sufficient to satisfy the requirements of the end user. This is the case, for example, when the conceptual modeling language is being used primarily as an aid to thought processes, visualization, documentation, communication, or high-level specification. For these purposes, the focus of the modeling abstractions are on global and real-world (application) views of data and their inter-dependencies. The utility and understandability of these abstractions depends on limiting the information included to the essential and relevant real-world characteristics in the application domain context. Implementation details relating to physical representation or efficiency concerns are omitted, as they would unnecessarily complicate the schema (since these details are not important and may not even be known at this stage). At this stage, it is sufficient and even desirable to represent spatial and temporal data as abstractions, e.g. the representations of spatial extents and timestamps introduced in 3.3.1.

In other cases, the goal is to realize a software implementation through the transformation of these high-level models to detailed, implementable, low-level specifications and then to code, using either manual or automated methods or some combination of the two. Implementation issues related to efficient data representation and access play a critical role in this transformation process. Such issues are even more critical in the context of spatiotemporal applications, given the potential for generation of large volumes of data and the inherent complexity of spatial data. Many different discrete representations for spatial data have been proposed, including boundary (e.g. vectors, arcs and nodes, or polygons), raster, and tessellated (i.e. describing a surface in terms of non-overlapping polygons) representations. Continuous representations in terms of mathematical models have also been proposed. The selection of a suitable spatial representation and related data access mechanisms are highly application dependent. Such considerations are outside the scope of this thesis and are not discussed further here, but extensive discussions of these issues can be found in [Worb95] for spatial, [Tans93] for temporal, and [Lang93] for spatiotemporal data.

Assuming that the goal of conceptual modeling is eventual translation into implementable designs, the transformation rules presented in this chapter provide a theoretical basis for implementing STUML schemas using tools developed for UML or extending those tools with spatiotemporal extensions based on STUML. For example, the transformation rules presented in this chapter could be used to adapt an existing UML case tool, which can be used to generate an implementable schema from a conceptual schema, for use with STUML schemas. Where there is more than one equivalent UML schema; the particular alternative selected can influence the effectiveness of any subsequent automated or semi-automated generation of an implementation using a case tool. This chapter, therefore, describes a set of guidelines for mapping STUML to UML that are generally applicable in this context and designed to minimize redundancy in any implementation generated from the resulting UML schema.

The general mapping approach adopted is described in Section 4.2. The transformation rules for spatiotemporal properties at the attribute (and attribute group) level is given in Section 4.3 for object attributes and Section 4.4 for association attributes. The transformation rules for spatiotemporal properties at the object level are given in Section 4.5 and at the association level in Section 4.6. The rules are summarized in Section 4.7.

4.2 An Overview of Transformation Rules

The transformation rules presented in this chapter were formulated based on two primary considerations: (*i*) general applicability and (*ii*) minimizing redundancy to reduce implementation storage requirements. First, the transformation rules are formulated assuming that the resultant UML schemas use atomic rather than composite attribute domains. The use of composite attribute domains results in more compact schemas and reduces the need for artificial identifiers. However, it represents a less general solution than using atomic domains since many models (e.g. ER, relational), case tools, database management systems, and programming languages still support only primitive data types as attribute domains.

To avoid reliance on composite attribute domains, one option is to translate composite attribute domains to separate attributes within the same class. However, as discussed in Section 3.3.1, this does not allow identical composite attribute instances (i.e. data values) to be shared rather than copied. A more general solution that allows sharing of attribute instances is to model composite attribute domains using separate classes. This is particularly important in the case of storage-consuming spatial data, illustrated in Figure 3.2 by a single *Measurement-Region* whose thematic attribute values vary over time. Such considerations lead to the second principle used here to formulate transformation rules: selecting design approaches that can be expected to minimize redundancy in the resulting implementation. This is analogous to the primary motivation for normalizing relational databases. (Note the contrast with the stated goals and priorities for STUML schemas, that they be simple, clear, and consistent without consideration of redundancy in later development stages.) Although there may be individual cases where efficiency concerns and application priorities require controlled redundancy to be introduced in later stages of application development, the general approach adopted in this chapter is to adopt strategies that can minimize redundancy in later development stages.

There are four different UML modeling templates used in the transformation rules, as shown in Figure 4.1. The label *Class* or *Association-Class* is used to show classes present in the original STUML schema before transformation. All the other classes shown in the figures are added during the transformation process. Figure 4.1 shows the template used when a spatiotemporal property is represented using (a) an existing class, (b) a new associated class, (c) a new associated class and association class (or a new association class for an existing association whose label is indicated in parentheses, or an existing association and association class whose labels are both indicated in parentheses), or (d) a new ternary association (created from an existing binary association) and association class.

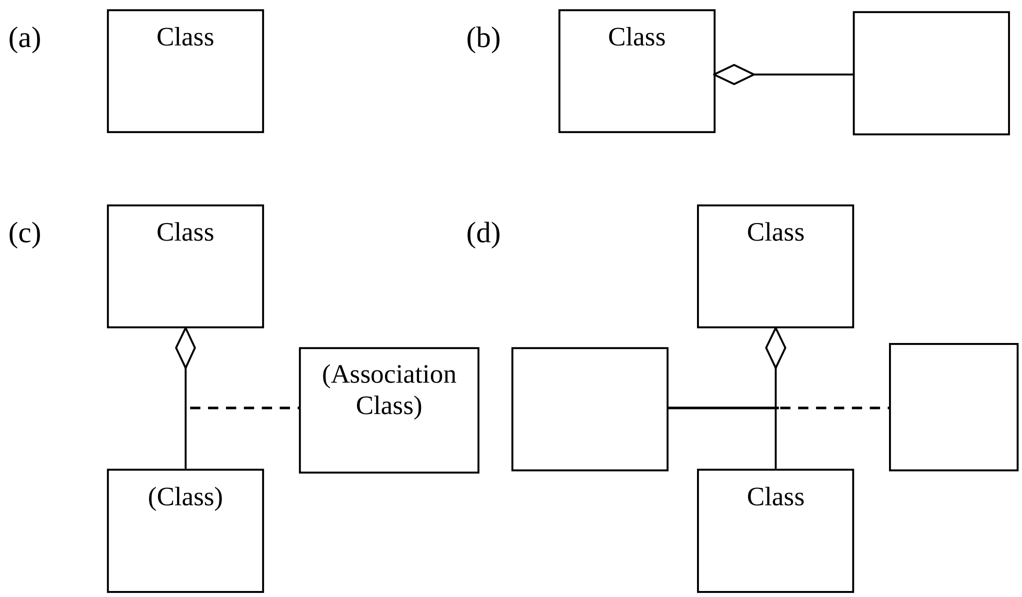


Figure 4.1 Templates for Modeling Spatiotemporal Data in UML

Note that artificial identifiers must be created for each new object or association class created, as discussed in Section 3.3.1 and illustrated in Figure 3.2. However, these identifiers are not shown in the figures in this chapter for the sake of readability. A spatial or temporal property is represented by an additional attribute, if one does not already exist. The value of the attribute is either a spatial extent (to represent a spatial property) or a timestamp (to represent a temporal property). The abstractions that were introduced in 3.3.1 for spatial extents and timestamps are used in this chapter to represent primitive (i.e. *atomic*) spatial and temporal data types. They are reproduced here (in BNF notation) for reference:

spatial-extent := { point | line | region | volume }^{n}

timestamp := { instant | interval | element }

Since the transformation rules do not rely on composite attribute domains; each timestamp, spatial extent, or thematic attribute value is represented as a separate attribute in UML. As discussed in Section 3.3.1 and illustrated in Figure 3.2, information contained in the specification box (e.g. time units for timestamps, dimensions for spatial extents, derivation rules) and existence-dependencies represented by the existence-dependent construct in STUML would be translated to UML constraints. In this chapter, the focus is on translation of the fundamental STUML constructs, i.e. the base spatiotemporal semantics represented by combinations of the spatial, temporal, and thematic symbols. The rules represent spatiotemporal properties in UML using one of the modeling templates from Figure 4.1 and adding attributes as required, indicated by including the appropriate attribute domain(s) in one or more classes of the template.

In general, the alternative preferred in the majority of cases will be incorporated into the transformation rule. Only if each alternative offers clear advantages, but for different cases, will more than one alternative be included in the set of mapping rules with accompanying selection guidelines. Unless specified otherwise in the caption, sub-figures in a figure show alternative transformation rules. Earlier sub-figures show the simpler and later sub-figures show the more general alternatives respectively. For example, Figure 4.1 (a) is the simplest and 4.1 (d) the most general solution. Similarly, in Section 4.3, Figure 4.2 (a) is simpler than 4.2 (b) and, in Section 4.5, Figure 4.8 (b) is simpler than 4.8 (c).

4.3 Converting Object Attributes

In STUML, object attributes with spatial or temporal properties are represented using one of the following modeling constructs: *spatial attributes, temporally dependent spatial attributes,* and *spatially, temporally,* or *spatiotemporally* *dependent thematic attributes*. The thematic attributes may be members of an attribute group.

Figure 4.2 shows two alternative rules for transforming an object's *spatial attribute* (an attribute having only spatial properties) in STUML to an equivalent representation in UML. If every class instance has a unique spatial extent distinct from that in any other class instance, then we can use the simple option of converting the spatial attribute to its UML equivalent. Thus the spatial property is still represented directly as an attribute in the class, as shown in Figure 4.2 (a). The new UML attribute has a spatial domain whose values are spatial extents. An example of using this transformation rule is the transformation of the spatial attribute *location* in the STUML schema shown in Figure 3.10 to an attribute of the same name whose value is a spatial extent in the UML schema shown in Figure 3.2.

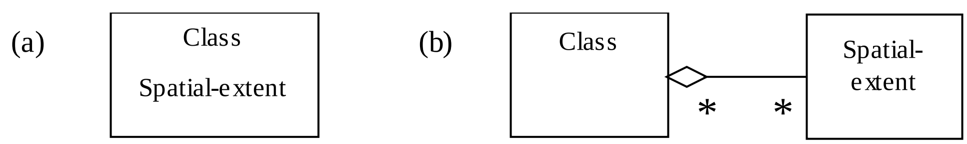


Figure 4.2 Modeling Attributes having only Spatial Properties in UML

If class instances can share the same spatial extent, then the representation shown in Figure 4.2 (a) would lead to duplication of the same spatial extent for each different class with that spatial extent. To eliminate this source of redundancy from the design, a new class is instead created for the spatial property and associated with the original class, as shown in Figure 4.2 (b). The spatial extent is represented as an attribute value in the new class. Applying this transformation rule to the spatial attribute *location* from Figure 3.10 would require that the *location* attribute from Figure 3.2 be modeled in a separate class, newly created for the purpose and associated with the hospital class.

To transform temporally dependent spatial attributes, spatially dependent thematic attributes, and spatiotemporally dependent thematic attributes of objects in STUML to UML equivalents, the spatial property (i.e. the spatial extent) is represented as an attribute in a new class associated with the original class. This is illustrated in Figure 4.3 (a).

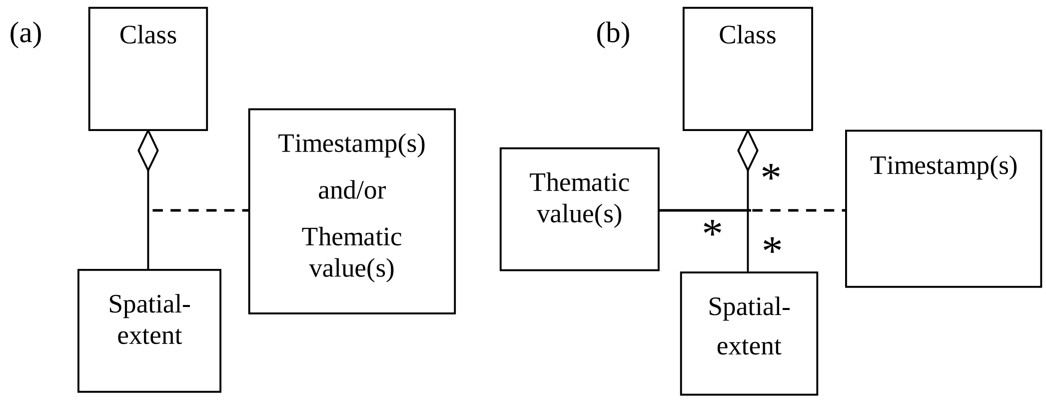


Figure 4.3 Modeling Attributes having Spatial and Other Properties in UML

A new association class is added to the association and used to represent the remaining temporal and/or thematic properties, where the temporal property consists of one or more timestamps (e.g. valid and/or transaction timestamps) and the thematic property consists of one or more thematic values (there will be several thematic values for each timestamp in the case of an attribute group). So given a temporally dependent spatial attribute, each timestamp would be represented as a separate attribute in the new association class. In the case of a spatially dependent thematic attribute, each thematic value would be represented using a separate attribute in the new association class. For spatiotemporally dependent thematic attributes, both the timestamp(s) and thematic value(s) would be represented as attributes in the association class. This transformation rule is illustrated by the transformation of the spatiotemporally dependent attribute group consisting of *populationDensity* and *averageLifespan* in the STUML schema shown in Figure

3.10 to attributes in a new association class *Has* and associated class *Measurement-Region* in the UML schema shown in Figure 3.2.

Using the representation in Figure 4.3 (a) for spatiotemporally dependent thematic attributes means that the same thematic attribute value(s) are duplicated for each different timestamp with those value(s), since the thematic and temporal properties are modeled together in a single class. There may be some cases where the thematic property value(s) are large enough that it is desirable to be able to model the thematic property separately from the timestamp(s) in order to eliminate this source of redundancy from the schema design. This is likely to be the case, for example, whenever the thematic property consists of an attribute group. In this case, a new ternary association is used to represent both the thematic value(s) and spatial-extent as separate classes associated with the original class. The association class is used to represent the temporal properties. This approach is illustrated in Figure 4.3 (b), using two new classes associated with the original class in a ternary relationship to represent the spatial and thematic properties respectively and a new association class to represent the temporal properties of a spatiotemporally dependent thematic attribute from STUML in UML. Applying this transformation rule for the spatiotemporally dependent attribute group in Figure 3.10 would have required the creation of a new class in Figure 3.2 for the thematic values in the attribute group (i.e. *populationDensity* and *averageLifespan*) and the addition of the new class to the existing association between *Province* and *Measurement-Region*, creating a ternary association with the same association class *Has*.

Temporally dependent thematic attributes of objects in STUML are transformed to UML as shown in Figure 4.4. As discussed for spatiotemporally dependent thematic attributes, if the thematic values are large and repeated for different timestamps then it is desirable to separate the modeling of thematic from the temporal properties. This is illustrated in Figure 4.4 (b) where the thematic property is modeled as a new class associated with the original class and the temporal property is modeled as a new association class. Otherwise, the simpler mapping approach illustrated in Figure 4.4 (a) can be employed, where both the thematic and temporal properties are modeled in the new class. Consider the example of the temporally dependent thematic attribute *numBeds* from the STUML schema shown in Figure 3.10. To apply these transformation rules to the UML schema in Figure 3.2, the integer and valid-timestamp values would be represented using separate attributes (i.e. *numBeds* for the integer value and a new attribute for the valid timestamp) and a new class would have to be created and associated with the *Hospital* class. Using the simpler transformation rule from Figure 4.4 (a), both the integer and valid-timestamp attributes would be added to the new class associated with the *Hospital* class. Using the more general transformation rule from Figure 4.4., only the integer attribute would be included in the new class. The valid-timestamp attribute would be included in a newly created association class added to the association with the *Hospital* class.

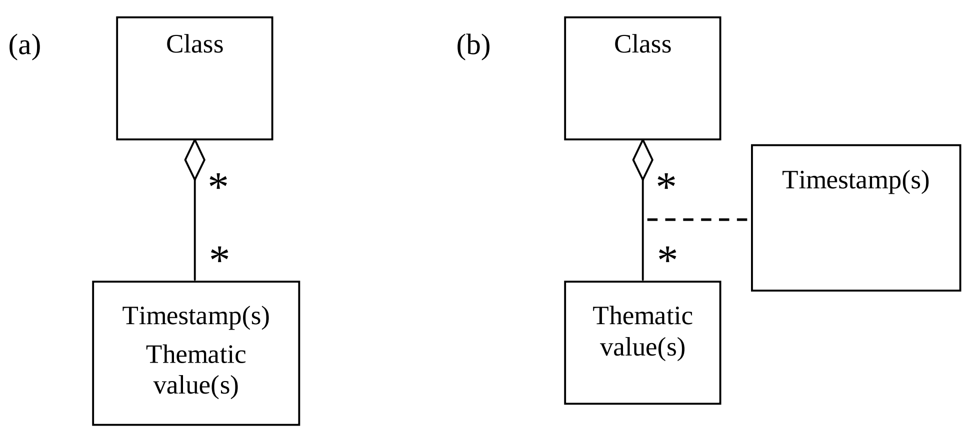


Figure 4.4 Modeling Attributes having Temporal and Thematic Properties in UML

4.4 Converting Association Attributes

As is the case with object attributes, associations in STUML can have *spatial attributes*, *temporally dependent spatial attributes,* and *spatially, temporally,* and *spatiotemporally dependent thematic attributes*. The thematic attributes can be members of an attribute group. Following the standard UML convention, attributes are added to associations in STUML by attaching an association class with the attributes as discussed in Section 3.4.2. This is the case whether the attribute has spatiotemporal properties or not.

Consider the case of classes linked by an association and an attached association class that has an attribute with spatial and/or temporal properties (i.e. one of the five types of attributes listed above). To transform this attribute from STUML to UML, the most general solution is to promote the association class to a full object class with the same attributes and add the new object class to the existing association (increase the degree of the association by one). Any attribute with spatial and/or temporal properties can then be converted to UML using the transformation rules for attributes of objects described in Section 4.3.

Using this approach, the association class *In* from Figure 3.6 would be transformed to an object class of the same name with a temporally dependent thematic attribute *rank*. The binary association between the *Hospital* object class and the *Province* object class would then be changed to a ternary association between the *Hospital* object class, *Province* object class, and the new *In* object class. Assuming that *rank* is associated with valid and transaction timestamps, the temporally dependent attribute *rank* in the new *In* object class can then be converted to UML using the transformation rule described in Section 4.3 for temporally dependent thematic attributes of objects. Assuming that the transformation rule depicted in

Figure 4.4 (a) is used (since the value of *rank* is an integer), the result of this two-stage conversion process is illustrated in Figure 4.5 (b). (Note that the many-to-one (*:1) cardinality from the original binary association between *Hospital* and *Province* is no longer evident from the schema. This would have to be explicitly specified as an additional constraint using the UML convention of enclosure in curly braces.)

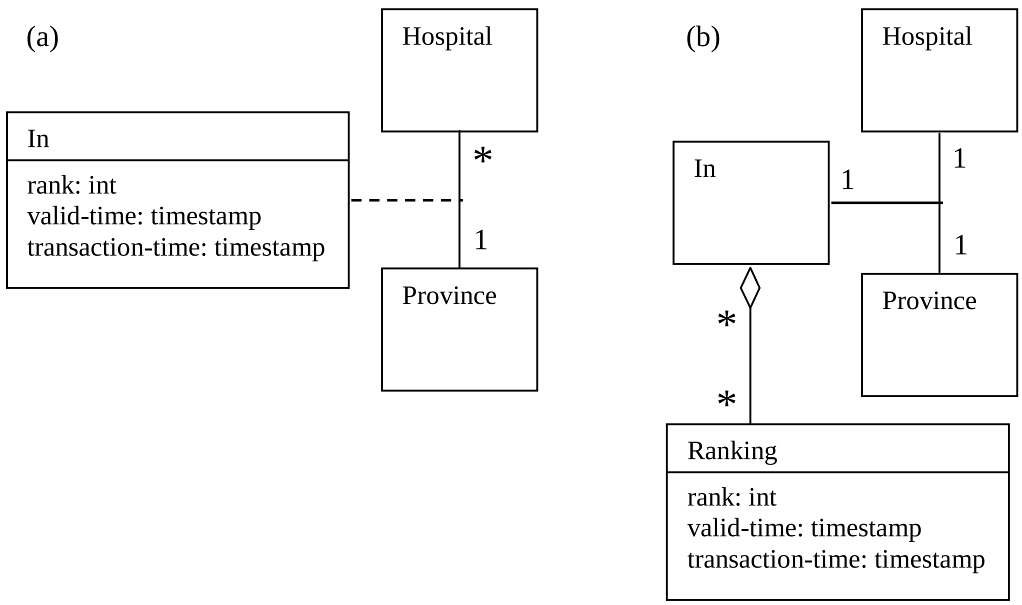


Figure 4.5 Example of Modeling a Temporally Dependent Thematic Attribute in UML

If the STUML association class *In* had additional thematic attributes, then these would be directly transferred to the new UML object class *In*. If the STUML association class *In* had additional attributes with spatial and/or temporal properties, then each of these attributes would first be transferred to the new object class *In* and then converted separately using the rules described in Section 4.3 for attributes of objects. This is the same two-stage conversion process that was illustrated using the temporally dependent attribute *rank*.

If an association class has only one spatial attribute or temporally dependent thematic attribute (as is the case with *In*) where the thematic value is small (as is the case with *rank*), then a simpler transformation rule can be used. The rule is analogous to the simpler transformation alternatives described previously for spatial attributes and temporally dependent thematic attributes, illustrated in Figure 4.2 (a) and Figure 4.4 (a) respectively. Instead of promoting the association class to an object class, the spatial or temporally dependent thematic attributes are converted to their UML equivalents within the association class. Thus the spatial, temporal, and thematic properties are still represented directly as attributes in the association class, as shown in Figure 4.6 (a) for spatial attributes and Figure 4.6 (b) for temporally dependent thematic attributes. To convert from STUML to UML, a spatial attribute in an association class would simply be converted to a new UML attribute in the same association class having a spatial extent as its value. Similarly, a temporally dependent thematic attribute from an association class would be converted to new UML attributes each having a timestamp or thematic value. This simpler conversion process is illustrated in Figure 4.5 (a) for the temporally dependent attribute *rank* from the association class *In* (from the STUML schema in Figure 3.6).

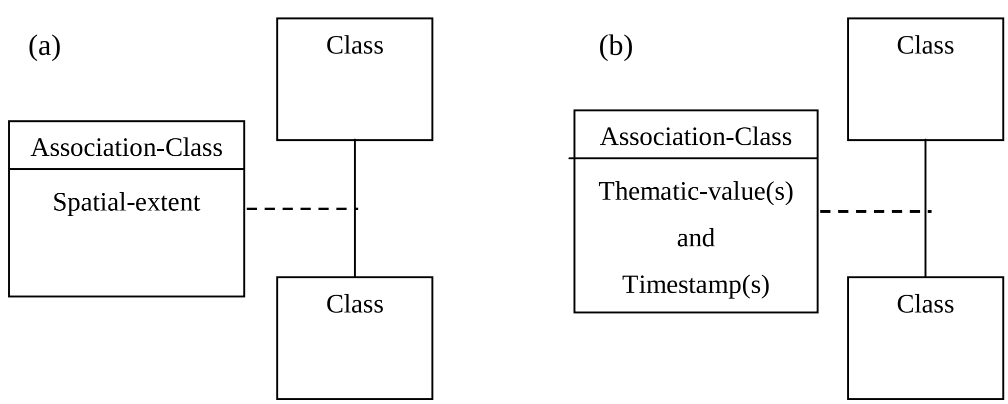


Figure 4.6 Modeling Spatial (a) or Temporally Dependent Thematic (b) Attributes in UML

4.5 Converting Objects

In STUML, an object with spatial or temporal properties is represented using one of the following STUML modeling constructs: *spatial object, temporal object, spatiotemporal object*, *temporally dependent spatial object*, or *temporally dependent spatiotemporal object.*

Transforming a spatial object in STUML, an object associated with a spatial extent, is essentially the same as transforming a spatial attribute of an object in STUML. The only difference is that the spatial property of the object in STUML is converted to UML by explicitly creating a new attribute (whose value is a spatial extent), since one does not already exist. If every class instance has a unique spatial extent distinct from that in any other class instance, then the new attribute is added to the existing object class; otherwise, a new object class is created and the new attribute is added to that class. The alternative transformation rules are illustrated by Figure 4.7 (a) and (b) respectively. This is identical to Figure 4.2 for spatial attributes, but reproduced here for convenience and consistency. Assuming a STUML schema has a spatial object *Hospital* (i.e. denoted by the use of a spatial symbol in the class name compartment), the use of the *location* attribute in the UML schema shown in Figure 3.2 illustrates the simpler transformation rule for spatial objects from Figure 4.7 (a). If the more general transformation rule from Figure 4.7 (b) were used, then a new class associated with *Hospital* would have to be created for the *location* attribute in Figure 3.2.

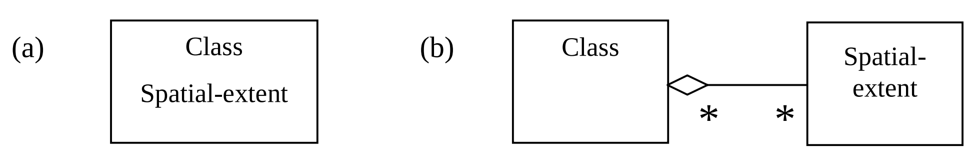


Figure 4.7 Modeling Objects having only Spatial Properties in UML

A temporal object in STUML, associated with existence and/or transaction time, is converted to UML by representing the temporal property as an attribute having timestamp value(s) in the original object class as illustrated in Figure 4.8 (a). This is analogous to the simple transformation rule for a spatial object in Figure 4.7 (a). The transformation rule from Figure 4.8 (a) is illustrated by the transformation of the temporal object *Hospital* in the STUML schema shown in Figure 3.10 to the *hospital-existence-time* and *hospital-transaction-time* attributes in the UML schema shown in Figure 3.2.

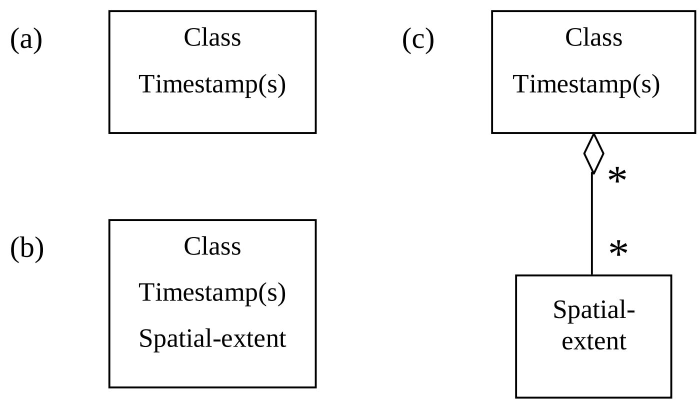


Figure 4.8 Modeling Temporal (a) and Spatiotemporal (b,c) Objects in UML

The transformation rule for a spatiotemporal object is a combination of the rule for transforming a temporal object, shown in Figure 4.8 (a), and either alternative rule for transforming a spatial object, shown in Figure 4.7 (a) and 4.7 (b). The result is new attributes having either timestamp or spatial-extent values. Note that, for a spatiotemporal object, the timestamp is not related to the spatial extent but rather to the object itself. If the spatial extent is unique to each class, then both attributes can be modeled in the original class, as illustrated in Figure 4.8 (b); otherwise, a new class is created for the spatial extent and associated with the original class, as illustrated in Figure 4.8 (c).

If we assume that a STUML schema has a spatiotemporal object *Hospital* (i.e. denoted by the use of a spatial and a temporal symbol placed next to each other in the class name compartment), then the simpler transformation rule for spatiotemporal objects shown in Figure 4.8 (b) is illustrated by the use of the *hospital-existence-time*, *hospital-transaction-time*, and *location* attributes in the UML schema shown in Figure 3.2. If the more general transformation rule from Figure 4.8 (c) were used, then a new class associated with *Hospital* would have to be created for the *location* attribute in Figure 3.2.

To transform a temporally dependent spatial (or spatiotemporal) object, where the object's spatial extent is associated with valid and/or transaction time; the temporal property of the spatial extent is represented by new attribute(s) having timestamp value(s) in a new association class. The new association class is added to the association shown in Figure 4.7 (b) for a spatial object or in Figure 4.8 (c) for a spatiotemporal object. (The more general transformation rule for spatial and spatiotemporal objects is used, assuming that the same spatial extent can be repeated for different timestamps.) This is illustrated in Figure 4.9 (a) and (b) respectively

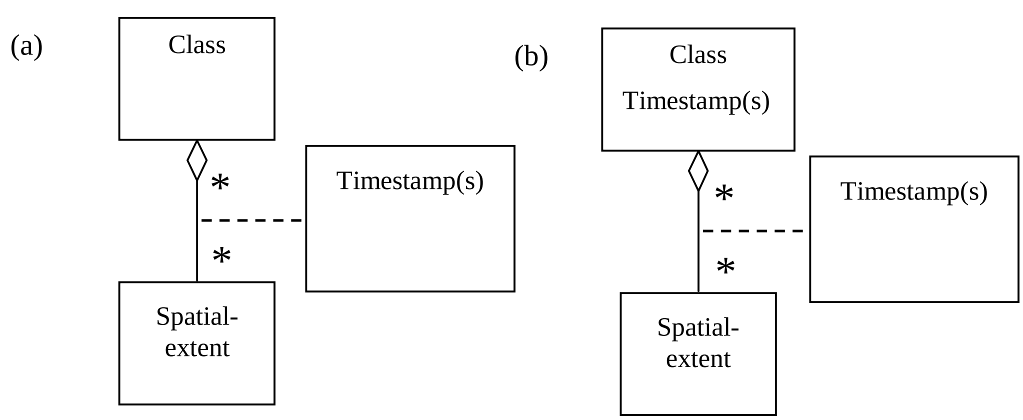


Figure 4.9 Modeling Temporally Dependent Spatial (a) and Spatiotemporal (b) Objects in UML

Consider the temporally dependent spatiotemporal object *Hospital* from Figure 3.7 (c), assuming that the existence and transaction times are recorded for the hospital and valid time for its spatial extent. Figure 4.10 shows the results of applying the transformation rule shown in Figure 4.9 (b) to *Hospital* (the attribute *numBeds* is not shown). As described previously for spatial objects and spatiotemporal objects, a new class associated with the *Hospital* class is created for the *location* attribute. Thus the *Hospital* instance need not be changed every time that the hospital location changes. We assume that a new hospital could be built on the site of an old hospital (i.e. one that no longer exists): thus *M:M* cardinality is shown between the *Hospital* and *Hospital-location* classes. As described previously for temporal objects, the *hospital-existence-time* and *hospital-transaction-time* attributes in the *Hospital* class are used to represent the hospital existence and transaction time. Finally, the valid time of the hospital's spatial extent is represented by the *valid-time* attribute in the new association class *Has*. The conversion of a temporally dependent spatial object using the transformation rule shown in Figure 4.9 (a) would look the same, except that the *hospital-existence-time* and *hospital-transaction-time* attributes would be omitted.

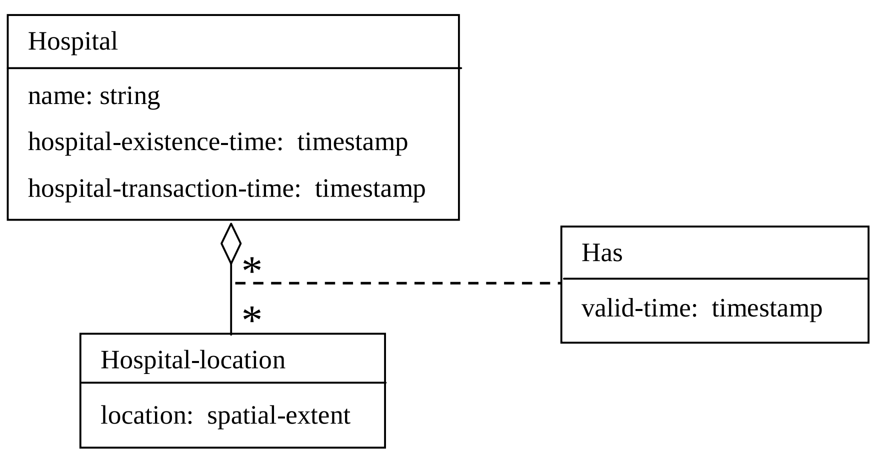


Figure 4.10 Example of Modeling a Temporally Dependent Spatiotemporal Object in UML

4.6 Converting Associations

In STUML, an association with spatial, temporal, or both spatial and temporal properties is represented as a *spatially dependent association*, *temporally dependent association*, or a *spatiotemporally dependent association* respectively. To transform a spatially or temporally dependent association from STUML to UML, new attributes are created for the spatial extent or timestamp(s) respectively. A new association class is then created for the new attribute(s) and associated with the original association, illustrated in Figure 4.11 (a). The result of applying this rule to the temporally dependent association *Is-of* in the STUML schema from Figure 3.10 is the *Is-of* association class (with its attributes) in the UML schema from Figure 3.2. Note that if an association class already exists for the original association (i.e. if the association has additional thematic attributes without spatial or temporal properties), the new attribute(s) can be directly added to the existing association class. However, if spatial extents are likely to be repeated for different thematic attribute values; then it is preferable to follow the rule for spatiotemporally dependent associations described below (shown in Figure 4.11 (b)) and create another object class for the spatial extent (with the extra thematic attributes in the association class).

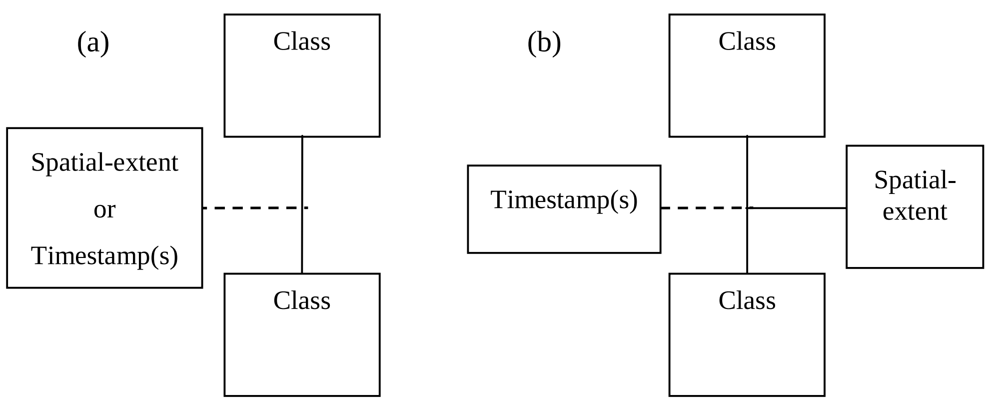


Figure 4.11 Modeling Spatially or Temporally Dependent (a) and Spatiotemporally Dependent (b) Associations in UML

A spatiotemporally dependent association can be associated with the same spatial extent at different times. Therefore, to allow sharing rather than duplication of the spatial extent, the spatial extent and timestamp(s) shown modeled together in one class in Figure 4.11 (a) should instead be modeled in two separate classes. Therefore, the transformation rule for a spatiotemporally dependent association has one additional step as compared to the rule just described for spatially or temporally dependent associations. Instead of including the spatial extent in the new association class, another object class is created for the spatial extent and added to the original association. That means that one degree is added to the original association (e.g. a binary association becomes a ternary association). The resulting transformation rule for spatiotemporally dependent associations is illustrated in Figure 4.11 (b). If the original association had any extra thematic attributes without spatial or temporal properties, they can be added to the association class with the timestamps.

For example, the result of applying the rule illustrated in Figure 4.11 (b) to the spatiotemporally dependent association *Contains* in the STUML schema from Figure 3.10 is the ternary relationship and *Contains* association class in the equivalent UML schema from Figure 3.2. If the spatial extent associated with an administrative region were added as an additional attribute to the new association class *Contains*, then the spatial extent would have to be duplicated whenever there is a change in the regulations affecting ward-categories (even if the administrative region in which the regulation applies hasn't changed). Adding another class for the administrative region allows the associated spatial extent to be shared rather than duplicated.

4.7 Summary

In this chapter, a comprehensive set of transformation rules has been presented for mapping STUML schemas to equivalent UML schemas. The transformation rules provide a theoretical basis for implementing STUML schemas using tools developed for UML, for example, either by mapping the STUML schema to a format suitable for an existing UML case tool or by extending the case tool with a spatiotemporal extension. The case tool can then be used to generate an implementable schema. The transformation rules that have been presented were formulated to facilitate such a process. In general, it is possible to find more than one UML schema equivalent to a given STUML schema. The priority has been to find an UML mapping that provides a general solution (i.e. does not require the use of or assume software support for composite attribute domains) and that is likely to minimize or reduce redundancy in the physical implementation generated from the conceptual UML schema.

The transformation rules are summarized in Table 4.1. For each STUML construct, the table shows the new attributes for the existing class, new associated object class(es), and/or a new association class required to represent the STUML construct in UML. Timestamps describe attributes, associations, objects, or an object's spatial extent. Adding one (e.g. for a spatial object) or two (e.g. for a spatiotemporally dependent thematic attribute) new associated object classes to an existing object class in a schema results in a new binary or ternary association respectively. Adding one new associated object class to an existing association in a schema (i.e. for a spatiotemporally dependent association or an association having attributes with spatial and/or temporal properties) results in an increase in the association degree by one.

Table 4.1 Summary of STUML to UML Transformation Rules

(Notes: * = increase degree of existing association by 1)

STUML construct	**Existing Class**	**New Associated Class(es), default=1**	**New Association Class**
For objects: Spatial attribute	[spatial extent]	spatial extent	
For objects: Temporally dependent spatial attribute or Spatially dependent thematic attribute		spatial extent	timestamp(s) or thematic value(s)
For objects: Spatiotemporally dependent thematic attributes		2 classes: 1st: spatial extent 2nd:thematic value(s)	timestamp(s) [and thematic value(s)]
For objects: Temporally dependent thematic attributes		thematic value(s) [and timestamp(s)]	timestamp(s)
For associations: Temporally dependent thematic attribute or Spatial attribute	Promote existing association class to object class* and follow rule for object attribute.		[(timestamp(s) and thematic value(s)) or spatial extent]
For associations: Temporally dependent spatial attribute Spatially dependent thematic attribute Spatiotemporally dependent thematic attributes	Promote existing association class to object class* and follow rule for object attribute.		
Spatial object	[spatial extent]	spatial extent	
Temporal object	timestamp(s)		
Spatiotemporal object	object timestamp(s) [and spatial extent]	spatial extent	
Temporally dependent spatial object		spatial extent	timestamp(s) of spatial extent
Temporally dependent spatiotemporal object	object timestamp(s)	spatial extent	timestamp(s) of spatial extent
Spatially dependent association or Temporally dependent association			spatial extent or timestamp(s)
Spatiotemporally dependent association		spatial extent*	timestamp(s)

Square brackets indicate a simpler alternative rule when each value of the bracketed type is expected to be unique or, if not unique, of small enough size that duplication is not a problem. For example, the STUML spatial attribute construct for objects has square brackets enclosing the *spatial extent* type under the *existing class* column. This means that if each spatial attribute value is unique to a single object instance, the spatial extent for a spatial attribute can be represented as an attribute in the existing object class rather than as an attribute in a new associated object class.

The STUML language and its mapping to UML have been presented thus far in the thesis, providing general support for conceptual modeling of spatiotemporal applications. Such applications are further characterized by the need to manage complex spatial objects formed from spatial sub-units. STUML provides support for modeling spatial objects and associations; however, there are no explicit provisions for modeling the different types of complex spatial objects commonly seen in geographic and other spatial applications. In the next chapter, the specific problem of modeling complex spatial objects is investigated.

Chapter 5

Modeling Part-Whole Relationships for Spatial Data

5.1 Introduction

One of the characteristics of spatial (and spatiotemporal) applications is the need to record and manage complex relationships between spatial objects. Some of these relationships involve the aggregation of spatial objects into a larger composite; for example, a collection of countries aggregated into a supranational organization such as the European Union (EU) or North Atlantic Treaty Organization (NATO). In the same way, a national land transport network consists of separate rail, bus, and road networks. Such relationships may lead to hierarchies of composite spatial objects. For example, a road network is itself composed of separate roads. In all of these cases, the spatial extent of one object (e.g. a supranational organization, national land transport network, or road network) is derived from the spatial extents of its components (e.g. member countries, individual transport networks, or individual roads respectively); however, there are other attributes (e.g. the location of the organization headquarters, the budget for the road network) that cannot be derived.

Other complex relationships may involve spatial constraints, rather than derivation, between spatial objects. For instance, the spatial extent (i.e. area) of a building site must include all of the buildings and other structures on the site. Note that the spatial extent of the building site cannot necessarily be derived from those of its contained structures since it may contain empty space not occupied by any structure. However, other attributes of the building site such as installed power or percentage green space[1] will be so derived. As in the earlier examples, the building site also has attributes, such as purchase date or price, that are independent of the attributes of its contained buildings or structures. Examples of other types of spatial constraint relationships include a guaranteed mobile phone service area that must be completely covered by the combined ranges of phone service cells and an administrative area that must be equal to the combined area of its land use zones.

We can see that these examples all involve an asymmetric (i.e. *non-peer*) relationship between spatial objects, where one object (i.e. the *whole*) can be regarded as, in some sense, representative of or an abstraction of a group of other objects (i.e. the *parts*). Thus a reference to the EU represents an implicit reference to its group of member countries. However, the whole is more than just the sum of its parts, since it has some properties such as EU headquarters not dependent or derviable from those of its parts. In this thesis, we refer to such relationships generally as *Part-Whole* (*PW*) *relationships* or, when the related objects are *spatial objects* (i.e. associated with a spatial extent), *spatial PW relationships.* These include relationships commonly called aggregation, part-of, part/whole, composition, or membership relationships in the literature.

In the research presented in this chapter, we adopt a practical approach to defining spatial PW relationships. The intention is to facilitate conceptual analysis and design of spatial applications by providing explicit support for modeling spatial PW relationships. This work integrates efforts from the object-oriented and spatial research communities; therefore, relevant research from each field is reviewed in Section 5.2. Section 5.3 introduces basic terminology and definitions relevant to PW

[1] The installed power of a site is derived from the installed power of the buildings on that site. The percentage green space of a site depends on the area used for buildings and other structures.

relationships. Section 5.4 describes a framework for describing *spatial PW relationships* based on spatial derivation and constraints between whole and part objects. This framework can then be used as a basis for defining specific constructs as needed for different applications. Adopting this approach, we then identify five spatial PW relationships of general utility in geographic applications from a range of examples, viz., *spatial part*, *spatial membership*, *spatial inclusion*, *spatial cover*, and *spatial equal*, and define modeling constructs based on these relationships. These spatial PW relationships and associated modeling constructs are described in Sections 5.5 and 5.6. A technique for supporting such relationships explicitly in a conceptual data model is described in more detail in Section 5.7, using STUML. Although STUML is used here for illustration, the principles described are general and could be applied to any object-oriented model. Finally, in Section 5.8, the results of the investigation described in this chapter are summarized.

5.2 Review of Part-Whole Relationships in the Literature

PW relationships such as aggregation have been well researched in the object-oriented context [Hend98, Hend99a, Hend99b, Kilo94, Mots99, Odell94, Saks98, Wins87]. This work is marked by continuing efforts to (*i*) define the semantics of such relationships, (*ii*) differentiate between various types of PW relationships, and (*iii*) resolve inconsistent, ambiguous, or even self-contradictory use of terminology in the literature and in object-oriented standards such as the Unified Modeling Language (UML). UML [Booch99, OMG99, Rumb99] includes an *aggregation* construct for modeling relationships with PW semantics and a *composition* construct for modeling specialized aggregations that do not allow part instances to be shared. However, UML's approach to partitioning PW relationships into different categories based on part shareability is not necessarily the best choice for describing spatial PW relationships. Furthermore, the inconsistent and self-contradictory definitions of aggregation and composition constructs in UML, which have been well-documented in [Hend99a, Hend99b], make it difficult to base the discussion or definition of spatial PW relationships on these UML constructs.

Of particular relevance for modeling spatial PW relationships, the proposals for the OPEN Modelling Language (OML) [Hend98] consider one type of spatial relationship explicitly, that of containment (i.e. inclusion) and define a graphical notation for this relationship within OML. However, the construct is described as a referential relationship rather than a spatial PW relationship, since attributes of the containing object are not derived from those of the contained objects.

In general, the object-oriented research community has not explicitly considered PW relationships in the context of spatial data and therefore their work does not adequately describe the common spatial PW relationships typical of spatial (and spatiotemporal) applications. For instance, the distinction between *component/object*, *portion/mass*, and *place/area* PW relationship categories listed in [Mots99, Wins87] is not relevant in the context of spatial objects. Characteristics of relationships relevant to spatial applications, such as the spatial derivation and constraint characteristics described earlier, are not considered. Consequently, the common classes of spatial PW relationships remain unexplored.

There has been some exploration of spatial PW relationships in the spatial community. In [Tryf99], spatial constraint relationships can be modeled but they are treated as referential rather than PW relationships, as with OML. Modeling derived spatial extents is not supported. Parent [Paren98, Paren99] defines a spatial aggregation construct and allows the user to specify inter-object constraints on geometry or lifespan. Examples of inclusion and other topological constraints are given to illustrate aggregation constraints that can be specified by the user. However, there are no formal definitions given and no semantic or notational differentiation between the different types of aggregation.

Extensions to Object Modeling Technique (OMT) for Geographic Information Systems (GIS) are defined in [Tryf97b] for spatial aggregation and grouping relationships, involving spatial derivation and spatial inclusion respectively. A spatial aggregation is formally defined as the relationship between a group of objects where the spatial extent of one spatial object is derived from the geometric union (GU) of the spatial extents of the other spatial objects in the group. The GU is the union of all points from a set of spatial extents. A spatial grouping is defined as a group of spatial objects each of whose spatial extents are of the same geometric type and are covered by (i.e. included in) the spatial extent of another spatial object.

However, the semantic distinction between spatial derivation and grouping is not clearly illustrated. For example, the first example given for spatial grouping, a network composed of segments, is likely to be a result of spatial derivation and thus necessarily covering. Another example of spatial grouping given is partitioning the area of a landparcel by soil type. This example does not involve spatial derivation. It does involve spatial cover since every soil type partition must be inside the land parcel. However, the definition of cover in [Tryf97b] does not fully describe the spatial relationship since it does not account for the additional requirement that the land parcel must be *completely* covered by soil type partitions, i.e. every point of the land parcel must be classified as being of some soil type. Furthermore, non-spatial characteristics of aggregation relationships such as transitivity, asymmetry, and separability are not considered.

5.3 Basic Terminology and Definitions

In this section, we consider the characteristics of relationships most relevant to the definition of PW relationships (and hence to spatial PW relationships) in that they can be used to differentiate such relationships from general associations[2] in the object-oriented model. Given the inconsistent or ambiguous references that can be found in the literature, the aim of this section is to precisely define such characteristics and the specific terminology adopted in this thesis as a basis for defining spatial PW relationships in Sections 5.4 through 5.6.

In order to avoid the confusion resulting from the inconsistent use and definition of phrases such as *part-of*, *composition*, and *aggregation*, we adopt the terminology introduced earlier for non-peer relationships: *PW relationship* for an asymmetric relationship between a *whole* object and *part* objects or *spatial PW relationship* when the whole and parts are *spatial objects*. These terms will be defined formally in terms of their fundamental characteristics in subsequent sections. Throughout the following discussion, the term *property* is used for application information or data that needs to be recorded (e.g. the purchase date of a building site), the term *attribute* is used to refer to a specific modeling construct representing such data (e.g. an attribute called *purchaseDate* used to represent the purchase date property), and the term *characteristic* is used to refer to constraints inherent to the definitions of specific modeling constructs (e.g. the asymmetry constraint used to define a PW relationship).

Characteristics relevant to the definition of PW relationships are summarized in Table 5.1 and then formally defined in this and subsequent sections.

---

[2] The terms *relationship* and *association* are used as synonyms to describe a semantic link between objects.

Table 5.1 Summary of Relationship or Association Characteristics

Type of Characteristic	Characteristic	Description of the Relationship (or Association) having that characteristic. (GU=geometric union)
Mathematical	transitivity	If an object has a relationship with a 2nd object that has the same relationship with a 3rd object, then the same relationship exists between the 1st and 3rd object.
	instance anti-symmetry	The same 2 object instances cannot participate in 2 different instances of the same relationship type with roles reversed.
	type anti-symmetry	The same 2 object types cannot participate in 2 different instances of the same relationship type with roles reversed.
	instance irreflexivity	An object cannot have a relationship with itself.
	type irreflexivity	The relationship cannot have an instance between two objects of the same type.
	instance asymmetry	The relationship is both instance anti-symmetric and instance irreflexive.
	type asymmetry	The relationship is both type anti-symmetric and type irreflexive.
	exclusivity	A single part object instance can participate in only 1 PW relationship instance.
Dependency-related	existence dependence	A part object instance cannot exist without or be separated from an associated whole object instance.
	inseparability	After being associated with a whole object instance, a part object instance cannot be removed from that association as long as it exists.
	essentiality	A part object must be associated with some whole object at all times or vice versa; however, transfers of part objects between associations can occur.
Structural	emergent property(s)	At least 1 property of the whole object is independent of (not derived from) the part objects' properties.
	resultant property(s)	At least 1 property of the whole object is dependent on (derived from) the part objects' properties.
	homeomerousity	At least 1 property is required to have identical values for all part objects (and possibly the whole object) in a PW relationship.
	propagation	The value of at least 1 emergent property of the whole object propagates to the part objects.
	configurational constraint	There are functional (e.g. logical order) or structural (e.g. topological) constraints on part objects in a PW relationship.
Spatial	spatial derivation	The spatial extent of the whole object is derived from those of its part objects (i.e. equivalent to the GU of the part objects' spatial extents).
	spatial constraint	The spatial extent of the whole object constrains those of its part objects, i.e. there is a topological constraint between the whole object's spatial extent and the GU of the part objects' spatial extents.
	spatial inclusion, cover, or equal	The GU of the part objects' spatial extents is less than or equal to, more than or equal to, or equal to the spatial extent of the whole object.

To understand the characteristics used to define and classify different types of PW relationships, we categorize them as being either: *mathematical*, *dependence-related*, *structural*, or *spatial*. The spatial characteristics used to classify spatial PW relationships are defined in Sections 5.4 through 5.6. Implementation characteristics such as reference versus value semantics and access characteristics are not relevant to the current discussion on conceptual data modeling but are considered in [Hend99a, Wins87]. The *mathematical, dependence-related,* and *structural* characteristics used to describe spatial PW relationships are defined as follows.

Mathematical characteristics of a relationship, i.e. constraints on legal sets of relationship instances in the database, include those of: *transitivity, anti-symmetry, irreflexivity, asymmetry,* and *exclusivity*. To define these characteristics, we assume that we have the following:

- object instances *a* and *a*' of object type[3] *A*, object instances *b* and *b*' of object type *B*, and object instances o_1, o_2, and o_3 of the same or different object types, where:
  - any two of *a, a', b, b'*, o_1, o_2, and o_3 could refer to the same or different object instances unless otherwise specified, and
  - A and B could refer to the same or different object types unless otherwise specified, and
- $a = a'$ means that *a* and *a*' are the same object instance, whereas $a \neq a'$ means that *a* and *a*' are not the same object instance, and
- A=B means that A and B are the same object type, and
- *a R b* means that instances *a* and *b* are related by a relationship of type *R.*

[3] The term *object type* is used synonymously with *object class* in this thesis to mean an extension defined by conformance to a given intension, as the distinction is not relevant to this thesis.

A relationship R is said to be *transitive* if and only if $(o_1\ R\ o_2) \wedge (o_2\ R\ o_3) \Rightarrow (o_1\ R\ o_3)$. Assume a relationship R that is transitive. Then if object instance o_1 has a relationship R with object instance o_2, and object instance o_2 also has the relationship R with object instance o_3, then object instance o_1 must have the same relationship R with object instance o_3. Transitive relationships are often characterized by hierarchies of associated object instances.

Symmetry and reflexivity are closely related characteristics and can be defined at either the object type or object instance level. Symmetry, its converse, asymmetry, and the related characteristic, anti-symmetry, involve two different object instances or object types; whereas, reflexivity and its converse, irreflexivity, involve only one object instance or object type.

If a relationship R is *instance anti-symmetric*, then $(o_1\ R\ o_2) \wedge (o_2\ R\ o_1) \Rightarrow o_1{=}o_2$. A relationship is said to be *type anti-symmetric* if and only if $(a\ R\ b) \wedge (b'\ R\ a') \Rightarrow A{=}B$. In other words, if two different objects (or object types) participate in an instance (or type) anti-symmetric relationship, then they cannot participate in another instance of the same relationship with their roles reversed. A relationship R is *instance irreflexive* if and only if $\neg \exists\ a\ (a\ R\ a)$. This means that a relationship instance between an object instance and itself is not legal. A relationship R is *type irreflexive* if and only if $\neg \exists\ a{,}a'\ (a\ R\ a')$. This means that a relationship instance between two objects of the same type is not legal.

A relationship R is said to be *asymmetric* if it is anti-symmetric and irreflexive. This can be defined as $(o_1\ R\ o_2) \Rightarrow \neg\ (o_2\ R\ o_1)$ for instances and $(a\ R\ b) \Rightarrow \neg \exists\ a'{,}b'\ (b'\ R\ a')$ for types. Assume a relationship R with instance asymmetry. Then if object instance o_1 has the relationship R with object instance o_2; then object instance o_2 cannot have the same relationship R with object instance o_1, where o_1, o_2 could be different instances (i.e. anti-symmetry) or the same instance (i.e. irreflexivity). Assume instead a relationship *R* with type asymmetry. Then if an instance of type *A* has relationship *R* with an instance of type *B*, no instance of type *B* can have the relationship *R* with an instance of type *A*; where *A* and *B* may be different types (i.e. type anti-symmetry) or the same type (i.e. type irreflexivity).

Exclusivity (sometimes called instance exclusivity) refers to whether an object instance can participate in more than one relationship instance. Because it is usually assumed that the participation of a whole object in PW relationships is not exclusive (i.e. may be shared by more than one part object), this characteristic is usually used to refer to the shareability of the part object between different whole objects. In general, exclusivity applies across different PW relationship types, i.e. a part object instance can never participate in more than one PW relationship instance. A less restrictive variant of exclusivity, where the constraint applies only within a single PW relationship type (i.e. a part object instance can participate in more than one PW relationship instance as long as they belong to different PW relationship types), can be specified using cardinality and so requires no special treatment.

Three *dependency-related* characteristics (constraints) of PW relationships are described in [Mots99]: *existence dependence* (i.e. connection mandatory and immutable), *essentiality* (i.e. connection mandatory but mutable), and/or *inseparability* (i.e. connection optional but immutable once established). *Existence dependence* implies that one object instance cannot exist without the other related object instance, i.e. its existence time must be contained in or equal to that of the related object. *Essentiality* implies that an object instance must be associated with some other object instance at all times; however, the specific association can change over time. Therefore, an object must be associated with another object on creation and must be either destroyed or transferred to another association if the first association is destroyed. Finally, *inseparability* implies that an association instance, once created, cannot be changed as long as the associated objects exist. Therefore, the objects can be created in any order; however, once associated, there are deletion dependencies. This implies that the destruction of one object, usually the whole object, results in the destruction of the associated or part objects.

In the context of PW relationships, existence dependence and inseparability usually refer to dependencies of parts on a whole object, i.e. a part cannot exist without an associated whole or a part cannot be separated from an associated whole after it is attached respectively. Essentiality can refer to either dependencies from the parts to the whole or vice versa.

Structural characteristics describe constraints on the properties of the whole and/or part objects in a PW relationship. The structural characteristics of a relationship include the following: *emergent, resultant*, *homeomerous, propagating,* and *configurational*. *Emergent* and *resultant* characteristics describe a relationship where one or more properties of the whole object are, respectively, *independent of* (i.e. not derived from) and *dependent on* (i.e. derived from) the parts' properties. In a *homeomerous* relationship, there is some property of the participating objects that is constrained to be identical (i.e. *homogeneous*) for all of the part instances and/or for the part and whole instances. For example, a relationship is homeomerous with respect to object type if the participating objects have identical types. A *propagating* relationship refers to the propagation of certain emergent property values of the whole object to the part objects. Finally, some authors [Hend99b, Kilo94, Mots99] have described a *configurational* characteristic as functional or structural constraints between part objects within a PW relationship, e.g. logical order or topological constraints on the part objects. Essentially, the configurational characteristic would be expressed as complex inter-object constraints on the legal values of a part object's attributes. Logical order would involve constraints on part sequence possibly expressed in terms of a *sequence-number* attribute on for each part object. Topological constraints limit the legal topological relationships allowed between the part's spatial-extents and are considered in depth in Chapter 6.

Any of the characteristics described above apply equally well in the context of spatial data. Mathematical and dependency-related characteristics of relationships are independent of the types of objects being related. As far as structural characteristics, the spatial properties of the whole object could be emergent (i.e. the whole's spatial extent is independent of the parts' spatial extents) or resultant (i.e. the whole's spatial extent is derived from the parts' spatial extents using set-based geometric operators such as geometric union). Homeomerous and configurational constraints could relate to spatial properties of the parts and/or the whole object. Spatial extents are of a certain geometric type and dimension, e.g. 2D line, 3D cube, and have geometric (i.e. measurement), topological (i.e. constant under rubber-sheet transformations), and directional (i.e. orientation) properties. Spatial PW relationships may involve homeomerousity constraints on geometric type and homeomerousity or configurational constraints on any of the spatial properties of an object (i.e. geometry, topology, or orientation).

In order to define a given modeling construct based on these characteristics, it is important to distinguish between the characteristics that are primary (i.e. defining and essential), derived (i.e. consequent on the primary characteristics), or secondary (i.e. varying and non-essential) for a given category of spatial PW relationships. Common or generally useful secondary characteristics can be identified in order to serve as a guideline to designers and as an aid in identifying and specifying variants of the base spatial PW relationship categories.

5.4 Spatial Part-Whole Relationships

Having precisely defined relevant characteristics, a formal definition of spatial PW relationships is possible, using the application examples described in Section 5.1 and UML notation for illustration purposes. The relevant UML diagram, the Class diagram, captures the static structure of a database design. As described earlier in Section 3.2, the diagram consists of descriptions of object classes (i.e. class name, attributes, and operations) inter-connected through *generalization* (i.e. sub-classes defined based on a super-class) or *association* (i.e. semantic relationships between object classes), including the specialized forms of association described earlier, i.e. *aggregation and composition*.

Spatial PW relationships are denoted in the diagrams using a new type of association construct with a circle at one end (i.e. analogous to UML's aggregation and composition constructs), where cardinality is indicated using the symbols *1* and *** to mean *one* and *many* respectively. The abbreviation inside the circle shows which type of spatial PW relationship is being modeled. Figure 5.1 shows the class relationship notation used relevant to the current discussion on PW relationships, with the addition of the spatial PW relationship.

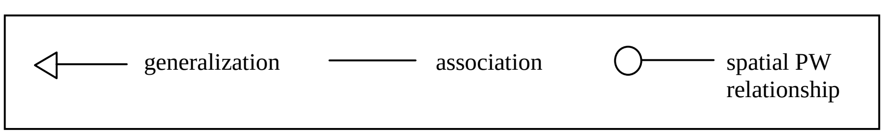


Figure 5.1 Class Relationships

The characteristics that are fundamental to the characterization of spatial PW relationships include at the minimum some degree of abstraction and asymmetry.

The role of abstraction is essentially that of information hiding to reduce complexity, where the details of the individual part objects are not relevant (and so can be temporarily hidden) and are, in some respect, represented semantically by the abstraction. Abstraction further implies that the whole object is more than just the sum of its part objects. At the minimum, the whole object has its own identifier in order to be modeled as an object separate from those of its parts. Emergent and resultant characteristics can be used to model the combination of independent and derived semantics that characterizes abstraction. Asymmetry can be formally defined using the anti-symmetry and irreflexivity characteristics described in the previous section. We therefore define a *spatial PW relationship* between a whole object and a finite number of part objects where:

- each object (whether whole or part) is a *spatial object*, i.e. has an associated spatial extent, and
- the whole object has at least one emergent and one resultant attribute, other than the object identifier, and
- the relationship has instance asymmetry (i.e. instance anti-symmetry and instance irreflexivity).

Based on this definition, we then distinguish between two fundamentally different categories of spatial PW relationships, *spatial derivation relationships* and *spatial constraint relationships*. The difference between these two categories is illustrated by the supranational organization and building site examples respectively. The fundamental characteristic used to categorize spatial PW relationships is whether the spatial extent of the whole is derived from or constrains that of its parts. Both are structural characteristics of a relationship, i.e. resultant and configurational (topological) respectively. This distinction is especially important in the context of spatial PW relationships in that it determines whether the spatial extent of the whole depends on those of its parts or vice versa.

Within the framework of spatial derivation and constraint relationships, five specific types of spatial PW relationships of general utility in geographic applications are identified from a range of examples[4]: *spatial part*, *spatial membership*, *spatial inclusion*, *spatial cover*, and *spatial equal.* Figure 5.2 illustrates how the definitions of these relationships are related in terms of their primary and derived characteristics, where a nested box indicates a more specific category of relationship. The abbreviation used to model each type of spatial PW relationship is also indicated. An overview of the relationships and their defining characteristics are presented next: individual categories are examined in detail in Sections 5.5 and 5.6.

The first two cases, *spatial part* and *spatial membership*, are examples of a *spatial derivation relationship*. Specifically, the whole object's spatial extent is equivalent to and directly derived from the GU of the part objects' spatial extents (called a *part-union*, both *GU* and *part-union* are defined formally in Section 5.5). Examples are the supranational organization, national land transport network, and road network examples discussed earlier. For instance, the spatial extent of the European Union depends on and is derived from that of its member countries. Similarly, the spatial extent of a national land transport network is derived from those of its component road, bus, and rail networks.

Although these relationships are all examples of spatial derivation, they have distinct differences in semantic interpretations and characteristics. The relationships between a national land transport network, a road network, and roads are transitive.

[4] Specification of additional variants or modeling constructs based on these categories, relevant in the context of a specific geographic application, are illustrated in Chapter 6.

That is, if a road is part of the road network, and that road network is part of a national land transport network, then the road must also be part of the national land transport network. This is an example of a *spatial part* relationship. *Spatial part* represents an assembly whose transitive semantics typically lead to the formation of nested hierarchies.

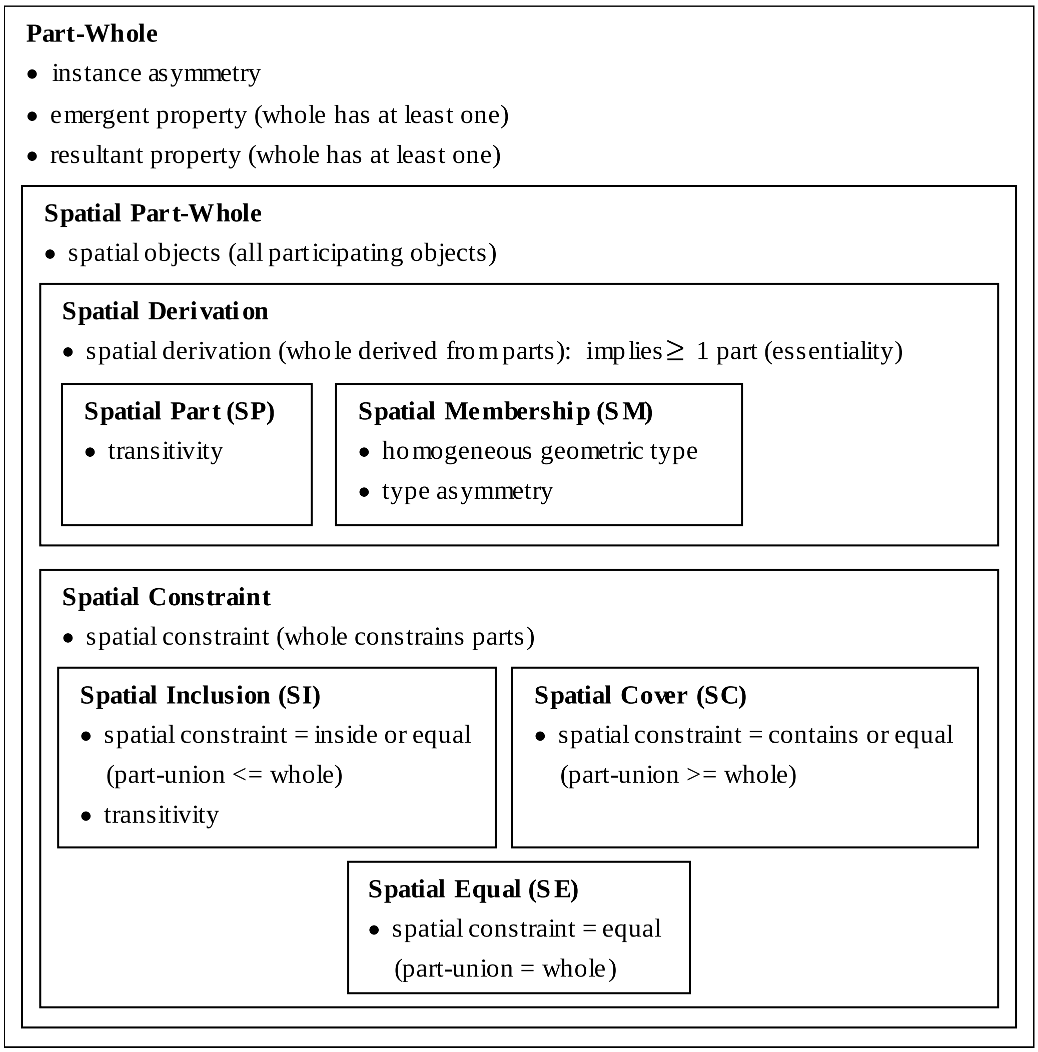


Figure 5.2 Spatial Part-Whole Relationships
(Primary & Derived Characteristics)

In contrast, *spatial membership* represents a grouping of objects which are homeomerous with respect to geometric type, which belong to (i.e. are members of) a collection or organization, and whose relationships do not exhibit transitivity. For example, all of the countries which are members of EU are associated with a bounded 2D area. Countries may be further subdivided into political units such as states; however, these political units are not considered members of the EU.

In a *spatial constraint relationship,* the whole and part objects each have separately defined spatial extents, i.e. the spatial extent of the whole cannot be derived from those of its parts. However, the spatial extent of the whole constrains the GU of the parts' spatial extents (i.e. the part-union) and thus the legal spatial extents of the parts. This is an example of a type of spatial PW relationship distinguished by a specific configurational characteristic describing topological constraints on the relationship between the spatial extent of the whole and the part-union.

Spatial inclusion is one specific type of *spatial constraint relationship,* where the constraint is one of inclusion. This can be illustrated by the case of a building-site and the buildings or structures located on that site. That is, the GU of the part's spatial extents (e.g. the structure's ground areas) must be less than or equal to that of the whole's spatial extent (e.g. the building site area). Or, equivalently, the area of the whole spatial object (the building site) *contains or is equal to* the combined (or individual) spatial extents of the part objects (the sites' buildings and structures).

Spatial cover, where the GU of the part's spatial extents is greater than or equal to the whole's spatial extent, is illustrated by the requirement that a combined range of a set of phone service cells completely cover a guaranteed mobile phone service area. Finally, the division of an administrative area into land use zones (e.g. residential, industrial, agricultural, recreational) demonstrates *spatial equal,* where the GU of the parts (the land use zones) must be exactly equal to the whole's spatial extent (the administrative area).

Note the distinction between the *spatial derivation* relationships and the *spatial equal* relationship. In both cases, the combined spatial extent of the parts is equal to that of the whole. In the case of *spatial derivation* relationships, this is because the spatial extent of the whole is directly *derived from* those of its parts (and cannot exist independently of its parts). In the case of the *spatial equal* relationship, this is because the spatial extent of the whole *constrains* those of its parts (but exists independently of that of its parts).

The distinctions between the two types of relationships can be further elaborated. As a consequence of the dependence of the whole's spatial extent on the part's spatial extents, at least one part must exist for the spatial extent of the whole to be defined in a spatial derivation relationship. Thus, the area of the European Union is not defined independently of the area of its member countries. In contrast, the spatial extent of the whole is defined prior to and independently of those of its parts for a spatial equal relationship.. This type of relationship typically involves a sub-division of an existing region into parts, where the combined spatial extents of the resulting parts are *constrained* to equal that of the whole For example, an existing administrative region can be divided into land use zones; however, the dimensions of its spatial extent are determined before (and independently of) those of its zones.

These relationships are explained in detail in Sections 5.5 and 5.6. Spatial derivation relationships are defined formally in Section 5.5 and spatial constraint relationships in Section 5.6.

5.5 Spatial Derivation Relationships

A *spatial derivation relationship* is a spatial PW relationship between a set of part objects and a whole object whose spatial extent represents the GU of the part objects' spatial extents. The GU function has the following signature and semantics. The signature is:

$$f_{\text{geometric-union}}\text{: } 2^{SE} \text{–> SE where:}$$

- SE is the domain of spatial extents and 2^{SE} is the power set for the domain, i.e. the collection of all subsets of SE, and
- the definition of GU function, adapted from the definition given in [Tryf97b], is as follows:

  Let p be a point in space. Let $s_1,\ldots,s_i,\ldots,s_n$ be a finite, non-empty (n>0) set of spatial extents and $s_{\text{geometric-union}}$ be the spatial extent resulting from the GU of $s_1,\ldots,s_i,\ldots,s_n$. Then

$$f_{\text{geometric-union}}(s_1,\ldots,s_i,\ldots,s_n)$$

$$= s_{\text{geometric-union}}$$

$$= \{p \mid \exists i\ (p \in s_i) \wedge \forall i\ (p \in s_i \Rightarrow p \in s_{\text{geometric-union}})\}$$

If the n spatial extents above represent the spatial extents associated with a set of n part objects in a spatial PW relationship, then the result of the GU of their associated spatial extents, $s_{\text{geometric-union}}$, is called the *part-union* for that spatial PW relationship. Therefore, a spatial PW relationship between a whole object o_{whole} with spatial extent s_{whole} and n part objects $o_1,\ldots,o_i,\ldots,o_n$ with spatial extents $s_1,\ldots,s_i,\ldots,s_n$ is called a spatial derivation relationship if and only if $s_{\text{whole}} = f_{\text{geometric-union}}(s_1,\ldots,s_i,\ldots,s_n)$ when n>0 and s_{whole} is undefined otherwise (when n=0).

Therefore, *spatial derivation relationships* have the characteristics of *spatial PW relationships* (i.e. the *instance asymmetry*, *emergent property*(*s*), and *resultant property*(*s*) of a PW relationship and the *spatial objects* of a spatial PW relationship) and the additional characteristic of a *derived spatial extent for the whole object* as described above. Note that spatial derivation of the whole's spatial extent actually implies that the relationship has instance asymmetry (since instance symmetry or reflexivity would make derivation impossible) and that the whole object has at least one resultant property (its spatial extent). However, these characteristics are already assumed as primary characteristics since a spatial derivation relationship is also a PW relationship. Further consequent on the characteristic of spatial derivation, we can derive the following additional constraint:

- *>= 1 part:* The whole object cannot exist without at least one part object (i.e. essentiality of whole on part), otherwise its spatial extent would be undefined.

The specific sub-categories of spatial derivation relationships, *spatial part* and *spatial membership* relationships, are described in Sections 5.5.1 and 5.5.2 respectively. Secondary characteristics for spatial derivation relationships, which lead to variants of the basic spatial part and spatial membership relationships, are described in Section 5.5.3.

5.5.1 Spatial Part Relationships: Primary Characteristics

The primary characteristics of *spatial part* are described in detail in this sub-section. The national land transport example discussed in Section 5.1 and shown in Figure 5.3 is used to illustrate the characteristics of spatial part relationships. Two alternative representations are shown. Figure 5.3 (a) uses generalization to define the different types of transport networks and to illustrate the use of spatial part relationships with generalization, e.g. each different type of transport network inherits the associated spatial part relationships. Alternatively, in Figure 5.3 (b), each type of transport network is defined directly as a separate part object type with its own spatial part relationship to the network segments. Although less compact, this approach more clearly shows that a single national land transport network can be composed of more than one road network, bus network, and/or rail network and allows the clear specification of the specific type of network segment applicable to each type of transport network.

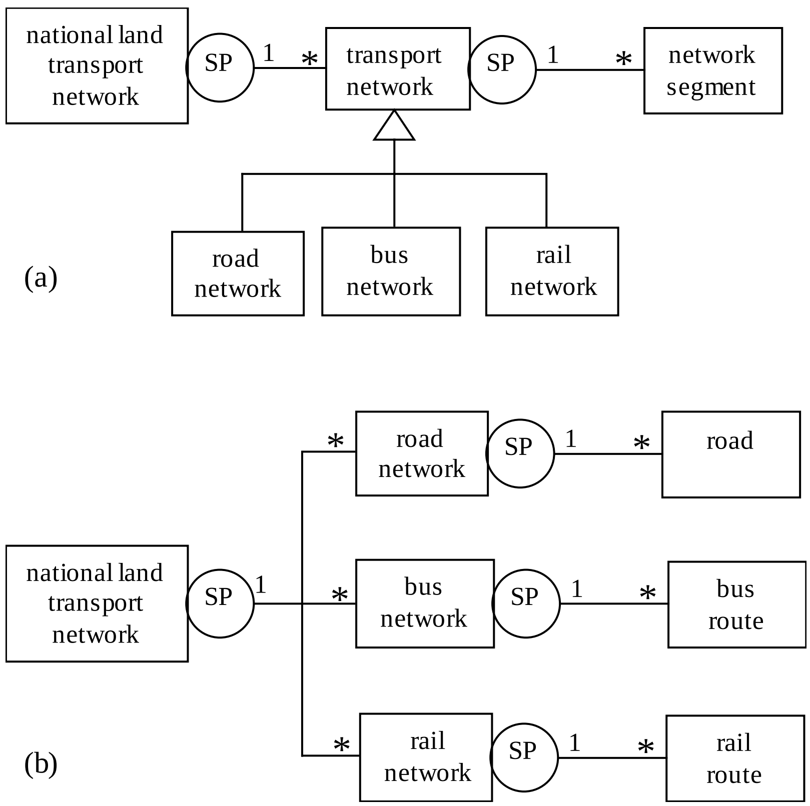


Figure 5.3 Spatial Part Relationship

As illustrated in Figure 5.2, the primary characteristics of a spatial part relationship include those of its "parent" categories as well as those characteristics specific to spatial part relationships. Because a spatial part relationship is a PW relationship, it has *instance asymmetry*, at least one *emergent property*, and at least one *resultant property*. A spatial part relationship consists of *spatial objects* because it is a spatial PW relationship and has *spatial derivation* because it is a spatial derivation relationship. Finally, a spatial part relationship is further characterized by its *transitivity*.

These characteristics can be illustrated using the national land transport example. All of the objects in the example, the national land transport network, individual transport networks, and individual network segments, have associated spatial extents and so are spatial objects. The spatial extent of the national transport network is derived from those of its constituent transport networks, which are in turn derived from those of its constituent network segments. As discussed in Section 5.5, this derivation implies both instance asymmetry and that the whole object has at least one resultant property (the spatial extents of the national transport network and individual transport networks); since it would be impossible to derive spatial extents if there was instance symmetry or reflexivity. For example, the hypothetical cases of a given transport network being part of itself (instance reflexivity) or part of a national transport network which is then part of the same transport network (instance symmetry) are not logical and would make it impossible to derive the spatial extent of the transport network. An emergent property can be illustrated by a *headquarters* attribute representing the headquarters of the national land transport network (not derivable from attributes of its individual transport networks). Transitivity can be demonstrated as follows. If a given network segment is part of a transport network and that transport network is part of a national land transport network, then the same network segment also forms part of the national land transport network.

Another spatial derivation relationship, spatial membership, is described in the next section. Because it has the same parent categories as the spatial part relationship (as illustrated in Figure 5.2), it shares the same characteristics except for the one characteristic specific to the spatial part relationship (i.e. transitivity).

5.5.2 Spatial Membership Relationships: Primary Characteristics

We now describe in detail the primary characteristics of *spatial membership*, using the example of supranational organizations, such as the EU or NATO, having member countries, that was introduced in Section 5.1 and is illustrated in Figure 5.4. We also show a member country having a spatial part relationship with its constituent states to illustrate the combination of different types of spatial PW relationships in a single diagram. The assumption here is that a country is formed from a union of its states, as with the United States, where the states have areas defined prior to and independently of their identification with a country but the reverse does not hold. Thus the spatial extent of a country is derived from that of its states. Analogously, a state could be considered to be formed from a union of its counties, leading to a transitive spatial part hierarchy.

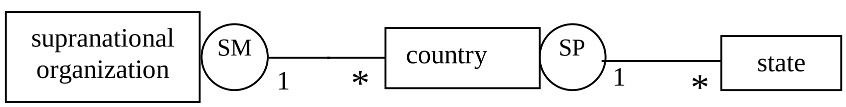


Figure 5.4 Spatial Membership Relationship

As illustrated in Figure 5.2, the primary characteristics of a spatial membership relationship include those of its “parent” categories as well as those characteristics specific to spatial membership relationships. Because a spatial membership relationship is a PW relationship, it has *instance asymmetry*, at least one *emergent property*, and at least one *resultant property*. A spatial membership relationship consists of *spatial objects* because it is a spatial PW relationship and has *spatial derivation* because it is a spatial derivation relationship. Finally, a spatial membership relationship is further characterized by its *type asymmetry* and the *homogeneous geometric type* of its part objects. The last characteristic means that the spatial extents of all part objects in a spatial membership relationship belong to the same base geometric type (e.g. points, lines, polygons, volumes).

These characteristics can be illustrated using the supranational organization example and the EU as a specific instance of a supranational organization. All of the objects in the example, the supranational organization and its member countries, have associated spatial extents and so are spatial objects. The spatial extent of a supranational organization such as the EU is derived from those of its constituent member countries. As discussed in Section 5.5, this derivation implies both instance asymmetry and that the whole object has at least one resultant property (the supranational organization's spatial extent); since it would be impossible to derive spatial extents if there was instance symmetry or reflexivity. For example, the hypothetical cases of the EU being part of itself (instance reflexivity) or a country such as France being part of the EU that is, in turn, a part of France (instance symmetry) are not logical and would make it impossible to derive the spatial extent of the EU. An emergent property can be illustrated by a *headquarters* attribute representing the headquarters of a supranational organization such as the EU (not derivable from attributes of its member countries). Other attributes of the EU, such as budget, may be derivable from the contributions of its members. However, the contributions are not, strictly speaking, attributes of the members themselves, but of their association with the EU.

The characteristics specific to spatial membership can also be illustrated using the EU example. Part objects with homogeneous geometric type are illustrated by the (possibly irregular) bounded 2D area associated with all EU member countries. The characteristic of type asymmetry can be illustrated by considering examples of its converse: type symmetry or type reflexivity. Type symmetry would imply that a supranational organization could be a member of a country (since a country can be a member of a supranational organization). Type reflexivity would imply that a supranational organization could be a member of another supranational organization. Although the first example is clearly not logical, the latter seems possible. However, a closer examination shows that the ambiguity of natural language hides the fact that semantically we have two different object types. A supranational organization has member countries, whereas a “mega” organization’s members are other organizations. Since a *mega organization* cannot have a country as a member, it cannot participate in the same types of relationships as the *supranational organization*. Therefore, although they may share the same ancestor object type *organization*, *supranational organization* and *mega organization* must be separate object types.

In the next section, common secondary characteristics of spatial derivation relationships are identified.

5.5.3 Spatial Derivation Relationships: Secondary Characteristics

In this section, we describe secondary characteristics of spatial derivation relationships. These characteristics are not essential or defining characteristics of spatial derivation relationships or their sub-categories, but instead one or more characteristics may be used to specify additional constraints on spatial derivation relationships that lead to variants of the basic *spatial part* or *spatial membership* relationship categories. Those secondary characteristics that apply only to spatial part relationships are indicated by an asterisk; otherwise, they apply equally to spatial part and membership relationships. The list of secondary characteristics is not intended to be exhaustive, but instead provides a set of generally useful secondary characteristics that can serve as a practical guideline to designers and as an aid in identifying and specifying variants of the base spatial derivation categories.

- *Set-based spatial constraints on parts* (*topological, orientation, geometric*): This includes topological configurational constraints such as requiring non-overlapping parts or equal parts and geometric constraints such as requiring parts with the same geometric type* (this refers only to spatial part, since it is a primary characteristic for spatial membership) or geometric property (i.e. shape, size, orientation, position).
- *Non-spatial constraints:*
    - *Homogeneous object type*: All objects are of the same object type. This refers only to part objects for spatial membership, since the type asymmetry characteristic mandates that the whole and part objects cannot have the same object type.
    - *Exclusivity*: A part object instance cannot be shared between different whole objects, i.e. be in a spatial PW relationship with more than one whole object.
    - *Inseparability of Part from Whole*: A part object instance cannot be disconnected from a given whole object instance once connected. This implies that a part object may be created before or after the whole object, but must be destroyed with the whole object.
    - *Existence Dependence of Part on Whole*: A part instance cannot exist separately from some whole instance. This implies that the part instance must be connected to some whole object instance when created and either destroyed or transferred to some other whole object instance if the first is destroyed.

- *Essentiality of Part on Whole*: A part must be connected to some whole instance but the connection can be changed.

Having considered the definition, sub-categories, and secondary characteristics of spatial derivation relationships, we do the same for spatial constraint relationships in the next section.

5.6 Spatial Constraint Relationships

A *spatial constraint relationship* is a spatial PW relationship between a whole object and a set of part objects where there is a specific topological relationship between the whole object's spatial extent and the GU of the part objects' spatial extents (i.e. the *part-union*). In other words, the spatial extent of the whole object constrains the part-union based on some topological constraint. The terms *GU* and *part-union* have the same definitions as that given for spatial derivation relationships in Section 5.5.

In contrast to spatial derivation relationships, where essentiality of the whole on parts is consequent on the derivation of the whole's spatial extent; a spatial constraint relationship does not depend on the existence of parts. That is, the spatial extent of the whole object is defined even when there are no part objects. For example, a building site area, guaranteed phone coverage area, and administrative area are defined even without any buildings, phone service cells, or land-use zones respectively.

Note that a spatial constraint between objects represents a spatial constraint relationship only if it is associated with the other characteristics of a spatial PW relationship (i.e. the *instance asymmetry*, *emergent property*(*s*), and *resultant property*(*s*) of a PW relationship and the *spatial objects* of a spatial PW relationship).

As illustrated in Figure 5.2, the specific topological constraint involved in a spatial constraint relationship determines the spatial constraint sub-category. The three sub-categories shown in Figure 5.2, *spatial inclusion, spatial cover,* and *spatial equal* relationships, are described in Sections 5.6.1, 5.6.2, and 5.6.3 respectively. The distinguishing topological constraint for each specific sub-category of spatial constraint relationship is described in terms of points belonging to the whole and part spatial extents. Secondary characteristics for spatial constraint relationships, which lead to variants of the basic spatial inclusion, spatial cover, and spatial equal relationships, are described in Section 5.6.4. A more general method of describing spatial constraint relationships is discussed in Chapter 6 in the context of providing a classification scheme for topological relationships useful for conceptual modeling.

5.6.1 Spatial Inclusion Relationships: Primary Characteristics

In the case of the spatial inclusion relationship, the topological constraint is one of inclusion (i.e. containment). The spatial inclusion relationship was described in Section 5.4 as being a relationship between a set of part objects, each having a spatial extent, and a whole object whose spatial extent contains or equals (i.e. includes) the GU of the parts' spatial extents. In this case, the spatial constraint relationship can be equivalently and more simply stated in terms of the spatial extent of each individual part object, i.e. the spatial extent of each part object is contained by or equal to (i.e. is included in) the spatial extent of the whole object. We can state this mathematically as follows.

Let *p* be a point in space. Let s_{whole} be the spatial extent of a whole object and $s_1,\ldots,s_i,\ldots,s_n$ the spatial extents of the part objects in a spatial PW relationship. Then saying that there is a spatial inclusion relationship between the part objects and the whole object means that:

$$\forall i \ (p \in s_i \Rightarrow p \in s_{whole})$$

This means that every point in any part object's spatial extent must also belong to (i.e. be included in) the whole object's spatial extent. However, note that the reverse may not be true, i.e. there may be points in the whole object's spatial extent that do not belong to any part object's spatial extent. The spatial inclusion constraint is further characterized by its transitivity, since the inclusion constraint between the whole object's spatial extent and the GU of the part objects' spatial extents can be equivalently expressed as a constraint between the whole and each part individually.

The primary characteristics of *spatial inclusion* can then be described in detail. The building site example (with its buildings and structures) discussed in Section 5.1 and shown in Figure 5.5 is used to illustrate the characteristics of spatial inclusion relationships.

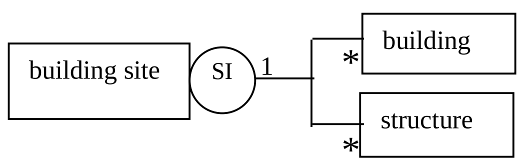


Figure 5.5 Spatial Inclusion Relationship

As illustrated in Figure 5.2, the primary characteristics of a spatial inclusion relationship include those of its "parent" categories as well as those characteristics specific to spatial inclusion relationships. Because a spatial inclusion relationship is a PW relationship, it has *instance asymmetry*, at least one *emergent property*, and at least one *resultant property*. A spatial inclusion relationship consists of *spatial objects* because it is a spatial PW relationship and has a *spatial constraint* because it is a spatial constraint relationship. The specific spatial constraint is that of inclusion, as defined above. This topological constraint is further characterized by its transitivity, as explained above. Therefore, the characteristics specific to spatial inclusion relationships are the topological constraint of *spatial inclusion* (between the whole object's spatial extent and the GU of the parts) and *transitivity*.

These characteristics can be illustrated using the building site example. All of the objects in the example, the building site, buildings, and structures, have associated spatial extents and so are spatial objects. The characteristic of instance asymmetry can be illustrated by considering examples of its converse: instance symmetry or instance reflexivity. That is, it does not make sense for a building site to contain itself (instance reflexivity), or to contain a building that in turn contains the same site (instance symmetry). (The characteristic of instance asymmetry can also be derived from the inclusion constraint, since the definition of containment precludes a spatial extent containing itself.) The original purchase price of the building site, the date the site was purchased, and the spatial extent of the building site cannot be derived from the price of its buildings and structures, the dates they were built, or their spatial extents; therefore, these represent examples of emergent properties. However, the installed power and green space of a building site are derived from attributes of its buildings and structures, as discussed in Section 5.1. Therefore, they demonstrate resultant properties. No structure or building on a building site can extend beyond the confines of that building site. So the spatial extent of the structure or building must be included in that of its building site. Therefore, there is an inclusion constraint between a building site and any of its structures or buildings. Transitivity can be illustrated using the building site example as follows. If a building contains an auditorium, and that building is located on the building site, then the auditorium is contained by the building site.

Another spatial constraint relationship, spatial cover, is described in the next section. Because it has the same parent categories as the spatial inclusion relationship (as illustrated in Figure 5.2), it shares the same characteristics except for the characteristics specific to the spatial inclusion relationship (i.e. the inclusion constraint and transitivity).

5.6.2 Spatial Cover Relationships: Primary Characteristics

In the spatial cover relationship, the topological relationship between the whole object's spatial extent and the GU of the parts' spatial extents is the reverse of that in spatial inclusion. The GU of the parts' spatial extents contains or equals that of the whole. We can state this mathematically as follows.

Let p be a point in space. Let s_{whole} be the spatial extent of a whole object and $s_1,\ldots,s_i,\ldots,s_n$ the spatial extents of the part objects in a spatial PW relationship. Then saying that there is a spatial cover relationship between the part objects and the whole object means that:

$$p \in s_{whole} \Rightarrow \exists\, i\ (p \in s_i)$$

This means that every point in the whole object's spatial extent must also belong to one of the part object's spatial extents. However, note that the reverse may not be true, i.e. there may be points in one or more part objects' spatial extents that do not belong to the whole object's spatial extent. In contrast to the spatial inclusion relationship, the spatial cover relationship is not transitive since it cannot be equivalently expressed in terms of the relationship between the spatial extent of each individual part and the whole. A given part's spatial extent may be disjoint from, overlapping, included in, containing, or equal to that of the whole. That is, no statement can be made regarding the topological relationship between the spatial extent of any individual part with that of the whole object.

The primary characteristics of *spatial cover* can then be described in detail using the mobile phone example discussed in Section 5.1 and shown in Figure 5.6. This illustrates the case where the combined range of a set of mobile phone service cells is required to completely cover an specific area with guaranteed mobile phone coverage. The spatial extent associated with each phone service cell represents its service range.

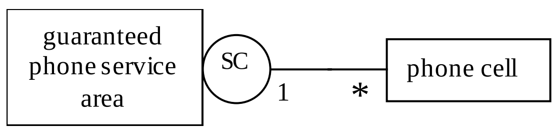


Figure 5.6 Spatial Cover Relationship

As illustrated in Figure 5.2, the primary characteristics of a spatial cover relationship include those of its "parent" categories as well as those characteristics specific to spatial cover relationships. Because a spatial cover relationship is a PW relationship, it has *instance asymmetry*, at least one *emergent property*, and at least one *resultant property*. A spatial cover relationship consists of *spatial objects* because it is a spatial PW relationship and has a *spatial constraint* because it is a spatial constraint relationship. The specific spatial constraint is that of *spatial cover* (between the GU of the part object's spatial extents and the whole object's spatial extent), as defined above.

These characteristics can be illustrated using the mobile phone example. All of the objects in the example, the guaranteed mobile phone coverage area and phone service cells, have associated spatial extents and so are spatial objects. The area of the guaranteed mobile phone service area cannot be derived from that of its phone cell ranges; therefore, this illustrates an emergent property. In contrast, the type of phone coverage (analog or digital) in the guaranteed mobile phone service area is derived from the type of phone coverage offered by its phone cells. This illustrates a resultant property. The GU of the phone service cell ranges contains or equals the guaranteed mobile phone service area, demonstrating a spatial cover constraint. The characteristic of instance asymmetry can be derived from the spatial cover constraint, since the definition of containment precludes a spatial extent containing itself. Obviously, it does not make sense for a guaranteed mobile phone service area to contain itself, or to contain a phone service cell that in turn contains the same guaranteed mobile phone service area.

Another spatial constraint relationship, spatial equal, is described in the next section. Because it has the same parent categories as the spatial inclusion and cover relationships (as illustrated in Figure 5.2), it shares the same characteristics except for the characteristics specific to those relationships (i.e. the inclusion constraint and transitivity for the spatial inclusion relationship and the cover constraint for the spatial cover relationship).

5.6.3 Spatial Equal Relationships: Primary Characteristics

In the spatial equal relationship, the GU of the parts' spatial extents equals that of the whole. We state this mathematically as follows.

Let p be a point in space. Let s_{whole} be the spatial extent of a whole object and $s_1,\ldots,s_i,\ldots,s_n$ the spatial extents of the part objects in a spatial PW relationship. Then saying that there is a spatial equal relationship between the part objects and the whole object means that:

$$\forall i \; (p \in s_i \Rightarrow p \in s_{whole}) \wedge (p \in s_{whole} \Rightarrow \exists\, i \, (p \in s_i))$$

This means that every point in any part objects' spatial extent must belong to the whole object's spatial extent and that every point in the whole object's spatial extent must belong to one of the part object's spatial extents. In other words, the spatial extent of the whole and the GU of the part object's spatial extents consist of exactly the same set of points. As with the spatial cover relationship, the spatial equal relationship is not transitive since it cannot be equivalently expressed in terms of the relationship between the spatial extent of each individual part and the whole.

We define the primary characteristics of *spatial equal* as follows, using the administrative region example introduced in Section 5.1 and shown in Figure 5.7, where the spatial extent of the administrative region must be the same as the GU of the administrative region's land-use zones (i.e. sub-divisions).

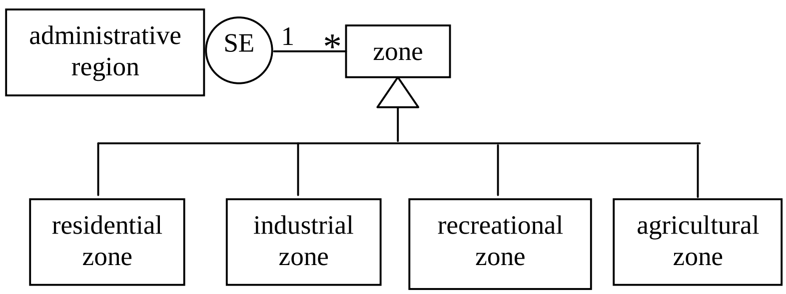


Figure 5.7 Spatial Equal Relationship

As illustrated in Figure 5.2, the primary characteristics of a spatial equal relationship include those of its "parent" categories as well as those characteristics specific to spatial equal relationships. Because a spatial equal relationship is a PW relationship, it has *instance asymmetry*, at least one *emergent property*, and at least one *resultant property*. A spatial equal relationship consists of *spatial objects* because it is a spatial PW relationship and has a *spatial constraint* because it is a spatial constraint relationship. The specific spatial constraint is that of *spatial equal* (between the GU of the part object's spatial extents and the whole object's spatial extent), as defined above.

These characteristics can be illustrated using the land-use zone example. All of the objects in the example, the administrative region and land-use zones, have associated spatial extents and so are spatial objects. The administrative region has emergent properties such as administrator and budget that cannot be derived from the properties of its land-use zones. The administrative region has resultant properties such as average allowable noise level, total resident population, or total working population that can be derived from properties of its land-use zones (assuming each zone has an allowable noise level, resident population, and transient working population representing employment figures specified). The spatial equal constraint is illustrated by the requirement that the spatial extent of the administrative region equal that of the GU of its land-use zones' spatial extents.

Note that the topological relationship between spatial extents in a spatial equal relationship may be instance reflexive (e.g. the area of an administrative region is equal to itself) or symmetric (e.g. for an administrative region consisting of only one land-use zone, the area of the region is equal to that of the zone and vice versa). However, when considering the full semantics of the spatial equal relationship (i.e. abstraction semantics with emergent and resultant properties), the spatial equal PW relationship is not instance symmetric or reflexive. It does not make sense to consider an administrative region part of itself or part of one of its own zones. This illustrates the instance asymmetry of the spatial equal relationship.

In the next section, common secondary characteristics of spatial constraint relationships are identified.

5.6.4 Spatial Constraint Relationships: Secondary Characteristics

In this section, we describe secondary characteristics of spatial constraint relationships. These characteristics are not essential or defining characteristics of spatial constraint relationships or their sub-categories, but instead one or more characteristics may be used to specify additional constraints on spatial constraint relationships that lead to variants of the basic *spatial inclusion, spatial cover,* or *spatial equal* relationship categories. The list of secondary characteristics is not intended to be exhaustive, but instead provides a set of generally useful secondary characteristics that can serve as a practical guideline to designers and as an aid in identifying and specifying variants of the base spatial constraint categories.

- *Set-based spatial constraints* (*topological, orientation, geometric*): This includes topological configurational constraints between parts such as requiring non-overlapping parts or equal parts and geometric constraints such as requiring all objects in the spatial PW relationship to have the same geometric type or geometric property (i.e. shape, size, orientation, position).
- *Non-spatial constraints:*
  - *Homogeneous object type*: All objects are of the same object type.
  - *Exclusivity* (*for spatial cover or spatial equal relationships only*): A part object instance cannot be shared between different whole objects, i.e. be in a spatial PW relationship with more than one whole object.
  - *Inseparability of Part from Whole*: A part object instance cannot be disconnected from a given whole object instance once connected. This implies that a part object may be created before or after the whole object, but must be destroyed with the whole object.

- *Existence Dependence of Part on Whole*: A part instance cannot exist separately from some whole instance. This implies that the part instance must be connected to some whole object instance when created and either destroyed or transferred to some other whole object instance if the first is destroyed.
- *Essentiality of Part on Whole or Whole on Part*: A part must be connected to some whole instance but the connection can be changed or vice versa.

Note that these secondary characteristics are basically the same as for spatial derivation relationships with the following exceptions. Exclusivity cannot be a secondary characteristic for spatial inclusion relationships, since transitivity in this case also implies inclusion by more than one object. Furthermore, homogeneous geometric type and geometric property or essentiality can also be applied to the whole object. (The homogeneity constraints cannot be applied to the whole object when the whole's spatial extent is derived from that of the parts. The essentiality of the whole on parts is consequent on spatial derivation, as explained in Section 5.5.)

To illustrate how the specification of one or more secondary characteristics can lead to variants of the basic spatial PW categories shown in Figure 5.2, consider an application involving a spatial equal relationship where all the part objects are further constrained to have the same spatial extent (i.e. dimensions, orientation, and position) as the whole object. Such a case could arise if a thematic attribute that varies over space is instead modeled as an object, i.e. as a spatial overlay. This approach might be warranted for thematic attributes which have a complex domain or when there are several such attributes with inter-relationships that must be modeled as objects with associations. An example, shown in Figure 5.8, would be a mountain with vegetation, hydrography, and elevation, where each is modeled as a spatial object over the same area. Such a case can best be modeled using secondary characteristics to specify the additional topological constraints between the parts.

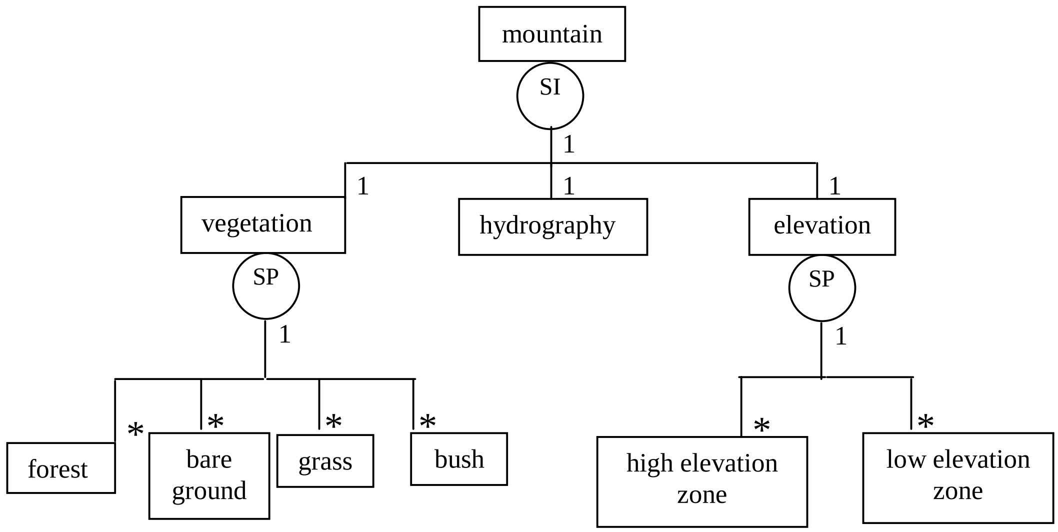


Figure 5.8 A Special Case of Spatial Equal: Spatial Overlay

Having described a framework for and characteristics of spatial PW relationships in Sections 5.3 through 5.6, the incorporation of these spatial PW relationships in a conceptual modeling language such as STUML is considered in the next section.

5.7 Using Spatial Part-Whole Relationships in STUML

In this section, we discuss the incorporation of spatial PW relationships in STUML. We discuss the additional notation required for spatial PW relationships and demonstrate using the supranational organization example from Section 5.1.

As discussed in Section 5.4, we incorporate spatial PW relationships in UML by introducing a new type of association (represented by a circle at one end and an abbreviation inside the circle to indicate the type of spatial PW relationship). The same approach applies to the incorporation of spatial PW relationships in STUML.

As with any other association in STUML, all of the standard UML notations (e.g. cardinality) can be used and the same inheritance rules apply as with standard UML associations. STUML temporal associations can be used to define time-dependent spatial PW relationships. As with any other STUML symbol, an identification label can be included in the circle and used to refer to a specification box giving further details of the spatial PW relationship's semantics. For instance, the specification box is used to specify the base geometric type of part objects for the spatial membership relationship. Variants of the base spatial PW relationship categories shown in Figure 5.2, for example, the special case of spatial equal discussed in Section 5.6.4, can be specified by adding additional constraints based on secondary characteristics to the associated specification box. The specification box for a spatial PW relationship is located in the specification compartment of the whole object type.

Since more than one spatial extent can be associated with an object in STUML (e.g. a supranational organization such as the EU may have a spatial extent describing its location and another spatial extent describing its headquarters), the spatial PW relationship always refers to the spatial extent directly associated at the object level with each whole and part object. Therefore, every object type in a spatial PW relationship must have a spatial extent modeled at the object level. This results in some loss of modeling flexibility from the original STUML, since the same spatial extent cannot be alternatively represented as a named attribute (although UML notes can be used to indicate the name for this spatial extent if that clarifies the semantics). However, it is preferable to adopt this approach in the interest of simplicity rather than adding another symbol to select the relevant spatial extent.

A down arrow is used to indicate any attribute of a part object whose value propagates from the whole object attribute of the same name and domain. Note that this allows the individual specification of propagation in terms of the particular attribute and part object type. Similarly, any attribute of the whole object that is derived from the values of part objects can be indicated with a labeled up arrow. The derivation formula is specified in the corresponding specification box in the whole object's specification compartment. Alternatively, we can use an operation that references the attributes of the part objects instead of a derived attribute.

We illustrate the use of STUML with spatial PW relationships in Figure 5.9 by modeling the supranational organization example with its member countries as follows.

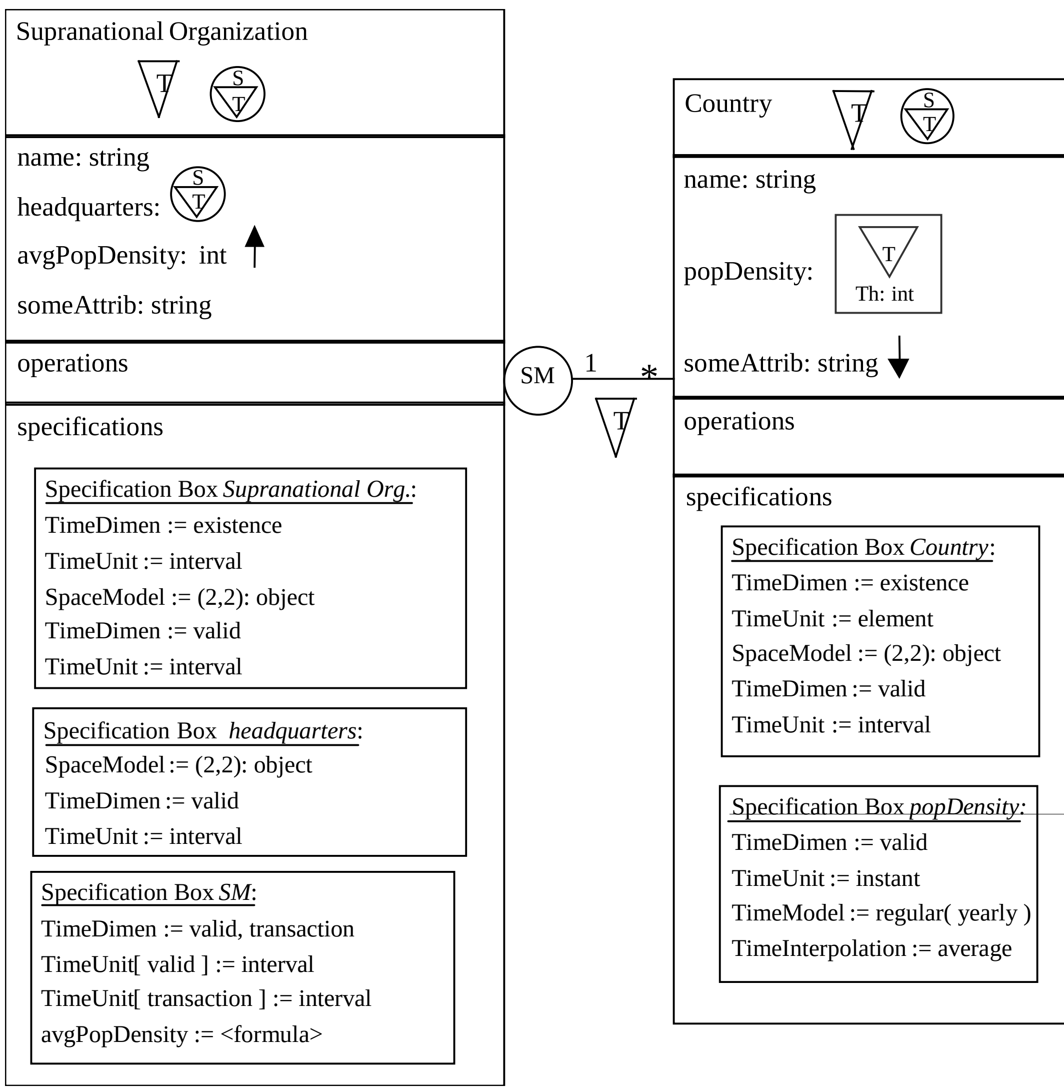


Figure 5.9 Supranational Organization with Spatial Part-Whole Relationship

A supranational organization has a name, headquarters, and average population density (derived from the current population density of its members). An up arrow is used to graphically represent the derivation of average population density from attributes of its parts and the specification box is used to specify the derivation formula.

The members each have a name and population density. There may be cases where the member countries have an attribute whose value propagates from the corresponding attribute (i.e. with the same name) in the supranational organization (e.g. passport for EU). We use the generic attribute *someAttribute* to represent such an attribute. A down arrow is used to illustrate the propagation of *someAttribute*'s value from the whole to the part objects.

A supranational organization, its member countries, and its headquarters each have a two dimensional spatial extent (mapped in 2D space) for which valid time intervals are recorded[5]. Since there may be historical periods during which a country does not exist as an independent object, its existence time is modeled as an element (set of time intervals). Its population density is measured yearly at a given point in time (an instant) and the average value used to estimate population density between yearly measurement points. The existence time of a supranational organization and the valid and transaction time of the association of a country with the organization are modeled as intervals.

Note that the spatial extent of the whole object in a spatial derivation relationship is always derived from that of its part objects. This is already implied by the use of a SP or SM spatial PW relationship; therefore, there is no need for any additional notation at the object level. Similarly, if the spatial derivation relationship, whole's

[5] The time specifications for an object and for its spatial extent are listed separately (see Section 3.5).

spatial extent, and parts' spatial extents are timestamped, then any timestamps for the whole object's spatial extent can be always be derived and there is no need of any additional notation to indicate this at the object level (i.e. it is implied by the use of the spatial derivation relationship). For example, if we know when certain countries were members of the EU and when their spatial extents changed during that time, then we can determine the complete history of the change in the EU's spatial extent over time. The same applies to derived attributes of the whole that are timestamped.

In this example, the spatial PW relationship corresponds to a separately and clearly defined modeling construct with an associated graphical notation. The use of explicit constructs for spatial PW relationships provides a standard graphical notation to represent the semantics of common spatial PW relationships such as spatial membership. In this way, the use of the spatial membership symbol in Figure 5.9 is associated with a well-defined set of constraints such as spatial derivation of the whole object's spatial extent (e.g. the supranational organization's location) and type asymmetry (e.g. a supranational organization cannot be a member of a country). Propagation or derivation of specific attributes can be clearly and consistently modeled on an individual basis as needed for different applications, as demonstrated by the *someAttribute* and *avgPopDensity* attributes respectively in Figure 5.9. Without the use of constructs for spatial PW relationships, users must individually specify such information using UML constraints or notes (described in Section 3.2 and illustrated in Figure 3.2); however, these will not be standardized between users or applications and, without an easily recognized graphical representation, will not be immediately obvious from the schematic diagram.

5.8 Summary

In this chapter, we have defined precisely and unambiguously characteristics relevant to spatial PW relationships and provided a formal definition of spatial PW relationships based on these characteristics. In addition, five different types of spatial PW relationships–spatial part, membership, inclusion, cover, and equal–that are of general utility in spatial applications have been identified and formally defined using a consistent classification framework based on spatial derivation and constraint relationships.

The intention was to develop modeling constructs of general utility in spatiotemporal data modeling and a consistent framework for their definition and use. This framework can easily be extended to accommodate other types of spatial PW relationships as they are identified or as required for individual applications. Alternatively, individual variants can be specified explicitly in applications based on their secondary characteristics using constraints to specify the appropriate values.

A consistent method for modeling such spatial relationships has been demonstrated by adding two modeling constructs to STUML: spatial PW relationships, represented as a new type of association (denoted by a circle), and *propagated/derived attributes* (denoted by a down/up arrow respectively). This allows for a consistent and clear modeling approach that would otherwise require individual users to formulate and specify a complex set of constraints.

In the next chapter, the issue of topological constraints in spatial PW relationships is examined in more detail. A formal classification scheme for binary topological relationships is proposed that is suitable for describing the part-whole topology of spatial constraint relationships. This scheme is then used as a basis to define formal modeling constructs for n-ary topological relationships suitable for specification of part-part topological constraints in spatial PW relationships. Together, the binary and n-ary topological modeling constructs allow the precise specification of whole-part or part-part topology in spatial PW relationships. The n-ary topological modeling constructs can be used, for example, to model the spatial overlay example discussed in Section 5.6.4. More generally, these modeling constructs can be used to extend the basic framework for spatial PW relationships with additional categories or variants based on topological constraints.

Chapter 6

Topological Constraints in Spatial Part-Whole Relationships

6.1 Introduction

Spatial Part-Whole (PW) relationships between spatial objects were defined and classified based on both their spatial and non-spatial characteristics in Chapter 5. This chapter focuses on the specification of one spatial characteristic, topology, in the context of spatial PW relationships. Support for describing both binary and n-ary topological constraints are required to model whole-part and part-part topology respectively. Furthermore, topological modeling techniques are required that support the diversity of geometric types typical of geographic applications (e.g. spatial objects with disconnected components or irregularities such as holes, cuts, crossings or unbounded areas) and are suitable in the context of requirements analysis and design (e.g. simple to use and understand). The objective of this chapter is the development of formal definitions and conceptual modeling techniques that address these requirements.

The work presented in this chapter is motivated by the importance of topology for conceptual modeling of spatial objects. Spatial properties include orientation (directional relations), geometry (also called *metric* properties because they involve quantitative measurements), and topology (properties invariant under rubber-sheet transformations such as scaling, translation, and rotation). Of these three spatial properties, topology serves as a particularly useful descriptor in conceptual application modeling for two reasons. First, because it is qualitative, topology is more intuitive than a quantitative property such as geometry. It is easier for a user to recognize a specific topology such as *contains* than a specific metric such as a *distance* measurement. Second, topology is a more reliable discriminator of object configurations than geometry or orientation since it is preserved through many of the common changes and distortions that can occur in representations of real world objects.

In the context of spatial PW relationships, topological relationships between the whole and all of its parts and between the individual parts are important for constraint specification at the modeling stage and update and query at the operational stage of application and database development. These relationships play an important role in the classification of spatial PW relationships, as described in Chapter 5. The binary topological relationship between the whole and the parts (their GU) is the primary characteristic used to differentiate between various spatial constraint relationships and differentiate them from spatial derivation relationships. The set-based topological relationship between the parts is described as one of the secondary characteristics used to define variants of the base spatial PW relationship types, such as the overlay variant of spatial equal described in Section 5.6.4 and illustrated in Figure 5.8.

In this chapter, a classification framework and modeling constructs intended to facilitate specification of general topological constraints between two or more spatial objects, in the context of spatial PW relationships, are proposed. The proposed method should be general enough to be suitable for a range of different applications yet simple to use and understand. Specifically, the goal is to cater for the following:

- composite (i.e. having disjoint components), complex (e.g. irregular, mixed dimension), and higher dimensional (i.e. 3D) as well as simple (i.e. connected) 1D and 2D spatial objects,
- specification of n-ary as well as binary topological relationships that are useful for modeling part-part as well as whole-part topological constraints in spatial PW relationships, and that are of general applicability in the context of composite spatial objects, and
- a level of complexity suitable for use in conceptual modeling.

In Section 6.2, previous work on topological relationships is reviewed. The assumptions and terminology relevant to this chapter are given in Section 6.3. Section 6.4 describes a simple approach to modeling binary topological relationships based on intersection and difference of spatial extents. This is extended to describe n-ary topological relationships in Section 6.5. In Section 6.6, we apply these methods to the spatial PW relationships discussed in Chapter 5, using the proposed binary and n-ary topological relationships to describe constraints on whole-part and part-part relationships respectively. Examples are given to show the applicability and ease of use of the approach adopted. The chapter is summarized in Section 6.7.

6.2 Review of Topological Relationships in the Literature

Binary topological relationships have been considered in the context of object-oriented modeling [Borges99, Hend98, Yang96] and spatial query languages [Card93, Egen94a, Guti91, Yang96]. Although not considering spatial objects in general, Henderson [Hend98] does include a modeling construct for spatial containment. An extension of the object-oriented modeling language OMT for spatial queries is proposed in Borges [Borges99] based on earlier work by

Clementini [Clem93] classifying binary topological operators. Both Egenhofer [Egen94a] and Yang [Yang96] base their spatial query languages on the classification scheme proposed in Egenhofer [Egen91b]. A limited, ad hoc set of binary topological operators is proposed in the spatial algebra of Güting [Guti91] and the graphical query language of Cardenas [Card93]. However, none of this work focuses on providing a comprehensive classification and definition of topological relationships.

Classification schemes for binary topological relationships have been the subject of extensive study over the years [Clar00, Clem93, Clem94, Clem95a, Clem95b, Egen90, Egen91a, Egen94a, Egen94b, Hadz92, Tryf97a], with the research focus on the development of mathematical formalisms to precisely and exhaustively specify topological relationships. The majority of topological research to date is based on assumptions that are too restrictive for use as a general modeling tool. They assume objects with simple, bounded (closed, i.e. including all boundary points), and regular regions and lines embedded in 2D space. However, many Geographic Information Systems applications involve semantic entities having holes, discontinuities, and other irregularities. For example, the country Italy has a spatial extent with an interior hole representing the autonomous entity Vatican City. The land mass of countries such as Denmark, Greece, and Indonesia does not form one contiguous region, but instead is composed of physically separated spatial parts. Applications mapping the spatial distribution of a non-spatial characteristic (e.g. soil acidity, soil type, linguistic groups) can lead to the formation of an *archipelago*, a single semantic entity having widely dispersed spatial fragments. Additionally, the dimension of spatial data objects may range from zero to three dimensions. For example, applications measuring location in terms of latitude, longitude, and elevation require 3D spatial objects and embedding space.

Recently, researchers have tried to address the challenges of extending topological research to include a wider range of spatial objects, including regions with holes and lines with multiple end-points [Egen91a, Egen94b] and composite spatial extents [Clar00, Clem94, Clem95a, Tryf97a]. In Egenhofer [Egen91a, Egen94b] and Tryfona [Tryf97a], topological relations between spatial objects are described based on boundary and interior intersections between object closures (including all points interior to and on the object's boundary to regularize spatial extents with holes or discontinuities) and object components or discontinuities (e.g. holes or gaps). The most comprehensive work, in terms of the range of spatial object types considered, is described in Clementini [Clem94]. It considers bounded composite[1] spatial objects formed exclusively from either lines (possibly with self-crossings or extra end-points beyond the usual two), points, or regions (possibly with holes). A mutually exclusive and complete set of binary topological relations—*touch, in, overlap, disjoint,* and *cross*—is defined based on boundary, interior, and object intersections (and their dimensions), using separate definitions of boundary and interior for each type of composite spatial object. In Clementini [Clem95a], equivalent definitions for the binary topological relations between composite regions are given in terms of relations between their components (using component pairs composed of a component from each composite object).

In Claramunt [Clar00], a more comprehensive solution to the challenge of describing topological relationships between composite objects at the component level is described. This is based on a complete set of adverbs that can be used to

[1] Used to mean disjoint components in this thesis, but including regions connected by a finite set of points in Clementini. The difference is not significant in this context and so can be disregarded.

refine an existing binary classification scheme by extending it to the component level. These adverbs are independent of the set of topological relationships or types of components used and are discussed later with respect to the topological relations proposed in this thesis chapter. Assume that we are given the following:

- two composite spatial extents *A*, *B* consisting of regions*;*
- a binary topological relationship *R*, where R(A,B) means that *A* is related to *B* by the relationship *R*, and its inverse R_{rev} (e.g. *contains* and *inside*, note that *R* = R_{rev} for symmetric *R* such as *equals*); and
- component pairs (c_A,c_B) consisting of one component each from *A* and *B.*

Then seven adverbs are used to describe possible micro-configurations as follows[2]:

- *never* (*R*(*A*,*B*)) when (*R*(c_A ,c_B)) is not true for any of the component pairs (c_A,c_B),
- *mostly* (*R*(*A*,*B*)) when for each of *B*'s components c_B, *A* has a component c_A such that *R*(c_A,c_B); this means that there may be one or more components c_A of *A* such that *R*(c_A,c_B) is not true for any component c_B of *B*,
- $mostly_{rev}$ (*R*(*A*,*B*)) for the inverse case of *mostly*, when for each of *A*'s components c_A, *B* has a component c_B such that *R*(c_B,c_A); this means that there may be one or more components c_B of *B* such that *R*(c_B,c_A) is not true for any component c_A of *A*,
- *partially* (*R*(*A*,*B*)) when (*R*(c_A ,c_B)) for some (at least one) of the component pairs (c_A, c_B) and all the other component pairs (c_A, c_B) are disjoint,
- *occasionally* (*R*(*A*,*B*)) when (*R*(c_A ,c_B)) for some (at least one) of the component pairs (c_A, c_B),

[2] Note that *entirely*⇒*completely*⇒*mostly*⇒*occasionally* and *entirely*⇒*completely*⇒*partially* for the definitions as stated here and in Claramunt [Clar00]. Therefore, the proof of mutual exclusivity for these adverbs given in [Clar00] *does not hold* for the definitions as stated in the same paper.

- *completely* ($R(A,B)$) when *mostly* ($R(A,B)$) and *mostly* ($R_{rev}(B,A)$),
- *entirely* ($R(A,B)$) when for every component pair (c_A, c_B), ($R(c_A, c_B)$)and ($R_{rev}(c_B, c_A)$).

Figure 6.1 illustrates the difference between the adverbs *Mostly*, *Completely*, and *Entirely* using the definition of *Touch* (boundary overlap) from Clementini [Clem93]. The spatial extent *A* is colored red (with left diagonal stripes for 2D regions or 2D faces of solid 3D volumes) and the spatial extent *B* is colored blue (with right diagonal stripes for 2D regions or 2D faces of solid 3D volumes). This convention is followed throughout the chapter. In Figure 6.1, *A* and *B* are each composite with disjoint components. Thus, *A* consists of four disjoint 2D regions and one 1D point in Figure 6.1 (a) and two disjoint 2D regions in Figure 6.1 (c).

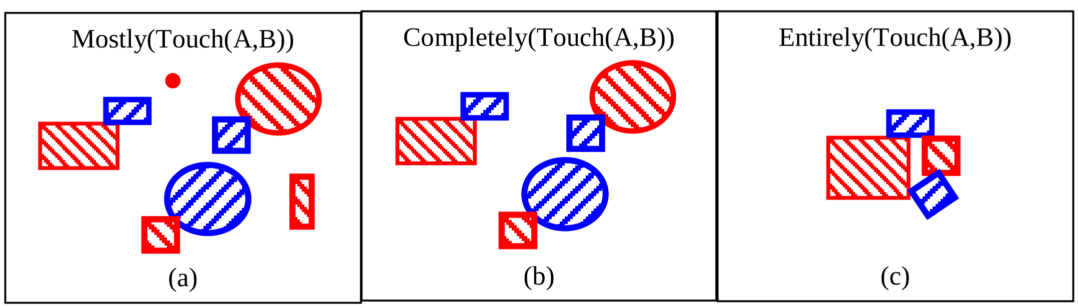


Figure 6.1 Illustrating Claramunt's Adverbs

In Figure 6.1 (a), every component of *B* touches a component of *A*; however, there are some extra components of *A* not touching any component of *B*. In Figure 6.1 (b), every component of *B* touches some component of *A* and every component of *A* touches some component of *B*. In Figure 6.1 (c), every pair of *A* and *B* components (i.e. with one component from *A* and one from *B*) touch.

To illustrate the use of these adverbs in an application context, consider the components of two countries such as Indonesia and the Philippines consisting of island archipelagos. Ideally, they should *never* overlap (i.e. no pair of components, one from each country, overlaps). However, if the reality of boundary disputes is considered, the two countries may *partially* overlap (i.e. there may be some component pairs with overlap where there are boundary disputes, but otherwise component pairs are disjoint).

In the context of modeling spatial PW relationships, existing topological research has limitations with respect to the range of spatial data types considered, the understandability of the models proposed, and support for modeling n-ary topological relationships (required to model constraints between spatial parts). To our knowledge, none of the binary topological classification schemes to date explicitly consider spatial extents that are not closed; irregularities such as loops, punctures, and cuts; mixed-dimension composites (e.g. a single composite object consisting of regions, lines, and points); or 3D objects and embedding space. In addition, even when the work considers more complex spatial objects, it is fundamentally based on boundaries and interior intersections [Egen91a, Egen94b, Clem94, Clem95a, Tryf97a]. Although this allows a high-degree of expressiveness in terms of being able to precisely describe a wide range of topological configurations, this comes at the price of increased complexity and reduced understandability. An example is the redefinition of boundary and interior required for each type of composite spatial object in Clementini [Clem94] or the identification of topological classes by number rather than name in Egenhofer [Egen91a] and in Hadzilacos [Hadz92] or by complex conjunctions in Tryfona [Tryf97a].

A further problem is that the formal definitions of boundary, interior, and dimension used vary considerably depending on the underlying mathematical model assumed and may not match the intuitive understanding the user has of these concepts. To illustrate, consider the differences between the point-set topologic model [Arms79]—based on sets of points—and the algebraic topologic model [Gibl77]—based on simplicial complexes. The definitions of boundary and interior are formulated only in terms of the dimensions of the spatial object in algebraic topology. In point-set topology, the definitions depend on the dimensions of the embedding space as well. Assuming an algebraic topologic model, the boundary of a simple line embedded in 1D, 2D, or 3D space consists of the line's end-points. In contrast, the boundary of a simple line in the point-set topologic model consists of the line's end-points in 1D but *not* in 2D or 3D embedding space. When the same line is embedded in a plane or volume (i.e. 2D or 3D embedding space), the whole line is considered to be the boundary with no interior points. These problems are discussed in detail in Zhilin [Zhil00], including the additional problem of defining boundary and interior for discrete, rasterized space. Similarly, the concept of dimension—used, for example, in Clementini [Clem94, Clem95a] to describe intersections—is less intuitive when applied in the following cases:

- to a composite spatial object having components of different dimensions,
- to a single spatial object composed from objects of different dimensions connected by a finite number of points (e.g. a line connected to a region by one point), or
- to intersections consisting of sets of disconnected spatial extents (is the dimension of a set of points the same as that for one point?).

In the context of analysis and design of spatiotemporal applications, we need a different modeling approach to address the requirements of application developers. The level of complexity must be suitable for use in early application development phases, i.e. general enough to be suitable for a range of different applications yet simple to use and understand. This potentially means sacrificing, to some degree, the expressiveness of the model (i.e. the number of different topological relationships that can be described) for the sake of generality (i.e. being able to model the range of different types of spatial objects that are found in spatiotemporal applications) and clarity (i.e. based on concepts and classifications that are highly intuitive).

Furthermore, in order to model spatial PW relationships, we must be able to describe the n-ary topological relationships between the parts. Topological research to date has focussed on binary topological relationships suitable only for describing the relationship between the whole and the geometric union of its parts. Research work describing n-ary temporal relations in the context of multimedia databases can be evaluated for its potential relevance to defining n-ary topological relations. Little [Little93] defines an n-ary temporal relation consisting of an ordered finite sequence of temporal intervals where any two adjacent intervals have an identical temporal relation. However, ordering is not suitable for describing topological relationships between a set of spatial objects since there is no inherent linear order in space (except in the special case of 1D space).

These limitations highlight the need for a topological classification scheme and conceptual modeling techniques applicable to spatial PW relationships in geographic applications, such as those that will be proposed in Sections 6.4 and 6.5. In the next section, basic assumptions and terminology related to topology are reviewed as a basis for discussing those proposals.

6.3 Assumptions and Terminology

In this section, we review the assumptions and terminology relevant to the topological modeling techniques presented in Sections 6.4 and 6.5. In the context of geographic applications (see the discussion in Section 2.2.2), it is sufficient to assume an Euclidean model of space with spatial objects embedded in that space, where the embedding space could be 1D, 2D, or 3D. The classification described here holds under either point-set or algebraic topology (based on sets of points [Arms79] or simplicial complexes [Gibl77] respectively); therefore, either can be used as a theoretical basis for discussion. We choose point-set terminology here as the more natural choice, given that the classification scheme proposed in Section 6.4 uses set-based concepts of intersection and difference. A *spatial extent* is then described as a subset of the points in the embedding space. The spatial extent is considered to be *connected* if any two of its points can be connected by a path consisting entirely of points within the spatial extent and considered to be *disconnected* otherwise. It is *weakly connected* if the same spatial extent becomes disconnected after removal of a finite number of points and *strongly connected* otherwise. The *dimension* of a strongly connected spatial extent is zero, one, two, or three for points; arcs or loops; areas; and volumes respectively.[3] A spatial extent that is disconnected is called a *composite* spatial extent consisting of a finite set of disjoint (non-intersecting) *components*, each of which is a weakly or strongly connected spatial extent. A *Geometric Union* (GU) of a finite number of spatial extents is the set consisting of all the points from each of the spatial extents including, if composite, all of their components.

Point-set topology is built from the concept of *neighborhoods*, where there exists a neighborhood both for every point in space and inside the intersection of any two neighborhoods for that point. A *near point* for a spatial extent is one where each of the point's neighborhoods includes a point in the spatial extent. A spatial extent—whether *connected* or *composite*—forms an *open set* if every point has a

[3] Dimension can be more precisely defined in terms of topological equivalence [Worb95]; however, an informal description suffices since dimension is not used in the classification proposed here.

neighborhood completely within the spatial extent and forms a *closed set* if it includes all its near points. A spatial extent is called *unbounded* if open, *bounded* if closed, and *partially bounded* otherwise. The largest *open set* in the spatial extent is usually called the *interior* and the remaining points the *boundary*; however, these terms are not always used consistently in the literature for the reasons discussed earlier. A spatial extent is called *simple* if it is connected and *regular* if it is bounded and contains no irregularities (e.g. no holes, crossings, isolated missing punctures or cuts, extra end-points for lines).

We adopt the well-known concept of a minimum bounding box (i.e. MBR or MBC). Rather than describing an inherent property of a spatial object, the minimum bounding box is used to approximate an object's location in the embedding space. The minimum bounding box in a 2D embedding space for a given spatial extent is the smallest rectilinear rectangle completely enclosing that spatial extent. More generally and without restricting the dimension or the type of figure used, the term *minimum bounding figure* is used here to refer to any simple, regular bounding figure in 1D to 3D space. For example, a circle could be used in 2D or cube in 3D space. This concept is important for establishing whether the components of two disjoint composite spatial extents are interspersed or completely separate.

With this foundation, the proposed classification scheme for binary topological relationships is described in the following section.

6.4 Modeling Binary Topological Relationships

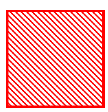
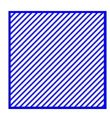
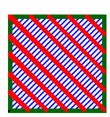
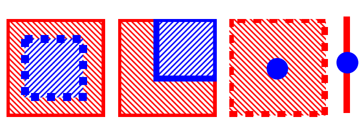
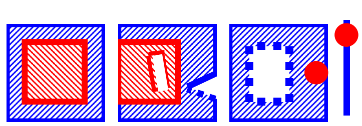
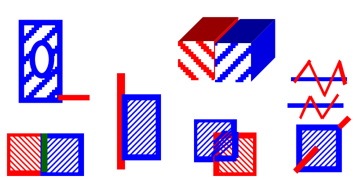
Consider the building site example described in Chapter 5. It is essential that any structure erected on that site does not extend beyond the site boundary. In an analogous manner, when an administrative region is divided into voting districts or land-use zones, the combined spatial extents of the resulting voting districts or land-use zones must be exactly equal to that of the administrative region. To model these constraints, we require a formal yet simple method of describing binary topological constraints such as the containment relationship between the building site and its structures or the equality constraint between the administrative region and its combined voting districts. In this section, we propose a classification method for topological relationships specifically designed to facilitate conceptual modeling of binary whole-part topological constraints in spatial PW relationships.

The classification is based on a mathematical formalism so that it (*i*) can be precisely specified and consistently interpreted to aid communication and (*ii*) can be automatically translated into an implementation in later stages of development. However, this formalism must be simple enough to be completely intuitive. Furthermore, as discussed previously, it should support the range of geometric types typical of geographic applications.

We introduce a two-level classification of topological relationships between two spatial extents. The approach adopted is not predicated on the geometric type restrictions described in Section 6.2 for other topological classification schemes in the literature. Thus, a given spatial extent can consist of any finite number of disconnected or weakly connected parts of the same or different dimensions (between 0D and 3D); can have irregularities such as holes, punctures, cuts, self-crossings, extra end-points and loops; and can be bounded, partially bounded, or unbounded.

The first level of classification is based only on whether the intersection and difference of the two spatial extents is empty or non-empty, concepts that are easily understandable, intuitive, and not dependent on the dimension of the embedding space. This classification scheme is illustrated in Table 6.1.

Table 6.1 Binary Topological Relationships using Intersection and Difference

Example(s) of *R*(*A*, *B*) **(*R* is the relationship, spatial extent *A* is red, *B* is blue)**	**Intersection**	***A* – *B*:** **Forward Difference**	***B* – *A*:** **Reverse Difference**	**Name of *R***
	$\emptyset$	$\emptyset$	$\emptyset$	not named ($A,B = \emptyset$)
	$\emptyset$	$\neg\emptyset$	$\emptyset$	not named ($B = \emptyset$)
	$\emptyset$	$\emptyset$	$\neg\emptyset$	not named ($A = \emptyset$)
	$\emptyset$	$\neg\emptyset$	$\neg\emptyset$	Disjoint
	$\neg\emptyset$	$\emptyset$	$\emptyset$	Equal
	$\neg\emptyset$	$\neg\emptyset$	$\emptyset$	Contains (Nested)
	$\neg\emptyset$	$\emptyset$	$\neg\emptyset$	Inside (Nested)
	$\neg\emptyset$	$\neg\emptyset$	$\neg\emptyset$	Connected

Colors are used to distinguish between the two spatial extents *A* (in red) and *B* (in blue). A dotted line is used to indicate a partially bounded or unbounded spatial extent and diagonal stripes indicate a 2D region or 2D face of a solid 3D volume. Only simple spatial extents are used in the table for the sake of understandability. Thus each pair of red and blue figures represents a different example of a binary topological relationship (i.e. with different values for *A* and *B*). So the categories *Contains, Inside,* and *Connected* each have multiple pairs shown in Figure 6.1 and thus multiple examples. Examples of composite spatial extents appear in Figures 6.2 through 6.4. Spatial dimensions of simple spatial extents or components of composite spatial extents range from 0D to 3D in the examples given in Figures 6.1 through 6.4.

After eliminating trivial cases where at least one of the two spatial extents is the empty set (∅), we have the following topological categories: disjoint, equal(s), contain(s), inside, and connected. Essentially, non-intersecting spatial extents (when the intersection is the empty set) are disjoint and intersecting spatial extents (when the intersection is non-empty) have one of the following topological relationships:

- equal (the difference in both directions is the empty set),
- contain (the forward difference, *A*–*B*, is non-empty and the reverse difference, *B*–*A*, is the empty-set),
- inside (the forward difference, *A*–*B*, is the empty-set and the reverse difference, *B*–*A*, is non-empty), or
- connected (the difference in both directions is non-empty).

So, for example, the *equal* and *contain* relationships can be used to model the administrative region and building site examples respectively.

This set of relationships is complete and mutually exclusive for two non-empty spatial extents, e.g. any relationship between two objects falls into exactly one of these categories. The intersection, forward difference, and reverse difference of any two spatial extents must each be either empty or non-empty. Therefore, by considering exhaustively all the possible permutations when defining the topological categories in Table 3.1, the resulting categories must be both complete and mutually exclusive. The two non-symmetrical relationships, *contain* and *inside* (i.e. contained-by), can be combined through disjunction into one symmetric *nested* relationship where either the forward difference or the reverse difference, but not both, is the empty set. The *connected* and *disjoint* categories have a further level of classification defined.

Connected objects can be further classified based on whether they have a *boundary*, *interior*, or *mixed overlap*, i.e. whether their intersection includes only boundary, only interior, or both boundary and interior points. Since boundary and interior points usually represent semantic differences in applications, it is useful to be able to specify whether the intersection involves object boundaries, interiors, or both. For example, in the case of voting districts for a given administrative region, interior points are used to represent administrative jurisdiction and boundary points are used to represent a change in jurisdiction. This example will be discussed further in Section 6.5, in the context of n-ary topological constraints between spatial parts.

A crucial aspect of the sub-categories of the *connected* relationship is that, in contrast to other proposed topological classification schemes, the only assumption is that every point in a spatial extent must be either a boundary or interior point but not both. Further definition is left to the user as appropriate to specific application requirements. This approach supports the intuitive notion that boundary points differ semantically from interior points, but does not dictate further those aspects of the definition that may vary between applications.

Figure 6.2 illustrates the three different sub-categories of *connected* objects. The spatial extents *A* and *B* are shown in red and blue respectively and separate pairs of red and blue figures represent different examples (and different instances of the spatial extents *A* and *B*). Note that the *Boundary-Overlap* category has one example of a composite spatial extent *A* (consisting of several disjoint line segments) and one example of a weakly-connected spatial extent *A* (consisting of a line connected by one point to a rectangle with a hole). A dotted line is used to indicate a partially bounded or unbounded spatial extent and diagonal stripes to indicate a 2D region.

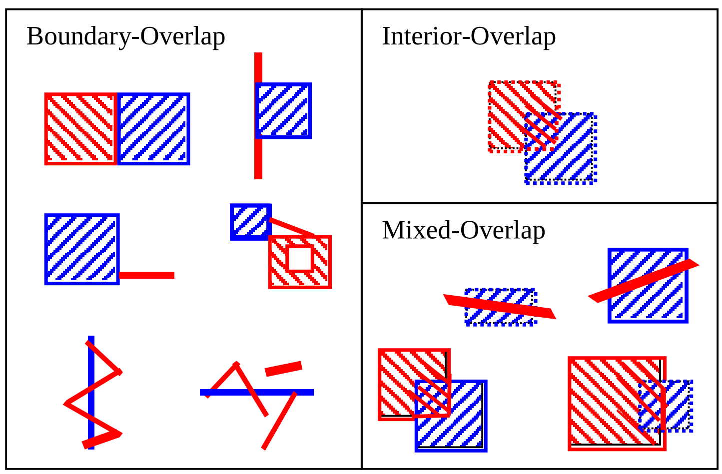


Figure 6.2 Sub-Categories of the Connected Relationship

In this figure, we assume a 2D embedding space and definitions of boundary and interior dependent on the embedding dimension as described in Section 6.2 and in Worboys [Worb95] for point-set topology. Thus (*i*) a single point and (*ii*) every point of a 1D line embedded in 2D space is a boundary point. If we were to assume that the embedding space was 3D instead of 2D, then all the examples of *Interior-Overlap* or *Mixed-Overlap* would also become *Boundary-Overlap*. This is because all the points in a 2D area embedded in 3D space are boundary points under the definition for boundary and interior assumed for this example.

A *disjoint* relationship between two spatial entities can be further distinguished based on whether their spatial extents interpenetrate, i.e. whether the minimum bounding figures of the two objects intersect. The method used to calculate the minimum bounding figure is application dependent, e.g. with respect to the orientation of the axes and granularity. As with the definition of boundary and interior used for the sub-categories of *connected*, the decision as to exactly how to determine the minimum bounding figure is left to the user. *Separate* is a disjoint relationship where the minimum bounding figures of the two spatial extents do not intersect and *interpenetrating* is a disjoint relationship where they do intersect. This distinction is particularly relevant for the applications involving so-called *archipelagos*, such as the distribution of soil types discussed in Section 6.2, where the spatial parts in a spatial PW relationship are widely dispersed.

Figure 6.3 shows examples of separate and interpenetrating disjoint relationships between two composite spatial extents *A* (in red) and *B* (in blue), each having its own set of 0D-3D components embedded in 3D space. A dotted line is used to indicate a partially bounded or unbounded spatial extent and diagonal stripes show a 2D region or 2D face on a solid 3D volume.

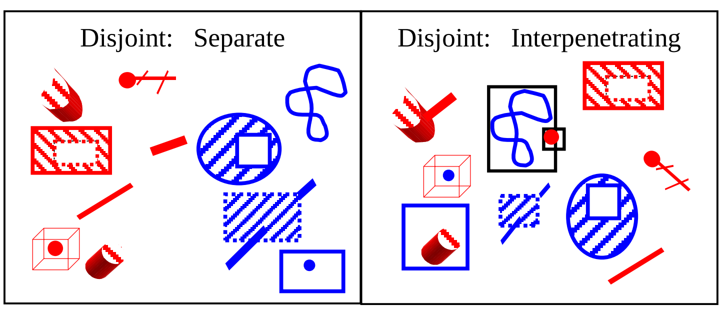


Figure 6.3. Sub-Categories of the Disjoint Relationship

Even two simple spatial extents can have an interpenetrating disjoint relationship. This is illustrated by the intersection of the two minimum bounding boxes shown in black, where one minimum bounding box encloses a single component from the composite spatial extent *A* and the other minimum bounding box encloses a single component from the composite spatial extent *B*.

The second level of classification consists of complete and mutually exclusive sub-categories within the specific category. For example, every disjoint relationship is either separate or interpenetrating but not both. That follows logically from the definition of the two categories based on whether the minimum bounding figures of the two spatial extents intersect. Each minimum bounding figure represents a spatial extent, and we previously said that the intersection of two spatial extents must be either empty or non-empty. Similarly, mutual exclusivity of the *connected* subcategories follows logically from the assumption stated earlier that a spatial extent can be completely partitioned into mutually exclusive sets of boundary and interior points.

Thus the eight categories—*separate*, *interpenetrating*, *equal*, *contain*, *inside*, *boundary-overlap*, *mixed-overlap*, and *interior-overlap*—represent a complete and mutually exclusive set of binary topological relationships. Note that we do not adopt a minimal set, since that would require the use of negation, a less natural modeling technique. The more general categories—*disjoint*, *intersecting*, *nested*, *connected*—can be derived from these eight relationships. Except for *contain* and *inside*, all of the relationships are symmetric. We use the notation *R*(*A*,*B*) to indicate that *A* has the relationship *R* with *B*, where *A* and *B* are spatial extents and *R* is a binary topological relationship.

Although the set of topological relationships is complete and mutually exclusive, certain applications may require a greater degree of precision even at the requirements analysis and conceptual modeling phases of system development. For applications requiring a more detailed understanding of the topological relationships between components in pairs of composite objects, Claramunt's adverbs [Clar00] (described in Section 6.2) can be employed with the binary topological relationships introduced in this section. Note that the adverbs were originally defined in Claramunt [Clar00] only for a union of bounded, simple, non-intersecting regions in 2D space (i.e. composite spatial extents with simple, bounded regions as components). However, the adverbs apply equally as well in the more general case when each component is only constrained to be connected and substituting the word *component* for *region* in the adverb definitions. The adverbs can then be used explicitly with the binary topological relationships defined in this section (or disjunctions and/or conjunctions of those constraints) to express constraints between components of two different composite objects. This is illustrated in Figure 6.4 using two composite spatial extents *A* (in red) and *B* (in blue), each having its own set of 0D through 3D components embedded in 3D space. A dotted line is used to indicate a partially bounded or unbounded spatial extent and diagonal stripes a 2D region or 2D face of a solid 3D volume.

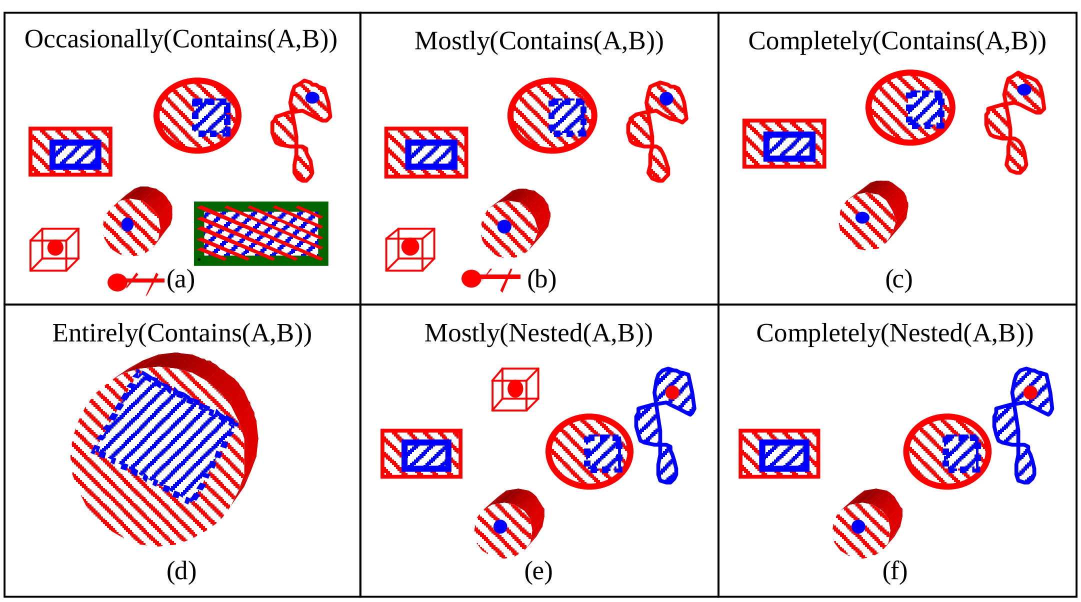


Figure 6.4 Using Adverbs for Component-Level Topological Constraints

Figure 6.4 (a) through (d) are all examples of the binary inclusion relationship *Contains*(*A*,*B*). The adverbs can be used to specify more restrictive constraints and differentiate between these four examples as shown in Figure 6.4, with increasing restrictions from (a) to (d). Both Figure 6.4 (e) and (f) are examples of *Connected*(*A*,*B*). Note that although *Nested* is true at the component level (based on pairwise comparison of components) in Figure 6.4 (e) and (f); it is not true at the composite level (at the level of the whole object), since both the forward and reverse differences are non-empty.

Adverbs can be used to indicate specific cases of *Connected*(*A*,*B*) with nesting of component pairs, with increasing restrictions from (e) to (f). In Figure 6.4 (e), the use of the adverb *Mostly* indicates that every component of *B* must be nested with a component of *A*. The use of the adverb *Completely* in Figure 6.4 (f) further specifies that every component of *A* must be nested with a component of *B*. Similarly, to specify that two composite spatial extents should have at least one component in common, we can use:

$$\mathit{Occasionally}(\ \mathit{equal}\ (\ A,B\ )\ )$$

Or to specify that each component of either composite spatial extent should be equal or have a boundary-overlap with at least one component of the other composite spatial extent, we can use:

$$\mathit{Completely}(\ \mathit{equal}\ (\ A,B\ ) \vee \ \mathit{boundary\text{-}overlap}\ (\ A,B\ )\ )$$

In other cases, further distinctions in the topological relationships between connected components may be required. For example, what if we need to be able to distinguish between the different cases of boundary-overlap shown in Figure 6.2? In this case, the binary topological relationships described here, even in combination with the adverbs from Claramunt [Clar00], will not suffice. In these specific cases, models involving a more limiting set of assumptions and more complex geometric concepts, such as those described in Section 6.2, can be employed.

Note that the efficient implementation of the binary topological constraints in later development phases based on a specific topological classification scheme (whether those described in Section 6.2 or that proposed here) requires the use of representation-based algorithms to verify intersection and difference of spatial extents, their boundaries, and their interiors. An overview of the types of representations used for 0D through 3D spatial objects and associated algorithms used for these operations are described in [Worb95].

The next subsection discusses the semantics of the proposed binary topological relationships when applied to composite spatial objects, from the perspective of the possible component configurations characterizing each relationship. These semantics were illustrated by example in this section. They are further elucidated in the next subsection through a comparison of the proposed relationships to equivalent expressions using Claramunt's adverbs [Clar00] to describe explicitly component relationships between the two composite objects.

6.4.1 Expressing Binary Topological Relationships Using Adverbs

The topological relationships *disjoint*, *equal*(*s*), *contain*(*s*), *inside*, *nested*, and *connected* can be re-stated in terms of component relationships between two composite objects using the adverbs from Claramunt [Clar00]. Assuming a binary topological relationship *R* and two composite spatial extents *A* and *B*, the same relationship can be expressed in terms of component relationships as follows:

- $disjoint\,(A,B) \Leftrightarrow entirely(\,disjoint\,(A,B)\,)$
- $equal\,(A,B) \Leftrightarrow completely(\,equal\,(A,B)\,)$

- *inside* (*A*,*B*)

  $\Leftrightarrow$ *mostly*(*contain* $\vee$ *equal* (*B*,*A*)) $\wedge$ ($\neg$ *completely*(*equal* (*A*,*B*))

- *contain* (*A*,*B*)

  $\Leftrightarrow$ *mostly*(*contain* $\vee$ *equal* (*A*,*B*)) $\wedge$ ($\neg$ *completely*(*equal* (*A*,*B*))

- *nested* (*A*,*B*)

  $\Leftrightarrow$ *inside* (*A*,*B*) $\vee$ *contain* (*A*,*B*)

  $\Leftrightarrow$ (*mostly*(*contain* $\vee$ *equal* (*B*,*A*)) $\vee$ *mostly*(*contain* $\vee$ *equal* (*A*,*B*)))
  $\wedge$ ($\neg$ *completely*(*equal* (*A*,*B*))

- *connected* (*A*,*B*)

  $\Leftrightarrow$ ($\neg$ *disjoint* (*A*,*B*)) $\wedge$ ($\neg$ *nested* (*A*,*B*)) $\wedge$ ($\neg$ *equal* (*A*,*B*))

  $\Leftrightarrow$ *occasionally*(*connected* $\vee$ *equal* $\vee$ *nested* (*A*,*B*))

  $\wedge$ ($\neg$ *mostly*(*contain* $\vee$ *equal* (*B*,*A*)))

  $\wedge$ ($\neg$ *mostly*(*contain* $\vee$ *equal* (*A*,*B*)))

Note that the equivalent definition of *contain* using the adverbs has to account for all four cases of *contain* in Figure 6.4 (a) through (d). Similarly, the equivalent definition of *connected* must account for all of the possible component combinations, some of which are illustrated in Table 6.1 under the *connected* category and in Figure 6.4 (e) and (f).

The sub-categories of *connected* and *disjoint* have no equivalents using the adverbs. For the former case, the initial instinct would be to define the categories of connected as follows:

- *boundary-overlap*(*A*,*B*)

  $\Leftrightarrow$ *connected*(*A*,*B*) $\wedge$ *never*(*interior-overlap*(*A*,*B*))

- *interior-overlap*(*A*,*B*)

  $\Leftrightarrow$ *connected*(*A*,*B*) $\wedge$ *never*(*boundary-overlap*(*A*,*B*))

However, consider the case where *connected* composite spatial extents have some component pairs *nested* (see Figure 6.4(e) and 6.4(f)) or *equal*. To determine which *connected* sub-category we have, we must be able to determine whether the intersection of a *nested* or *equal* component pair includes component boundary and/or interior points. This would require the definition of additional binary topological relations based on whether the intersection of two *nested* or *equal* spatial objects consists of only boundary, only interior, or both types of points.

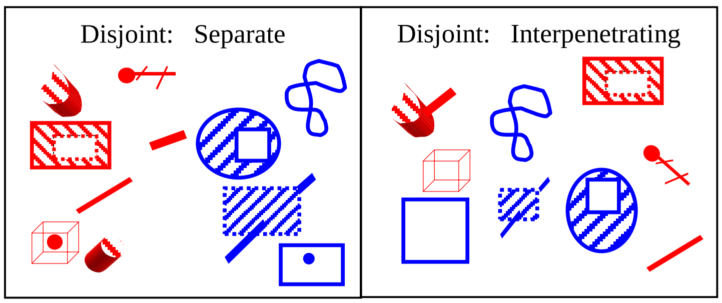


Figure 6.5 Disjoint Relationship Examples with Entirely Separate Components

Finding an equivalent of the *disjoint* sub-categories in terms of component relations is more difficult and cannot be resolved by defining additional binary topological relations. This is illustrated in Figure 6.5, using two composite spatial extents *A* (in red) and *B* (in blue), each having its own set of 0D-3D components embedded in 3D space, where dotted lines indicate an unbounded or partially bounded spatial extent and diagonal lines indicate a 2D region or 2D face of a solid 3D volume.

Two *interpenetrating* or two *separate* composite extents can both have *entirely separate* pairs of components. For example, *entirely*(*separate*(*A*,*B*)) is true for both the examples of the *disjoint* relationship (i.e. *separate*(*A*,*B*) and *interpenetrating*(*A*,*B*)) shown in Figure 6.5. Therefore, none of the other binary topological relations defined can be used to distinguish further between these categories at the component level. In fact, the definitions of *disjoint* sub-categories *interpenetrating* and *separate* depend, for the two spatial extents, on the minimum bounding figure of the GU of all of the components in a composite spatial extent and cannot necessarily be determined by the examination of individual components. Therefore, Claramunt's adverbs, which are based on topological relationships between pairs of individual components, are not relevant in these two cases.

6.5 Modeling N-ary Topological Relationships

The binary topological classification described previously in Section 6.4 is sufficient to describe topological constraints between a whole and the GU of its parts (i.e. between the *spatial extent* of the whole and the GU of the *spatial extents* of its parts, where the latter is called the *part-union*). However, n-ary topological relationships are required to describe topological constraints between the parts. For example, the voting districts created for an administrative region cannot have overlapping interiors, as this would lead to the possibility of a single constituent being able to vote in more than one district. In this section, a general method of modeling n-ary topological relationships is described.

Given some binary topological relationship *R* defined for two spatial objects, how can this be extended to *n* spatial objects? For example, how can the definition of *boundary-overlap* be extended to describe the constraint on the set of voting districts, i.e. that none of the voting districts can share interior points? It follows logically that if a binary topological constraint *R* is extended to *n* spatial objects at least one of the following three conditions is true:

- *Condition 1:* *R* holds for every pair (i.e. *all*) of the *n* spatial objects.
- *Condition 2:* *R* holds for at least one pair (i.e. *some*) of the *n* spatial objects.
- *Condition 3:* *R* holds for no pair (i.e. *none*) of the *n* spatial objects.

Although complete (i.e. given a binary relationship *R* and *n* spatial extents at least one of the three conditions holds), the three conditions are not minimal, since condition 3 is equivalent to the negation of condition 2. Nor are they mutually exclusive, since condition 2 does not preclude condition 1. Exclusivity would require that the second condition be modifed from *some* to *some but not all* as follows:

- *R* holds for at least one pair (i.e. *some*) of the *n* spatial objects and *R* does not hold for at least one pair (i.e. *some*) of the *n* spatial objects.

With condition 1, this would form a minimal, mutually exclusive, and complete set of conditions, since condition 3 could then be expressed as the negation of the disjunction of the two other conditions.

However, the conditions are formulated with reference to conceptual modeling with simplicity and ease of modeling as a priority. It is more intuitive to model the constraint *some* as *at least one* as evidenced by common usage in natural language. If required, the constraint *at least one but not all* can still be expressed as *some* $\wedge$ ($\neg$ *all*) using the conjunction of condition 2 with the negation of condition 1. Analogously, although not strictly required, condition 3 is included because it is more natural to model this constraint directly rather than as a negation of condition 2. Therefore, the set of conditions 1, 2, and 3 as originally formulated are used as the basis for defining modeling constructs to describe n-ary topological relationships. These constructs are defined formally as follows.

We first describe the notation used for the spatial extents and their topological relationships:

Let $O \overset{\text{def}}{=} \{ o_1,\ldots,o_i,\ldots,o_j,\ldots,o_n \}$

$\overset{\text{def}}{=}$ a finite set of n spatial extents, where $n \geq 2$ and $i \neq j$.

Let R be a topological expression $\stackrel{\text{def}}{=}$:

(a) one of the binary topological relationships from Section 6.4, or

(b) a disjunction and/or conjunction of binary topological relationships from Section 6.4, or

(c) one of the adverbs *mostly*, $mostly_{rev}$, *completely*, *partially*, *occasionally*, *entirely*, or *never* from [Clar00] with (a) or (b)

(d) a disjunction and/or conjunction of (c).

Let $S \subseteq O$ (a non-empty sub-set of O)

$\stackrel{\text{def}}{=} \{ s_1,\ldots,s_k,\ldots,s_p \}$

$\stackrel{\text{def}}{=}$ a set of p spatial extents, where $p \geq 1$ and $p \leq n-2$.

We then define the following modeling constructs for describing n-ary topological relationships, assuming $i \neq j$ and $S \subseteq O$ as described above.

$$all(R, O) \stackrel{\text{def}}{=} \forall\, o_i, o_j \in O \ (o_i \text{ R } o_j)$$

$$some(R, O) \stackrel{\text{def}}{=} \exists\, o_i, o_j \in O \ (o_i \text{ R } o_j)$$

$$none(R, O) \stackrel{\text{def}}{=} \neg \exists\, o_i, o_j \in O \ (o_i \text{ R } o_j)$$

$$linked(R, O) \stackrel{\text{def}}{=} \forall\, o_i, o_j \in O \ ((o_i \text{ R } o_j) \vee (\exists S\, ((o_i \text{ R } s_1) \wedge (s_1 \text{ R } s_2) \wedge \ldots \wedge (s_{k-1} \text{ R } s_k) \wedge \ldots \wedge (s_p \text{ R } o_j))))$$

The first three constructs are based on the three conditions discussed earlier. The last construct, *linked*, describes a special case of *some* where any two spatial extents in the set can be related directly or indirectly by the given topological expression. This concept can be illustrated using the example of the *connected* relationship. Constraining a set of spatial extents to be *linked* for the *connected* relationship means that for any two spatial extents in the set, either (*i*) they are directly connected, or (*ii*) they can be indirectly connected by some sequence of intermediate spatial extents from the set.

Note that the definition of *O* excludes sets of spatial extents having zero members or one member. If *O* is empty or has only one member, then *all*, *some*, *none*, and *linked* are defined to be *true* for all *R*. If *O* has two members, then $all \Leftrightarrow some \Leftrightarrow linked$ for all symmetric *R*.

These modeling constructs allow specification of general topological relationships between the spatial extents, whether simple or composite, of *n* spatial objects. With the adverbs from Claramunt [Clar00], the same modeling constructs allow specification of topological relationships between components of pairs of *n* different composite spatial extents.

There may be some cases where we want to treat a set of composite spatial extents as a set of their individual components. This could be used to model topological constraints between all the individual components (*i*) of a composite object or (*ii*) of a set of composite spatial extents without any reference to the original composite configurations. To do this, an additional modeling construct that decomposes a set of spatial extents into the set of all their individual components is defined. That is, given a set *O* of *m* composite spatial extents $o_1,\ldots,o_i,\ldots,o_m$ with $n_1,\ldots,n_i,\ldots,n_m$ components respectively and where c_{ik} is the *k*th component of the *i*th composite spatial extent o_i, we define the following:

$$decompose(O) \stackrel{\text{def}}{=} \{\ldots,c_{ik},\ldots\} \text{ where } 1 \le i \le m \text{ and } 1 \le k \le n_i$$

We can then use any of the previously defined constructs for n-ary topological relationships, replacing *O* with *decompose*(*O*).

For example, consider the case of a national road network, with the entities being individual roads with spatial extents describing their location and geometry. Although a single road usually is a simple polyline, there may be cases where a road may consist of several disconnected segments. For instance, consider a long-distance road that is a freeway for most of the distance, but has a few segments inherited from local road networks that have different names, are not freeways, and may not even be administered by the same transport authority, as illustrated in Figure 6.6.

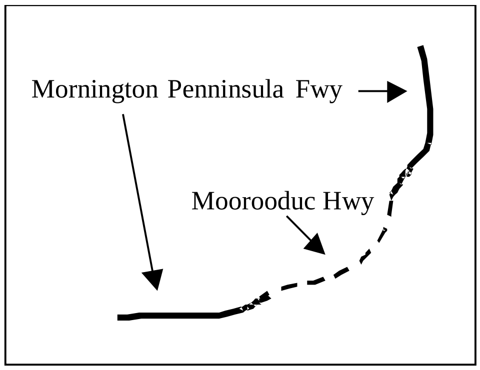


Figure 6.6 Disconnected Road Segments

When modeling the national road network, we want to enforce the constraint that the road network as a whole must be continuous, i.e. no part of the road network is isolated from the rest of the network. Since a road can have a composite spatial extent consisting of disconnected segments, this means that there must be some way to travel between every two segments of road in the network. In order to evaluate topological relationships between road segments rather than roads, the *decompose* operator is used to refer to individual road segments. The *connected* binary topological operator is used to compare pairs of road segments. The *linked* relation is then used to specify that it must be possible to find a finite sequence of connected pairs linking any two road segments. Assuming that we have the set of roads $r_1,\ldots,r_n$ in the road network, this constraint would be formally specified as:

$$linked(connected, decompose(\{r_1,\ldots,r_n\}))$$

6.6 Topological Constraints on Spatial PW Relationships

A classification framework for spatial PW relationships was described in Chapter 5. The binary and n-ary topological modeling techniques defined in Sections 6.4 and 6.5 respectively can now be used to refine that framework and illustrate its use. Specifically, the application of these techniques to defining sub-categories of *spatial constraint relationships* based on whole-part topology (one of the primary characteristics of spatial constraint relationships discussed in 5.6) and to specifying variants of *spatial PW relationships* based on inter-part topology (one of the secondary characteristics discussed in Section 5.5.3 and 5.6.4 for spatial derivation and spatial constraint relationships respectively) are discussed in Section 6.6.1 and 6.6.2 respectively. Extensive application examples are used in both sections to illustrate the utility of this approach. Based on these examples, Figure 6.7 summarizes the role topology plays in modeling spatial PW relationships.

6.6.1 Specifying Part-Whole Topology

Binary topological relationships can be used to describe *spatial constraint relationships* based on whole-part topology, as illustrated in the top portion of Figure 6.7. The specific binary topological constraint between the *part-union* and the *whole* is indicated in bold type for each sub-category. Using the binary topological relationships defined in Section 6.4, we can see that in *spatial inclusion*, *spatial cover,* and *spatial equal* the relationship of the part-union with the whole is respectively *inside or equals*, *contains* or *equals*, and *equals*. Note that these three categories of spatial constraint relationships are not mutually exclusive, since the *spatial equal* constraint is just a more restrictive version of the *spatial inclusion* and *spatial cover* constraints. This is illustrated graphically in Figure 6.7.

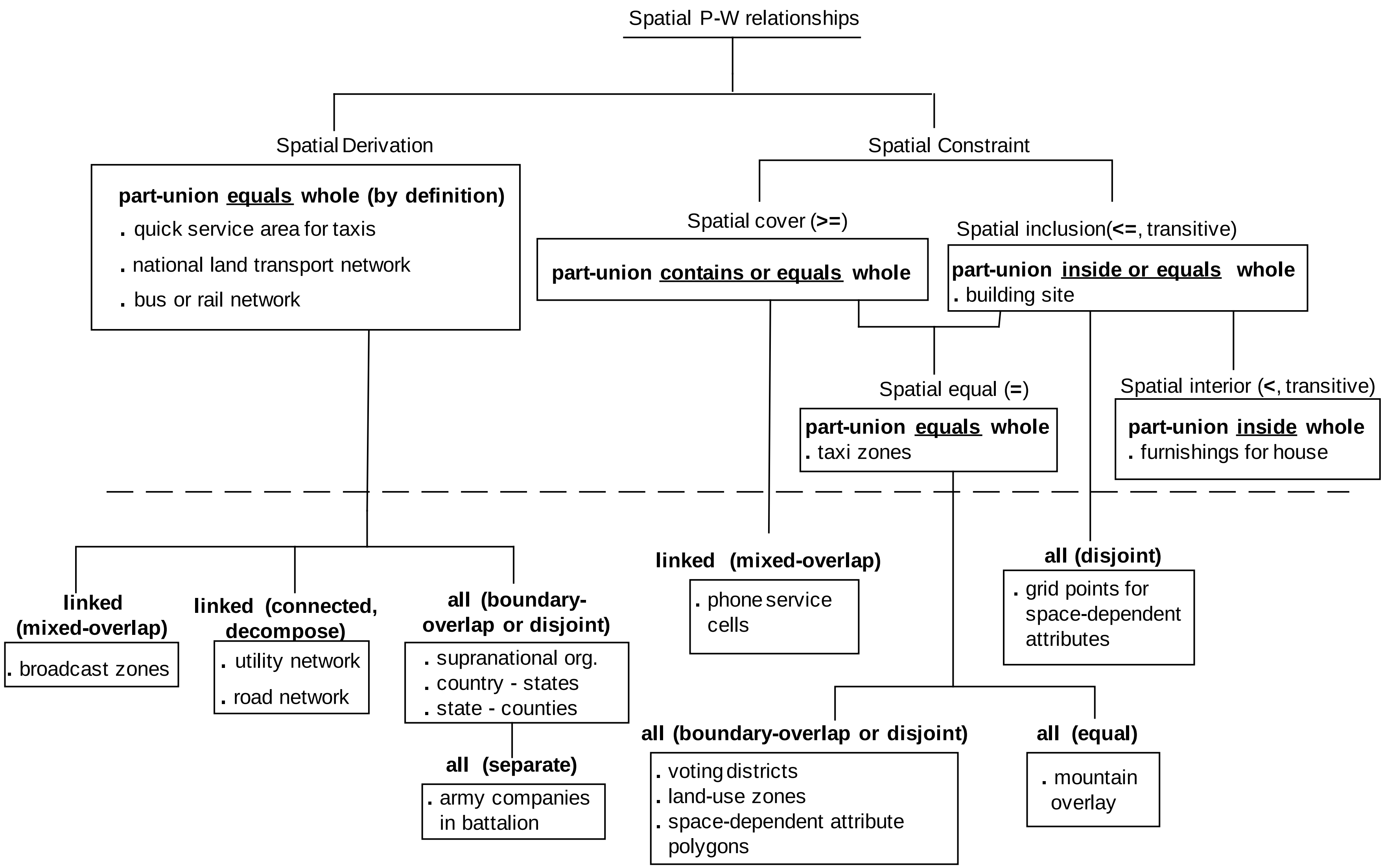


Figure 6.7 Topological Constraints in Spatial Part-Whole Relationships

If individual applications involve further restrictions on the legal topology of the whole and part-union than those defined for the three basic spatial constraint types described in Chapter 5, the binary topological relationships defined in Section 6.4 can be used to specify the additional constraints. For instance, consider the case where furnishings are required to be strictly inside a house. A new *spatial interior* category can be defined with the primary characteristic that spatial extent of the part-union must be *inside* that of the whole. This is a more restrictive constraint than that of *spatial inclusion*, as illustrated graphically in Figure 6.7.

It is important to consider whether the new sub-category has any additional non-spatial characteristics, as true with spatial inclusion. In the case of spatial inclusion, the constraint (*inside* or *equals*) between the part-union and the whole can be equivalently expressed as a constraint (*inside* or *equals*) between each part and the whole individually. This is the reason why the inclusion constraint is always transitive, e.g. any sub-components of a structure located on a building site are also located on that building site. The spatial interior constraint is transitive since it can also be expressed in terms of individual parts.

One or more application examples for each spatial constraint sub-category are given in Figure 6.7. As discussed in Section 5.1, *Spatial cover* is exemplified by a guaranteed phone service coverage area that must be completely covered by the GU of the phone service cells' spatial extents. That is, the combined phone service range *covers* (contains or equals) the guaranteed phone service area. This example is discussed further in Section 6.6.2, in the context of n-ary topological constraints. A building site and the structures on that building site represent an example of *spatial inclusion*, since no structure can extend outside the building site (also discussed in Section 5.1). The stricter constraint of *spatial interior* applies to house furnishings (referring here to appliances and furniture), since the furnishings must be inside but cannot completely cover the area of the house in order to ensure walking room. Note that if furnishings included carpets and lamp fixtures, then this would still be true, since then we would be dealing with 3D rather than 2D space and cannot completely fill the house volume in order to allow movement through the house! Finally, the GU of taxi dispatch zones (the area over which a given taxi driver ranges) must be exactly equal to the metropolitan area covered by the taxi company, i.e. *spatial equal*. This ensures complete coverage of the metropolitan area without risking cases outside the specified coverage area where the company insurance policy may not be applicable. Other examples of *spatial equal* are voting districts and land-use zones. They generally represent a deliberate sub-division of pre-existing land area for specific purposes. These two examples, and further examples of spatial constraint relationships shown in the bottom portion of Figure 6.7, are discussed in Section 6.6.2 in the context of n-ary topological relationships.

Spatial derivation is also included in the top portion of Figure 6.7 for completeness, although the topological relationship between the part-union and whole is equal by definition (since the whole is derived from the part-union) rather than constraint. The sub-categories of spatial derivation, spatial part and spatial membership, are not relevant to the current discussion (and therefore not shown in Figure 6.7) since their distinction is based on non-spatial characteristics. One example of spatial derivation is that of a quick service area for taxis. An area that can be serviced immediately (e.g. within 5 minutes) is defined for each taxi based on its current location. The total quick service area for the taxi company is then the union of those quick service areas for its taxis. Another example of spatial derivation (discussed in Section 5.1) is the national land transport network, composed of separate road, bus, and rail networks. Further examples of spatial derivation relationships shown in the bottom portion of Figure 6.7 are discussed in Section 6.6.2 in the context of n-ary topological relationships.

6.6.2 Specifying Inter-Part Topology

In the same way that binary topological relationships are used to describe whole-part topology, n-ary topological relationships can be used to describe part-part topology in spatial derivation and spatial constraint relationships. In particular, the n-ary topological relationships proposed in Section 6.5 can be used to specify variants of the basic spatial PW relationship types based on additional topological constraints between parts. This is illustrated by the examples in the bottom portion of Figure 6.7. The n-ary topological constraint applicable to a specific example is indicated in bold type. The second argument of the n-ary topological constraint (the set of spatial extents) is omitted in Figure 6.7 and in the following discussion on examples of n-ary topological constraints for the sake of readability.

Application examples can be used to illustrate the use of the modeling constructs defined in Section 6.5 for n-ary topological constraints for specifying variants of the basic spatial PW relationship categories based on the secondary characteristic of inter-part topology. To ensure that the placement of a set of broadcasting transmitters results in continuous broadcasting coverage across the set of transmitters, the *linked*(*mixed-overlap*) constraint is used to specify that there is a sequence of overlapping broadcast zones—each a simple, bounded spatial extent. Constraints on a set of phone service cells are similar to those on broadcast zones, except with the additional constraint that every point in a given area have phone service—that is, be contained in some phone service cell. This is reflected in the use of the *spatial cover* modeling construct instead of the spatial derivation construct, where the *whole* represents the area to be completely covered (i.e. coverage area) by the GU of the *parts* (i.e. the phone service cells).

In the road network and utility network application examples where a given road or utility component can have a composite spatial extent, *decompose* is used with the *linked*(*connected*) constraint to ensure a continuous road network and utility network (i.e. the *whole*) with no isolated components (i.e. the *parts*). Note that although individual roads may not be continuous (see Section 6.5), the overall road network should be for any given road network (assuming that the roads on any isolated land mass or island form a separate road network within the national land transport network). In contrast, bus and rail networks may not be continuous; however, individual bus or rail routes should be. The national land transport network is not necessarily continuous overall, since the only access to isolated islands may be by air or ship.

Since boundaries are used to uniquely partition administrative responsibility, there cannot be cases of overlapping interior points for member countries of a supranational organization, states in a country, counties in a state, or voting districts in an administrative region. Similarly, shared interior points would be inconsistent in the case of land-use zones and space-dependent attribute polygons, used to represent differences in permissible land usage or an observed attribute values respectively. The modeling construct *all*(*boundary-overlap or disjoint*) is used to specify the constraint that interior points cannot be shared between parts in a spatial PW relationship.

In the case of army companies, they must be disjoint and spread out (i.e. not interpenetrating so *all*(*separate*)) for strategic reasons and to reduce the risk of friendly fire. Sample points used to measure space-dependent attributes should be spread out (i.e. disjoint) to improve the estimation (i.e. sampling) accuracy within the sampling area. In this case, disjoint points are necessarily separate, so there is no need to specify this constraint explicitly.

Finally, we have the case of overlays for different thematic attributes over a given region (e.g. mountain vegetation, hydrography, and elevation) in a geographic application as illustrated in Figure 5.8 and described in Section 5.6.4. In this case, the constraint *all*(*equal*) is used to specify that the spatial extents of overlays must be equal.

The modeling constructs defined in this chapter and illustrated in these examples can be used in conjunction with STUML or other conceptual modeling languages to provide a clear, consistent method of specifying additional topological constraints (i.e. beyond those already defined in the spatial PW relationship classification framework from Figure 5.2 in Section 5.4) for spatial PW relationships. For example, Figure 5.9 in Section 5.7 illustrates the specification of a spatial PW relationship (spatial membership) between a supranational organization and its member countries in a STUML schema. As discussed above, member countries cannot share interior points. This can be specified in the STUML schema shown in Figure 5.9 by adding the constraint *all*(*boundary-overlap or disjoint, Countries*) to the specification box describing the spatial membership relationship (Specification Box *SM*) in the *Supranational Organization* class. In UML, the same constraint could be included in curly braces (used to indicate a constraint in UML as discussed in Section 3.2) on an association link between *Supranational Organization* and *Country* classes.

6.7 Summary

In this chapter, formal techniques for modeling binary and n-ary topological constraints during requirements analysis and conceptual design of spatial applications have been presented that are of general applicability in the context of composite spatial objects and spatial PW relationships. A two-level classification scheme for describing binary topological relationships has been proposed that is general enough to be suitable for a range of different applications yet is simple to use and understand. The first level of binary topological relationships—*disjoint*, *equal*, *contains*, *inside*, and *connected*—is based on intersection and difference, concepts that are intuitive and can be applied in a simple and consistent manner regardless of the complexity or dimension of the spatial objects considered. The second level represents a refinement of the disjoint and connected categories based on application dependent definitions of minimum bounding figures and boundary versus interior respectively. That is, users can select the definition best suited to their application domain as long as it conforms to certain minimal assumptions, i.e. that a given disjoint relationship cannot be both separate and interpenetrating and that a given point in a spatial extent cannot be both boundary and interior for that spatial extent.

The final set of eight topological relationships—*separate*, *interpenetrating*, *equal*, *contain*, *inside*, *boundary-overlap*, *mixed-overlap*, and *interior-overlap*—represents a complete and mutually exclusive set of binary topological relationships. The defined relationships include those between single and mixed-dimension composites, irregular, partially bounded or unbounded, and 3D spatial objects. Using this set of binary topological relationships, we then defined modeling constructs for the specification of n-ary topological relationships. Finally, we showed how the proposed techniques can be used to specify topological constraints between components of composite spatial objects and—both between parts and between the whole and parts—in spatial PW relationships. Existing conceptual modeling languages such as UML or spatiotemporal extensions based on UML [Brod00] contain provisions for general constraint specification but no specific support for describing topological constraints on spatial objects. The techniques that have been proposed here can be used in conjunction with such languages to add the necessary support for describing topological constraints between components of composite spatial objects and in spatial PW relationships.

Chapter 7

Conclusion

7.1 Contributions

This thesis has addressed the problem of providing support for conceptual analysis and design of spatiotemporal applications, focusing specifically on the requirements of geographic information systems. The stated goals were to develop modeling techniques general enough to support a wide variety of geographic applications, simple and practical enough to serve as useful aids in the early phases of application development, yet based on formal definitions to facilitate unambiguous specification and subsequent implementation.

A graphical modeling language, the SpatioTemporal Unified Modeling Language (STUML), has been proposed based on extending the object-oriented standard, the Unified Modeling Language (UML). Spatiotemporal semantics are represented using a small base set of modeling constructs—*spatial*, *temporal*, and *thematic*—that can be combined and applied to different UML constructs in an orthogonal manner, thus preserving language clarity and simplicity without sacrificing expressive power or flexibility. A consistent technique has been introduced for modeling temporal change in spatial data and spatiotemporal change in thematic or composite data. The main contributions of STUML include:

- support for representation of spatiotemporal semantics and dependencies at every level of the object-oriented STUML model (i.e. attribute, attribute group, object, association); resulting in simpler, more compact, and more reusable schemas for spatiotemporal applications as compared to equivalent UML schemas,

- explicit support for modeling common spatial and/or temporal properties in a related group of thematic attributes (i.e. alphanumeric, not spatial or temporal, attributes) through the introduction of the attribute group construct,
- the precise definition of existence time and the associated modeling construct,
- support for representing both space-dependent and time-dependent composite data,
- the introduction of a thematic construct and simple rules for combining spatial, temporal, and thematic constructs to allow consistent representation of functional composition semantics across all three domains,
- a consistent and simple mapping between graphical representation and spatiotemporal semantics, clearly distinguishing graphically between dependent and independent spatiotemporal semantics, object-based versus field-based spatial semantics, structural and integrity model components, and different levels of abstraction, and
- a formal functional specification of the semantic modeling constructs and their combinations to provide a rigorous theoretical basis for specifications.

Guidelines for mapping STUML to UML schemas have been presented, providing a theoretical basis for implementing STUML schemas using tools developed for UML or extending those tools with spatiotemporal extensions based on STUML. This provides an immediate strategy for implementing STUML by exploiting the extensive set of tools and commercial products already available for UML. This further serves as a proof of concept for STUML by demonstrating that any STUML modeling construct can be mapped to an equivalent expression in a well-established existing modeling language. A comprehensive set of transformation rules have been given to convert spatiotemporal modeling constructs from STUML to equivalent UML constructs. In the case of multiple UML mappings, the mapping providing the most general solution was selected, i.e. that does not rely on support for composite domains and that is expected to minimize redundancy in any implementation generated from the conceptual UML schema.

The specific problem of modeling the types of complex spatial objects typically found in geographic applications and the topological relationships characterizing such complex spatial objects has been addressed in this thesis. Explicit modeling constructs for different types of complex spatial objects and their topological relationships have been defined and their integration in a conceptual modeling language has been demonstrated using STUML.

The term *spatial Part-Whole (PW) relationships* has been introduced to convey the asymmetric nature of the relationship between a complex spatial object and its spatial sub-units. Five different types of spatial PW relationships that are of general utility in geographic applications have been described and a general classification framework for such relationships formally defined. The framework provides a systematic approach to specifying spatial part-whole relationship types based on both their spatial and non-spatial characteristics, providing a basis for defining both the five identified spatial PW relationship types and further spatial PW relationship types as required in a given application context. The primary contributions of the work on spatial PW relationships are:

- the identification of specific spatial PW relationship types useful in modeling geographic applications and the definition of corresponding modeling constructs that can be incorporated into a conceptual modeling language, and

- the formal definition and classification of spatial PW relationships, including the identified types, based on *both* spatial and non-spatial characteristics of the relationships.

Finally, a comprehensive method for specifying binary topological constraints between a complex spatial object and its spatial sub-units (i.e. the *whole* and its *parts*) and n-ary (set-based) topological constraints between the spatial sub-units (i.e. the *parts*) has been described and formally defined. The proposed method has the following advantages over other topological classification schemes described in the literature.

- The simplicity of the proposed method (characterized by a much lower level of complexity than other schemes proposed in the literature) makes it particularly suitable for use in early application development phases. The method is fundamentally based on the simple and intuitive concepts of intersection and difference of spatial extents and not dependent on the specific mathematical model used to describe topological concepts.
- The proposed modeling techniques cater for a wide range of irregular and composite object types required for general support of geographic applications. These include spatial object types not previously considered, such as spatial extents that are not closed; irregularities such as loops, punctures, and cuts; mixed-dimension composites (e.g. a single composite object consisting of regions, lines, and points); and 3D objects or embedding space.
- The proposed modeling techniques provide support for specification of set-based topological constraints between the spatial components of a composite spatial object and between the spatial part objects in a spatial PW relationship. Definitions of binary topological relationships between two spatial objects are

extended to describe n-ary topological relationships between *n* spatial objects, as required to specify set-based topological constraints between components in a composite spatial object or between parts in a spatial PW relationship.

7.2 Future Research

The approach taken in this thesis suggests several future areas of research in the context of providing conceptual modeling support for spatiotemporal applications. These include exploiting alternate STUML symbol nestings for physical schema and query optimization; incorporating a temporal dimension in the modeling techniques proposed for spatial PW relationships and their topology; extending set-based constraint specification to include orientation and metric constraints; and investigating the applicability of STUML to non-geographic applications. These future areas of research are discussed in Sections 7.2.1 through 7.2.4 respectively.

7.2.1 Physical Schema and Query Optimization

The different user perspectives of spatiotemporal application data represented by the different orders of nesting STUML spatial, temporal, and thematic symbols have been formally defined in the thesis. As discussed in the thesis, these alternative views may reflect distinctions in user query or access patterns, represented graphically by the specific symbol nesting selected. The different orders of nesting STUML symbols could therefore be deliberately exploited in later development phases to indicate preferred data clustering patterns or access strategies. These preferences could potentially be utilized by the database management system when generating the physical schema or by the query optimizer when selecting access strategies for individual queries based on expected query frequencies, access priorities, etc.

7.2.2 Adding Time to Spatial Part-Whole Relationships

To support spatiotemporal applications, time restrictions can be incorporated in the spatial PW relationship and topological modeling techniques proposed here. For example, spatial PW relationships may have specific restrictions on the existence time or association of parts relative to the whole or vice versa beyond that of simple dependencies, e.g. a part must be associated with a whole for a certain minimum time period. Specific constraints related to the type of PW relationship or their characteristics (whether spatial or non-spatial) may have additional time dependencies, i.e. time periods when they do or do not apply. This can be illustrated using the example of topological constraints on spatial PW relationships.

If topological constraints are specified for an application that maintains historical as well as current data, the default assumption is that these constraints *always* apply. Thus, even when the temporal dimension is not explicitly catered for, temporal semantics are implied by any constraint specified for the application. However, there may be cases where this assumption is not valid, e.g. where the constraint is valid (*i*) only since some historical point in time, (*ii*) only after some future point in time, and/or (*iii*) only until some future point in time. Such temporal restrictions may apply to individual database states, database transitions, sequences of database states, or user-defined interval relations between database states.

Single-state time restrictions applying to individual database snapshots can easily be introduced in the binary and n-ary topological relations that have been proposed in this thesis. Logic-based languages such as that described in [Bohl94, Chom94] are typically used to specify multi-state temporal constraints involving comparison of database snapshots; however, finding specification techniques simple enough to be practical for requirements analysis and conceptual design is an open question. More research is needed in this area.

7.2.3 Extending Set-Based Constraint Specification

The set-based constraint specification that has been proposed in this thesis for n-ary topological constraints can be applied to other spatial characteristics such as orientation or metrics. For example, the proposed set-based topological techniques were used in this thesis to enforce the constraint that the distance between any two road segments in a road network can be traversed through a sequence of adjacent intermediate roads. Similarly, set-based metric constraints could be used to ensure continuous phone coverage over an area by specifying that the distance between any two phone service cells can be traversed in intermediate phone service cell "hops" of less than *x* kilometers based on phone cell transmitting ranges.

7.2.4 Non-Geographic Applications

As discussed in the thesis, other non-geographic spatiotemporal applications include multimedia information retrieval systems, mobile computing, mobile telephony, and applications with non-geographic references such as molecular chemistry or medical pathology. For example, multimedia applications require the composition, cataloguing, retrieval, and display (or playback) of artifacts with both spatial and temporal dimensions, such as graphic animations, image sequences, video, or multimedia presentations. Conceptual models of the spatiotemporal properties of such artifacts facilitates their retrieval based on these properties and provide a template that can be used to synchronize their presentation in a prescribed manner. In mobile computing and telephony, maintaining a record of historical changes in the location of both the resources to be accessed and the users aids in trend analysis, performance tuning, and security management. Similarly, understanding chemical and biological processes (e.g. disease progression, anatomical development) requires that the dynamics of molecular or anatomical spatial configurations be recorded. The applicability of STUML (with the associated techniques for modeling spatial PW relationships and topology) to conceptual modeling in these application contexts is worth investigation.

References

[Abra99] Abraham, T. and J. F. Roddick. *Survey of Spatio-Temporal Databases*, GeoInformatica, Volume 3, Issue 1, 1999, 61-99.

[AlK99] Al-Khatib, W. et al. *Semantic Modeling and Knowledge Representation in Multimedia Databases*, IEEE Transactions on Knowledge and Data Engineering, Volume 11, Issue 1, 1999, 64-80.

[Allen83] Allen, James. *Maintaining Knowledge about Temporal Intervals*, Communications of the ACM, Volume 26, Issue 11, 1983, 832-843.

[AlTa93] Al-Taha, K., R.T. Snodgrass, and M.D. Soo. *Bibliography on Spatiotemporal Databases*, SIGMOD Record, Volume 22, Issue 1, 1993, 59-67.

[AlTa94] Al-Taha, K., R.T. Snodgrass, and M.D. Soo. *Bibliography on Spatiotemporal Databases*, International Journal on Geographical Information Systems, Volume 8, Number 1, 1994, 95-103.

[ANSI92] American National Standards Institute (ANSI) Document ANSI X3.135-1992, 1992.

[Arit97] Aritsugi, M., T. Amagasa, T. Tagashira, and Y. Kanamori. *An Approach to Spatio-Temporal Queries—Interval-Based Contents Representation of Images*, Proceedings of the Eighth International Workshop on Database and Expert Systems Applications, 1997, 202-213.

[Arms79] Armstrong, M. A. Basic Topology, McGraw-Hill, Maidenhead, 1979.

[Asla99] Aslandogan, Y.A. and C.T. Yu. *Techniques and Systems for Image and Video Retrieval*, IEEE Transactions on Knowledge and Data Engineering, Volume 11, Issue 1, 1999, 56-63.

[Baez99] Baeza-Yates, R. and B. Ribeiro-Neto. Modern Information Retrieval, Addison-Wesley, New York, 1999.

[Barb99] Barbara, D. *Mobile Computing and Databases—A Survey*, IEEE Transactions on Knowledge and Data Engineering, Volume 11, Issue 1, 1999, 108-117.

[Barr91] Barrera, R., A. Frank, and K. Al-Taha, editors. *Temporal Relations in Geographic Information Systems: A Workshop at the University of Maine*, SIGMOD Record, Volume 20, Issue 3, 1991, 85-91.

[Beck96] Becker, L., A. Voigtmann, and K. Hinrichs. *Temporal Support for Geo-Data in Object-Oriented Databases*, Proceedings of the Seventh International Workshop on Database and Expert Systems Applications, 1996, 79-93.

[Bedar99] Bédard, Y. *Visual Modelling of Spatial Databases: Towards Spatial PVL and UML*, Geoinformatica, Volume 53, Issue 2, 1999, 169-186.

[Belu97] Belussi, A., E. Bertino, and B. Catania. *Manipulating Spatial Data in Constraint Databases*, Proceedings of the Fifth International Symposium on Large Spatial Databases, Berlin, 1997, 115-141.

[Bohl94] Böhlen, M. H. *Valid Time Integrity Constraints*, Technical Report TR 94-30, Department of Computer Science, University of Arizona, 1994, 1-19.

[Booch99] Booch, G., J. Rumbaugh, and I. Jacobson. The Unified Modeling Language User Guide, Addison-Wesley, Reading, M.A., 1999.

[Borges99] Borges, K, A.H.F. Laender, and C.A. Davis Jr. *Spatial Data Integrity Constraints in Object Oriented Geographic Data Modeling*, Proceedings of the Seventh ACM International Workshop on Advances in Geographic Information Systems, 1999, 1-6.

[Brod00] Brodeur, J., Y. Bedard, and M. Proulx. *Modelling Geospatial Application Databases using UML-based Repositories Aligned with International Standards in Geomatics*, Proceedings of the Eighth ACM International Workshop on Advances in Geographic Information Systems, 2000, 39-46.

[Burr98] Burrough, P. A. and R. A. McDonnell. Principles of Geographical Information Systems, Oxford University Press, Oxford, 1998.

[Card93] Cardenas, A.F., I. T. Ieong, R. K. Taira, R. Barker, and C. M. Breant. *The Knowledge-Based Object-Oriented PICQUERY+ Language*, IEEE Transactions on Knowledge and Data Engineering, Volume 5, Issue 4, 1993, 644-657.

[Chom94] Chomicki, J. *Temporal Integrity Constraints in Relational Databases*, Bulletin of the Technical Committee on Data Engineering, Volume 17, Issue 2, 1994, 33-37.

[Chom97] Chomicki, J. and P.Z. Revesz. *Constraint-Based Interoperability of Spatiotemporal Databases*, Proceedings of the Fifth International Symposium on Large Spatial Databases, Berlin, 1997, 142-161.

[Chris95] Christodoulakis, S. and L. Koveos. *Multimedia Information Systems: Issues and Approaches The Object Model, Interoperability, and Beyond*, Chapter 16 in Modern Database Systems, editor Won Kim, Addison-Wesley, Reading, Mass., 1995, 318-337.

[Clar95] Claramunt, C. *Managing Time in GIS An Event-Oriented Approach*, Proceedings of International Workshop on Temporal Databases, 1995, 23-42.

[Clar00] Claramunt, C. *Extending Ladkin's Algebra on Non-convex Intervals towards an Algebra on Union-of Regions,* Proceedings of the Eighth ACM International Workshop on Advances in Geographic Information Systems, 2000, 9-14.

[Clem93] Clementini, E., P. Di Felice, and P. Oosterom. *A Small Set of Formal Topological Relationships Suitable for End-User Interaction*, Proceedings of the Third International Symposium on Large Spatial Database Systems, Lecture Notes in Computer Science, Volume 692, Springer-Verlag, Singapore, 1993, 277-295.

[Clem94] Clementini, E. and P. Di Felice. *A Model for Representing Topological Relationships between Complex Geometric Features in Spatial Databases*, Information Sciences, 1994, 1-17.

[Clem95a] Clementini, E., P. Di Felice, and G. Califano. *Composite Regions in Topological Queries*, Information Systems, Volume 20, Issue 7, 1995, 579-594.

[Clem95b] Clementini, E. and P. Di Felice. *A Comparison of Methods for Representing Topological Relationships,* Information Sciences, Volume 3, 1995, 149-178.

[Cock97] Cockcroft, S. *A Taxonomy of Spatial Data Integrity Constraints*, GeoInformatica, Volume 1, Issue 4, 1997, 327-243.

[Conn02] Connolly, T. M. and C. E. Begg. Database Systems A Practical Approach to Design, Implementation, and Management, Third Edition, Addison-Wesley, Harlow, England, 2002.

[Dagt98] Dagtas, S. et al. *A Hybrid Content-Based Retrieval Approach for Video Data*, Technical Report TR-ECE 98-13, Purdue Univ., 1998.

[Day95] Day, Young Francis, S. Dagtas, M. Iino, A. Khokhar, and A. Ghafoor. *Object-Oriented Conceptual Modeling of Video Data*, Proceedings of the Eleventh International Conference on Data Engineering, 1995, 401-408

[DelB95] Del Bimbo, A., E. Vicario, and D. Zingoni. *Symbolic Description and Visual Querying of Image Sequences Using Spatio-Temporal Logic,* IEEE Transactions on Knowledge and Data Engineering, Volume 7, Issue 4, 1995, 609-621.

[DelB96] Del Bimbo, A. and E. Vicario. *A Logical Framework for SpatioTemporal Indexing of Image Sequences*, Intelligent Image Database Systems, editors S.K. Chang, E. Jungert, and G. Tortora, World Scientific, Singapore, 1996, 165-196.

[Dimit97] Dimitrova, N.and F. Golshani. *Video and Image Content Representation and Retrieval*, in Chapter 4 in The Handbook of Multimedia Information Management, editors W. I. Grosky, R. Jain, and R. Mehrotra, Prentice Hall PTR (ECS Professional), New Jersey, 1997, 95-138.

[Dion98] Dionisio, J.D.N. and A.F. Cardenas. *A Unified Data Model for Representing Multimedia, Timeline, and Simulation Data*, IEEE Transactions on Knowledge and Data Engineering, Volume 10, Issue 5, Sept/Oct 1998, 746-767.

[Egen90] Egenhofer, M. and J.R. Herring. *A Mathematical Framework for the Definitions of Topological Relationships*, Proceedings of the Fourth International Symposium on Spatial Data Handling, 1990, 803-813.

[Egen91a] Egenhofer, M. and J.R. Herring. *Categorizing Binary Topological Relationships Between Regions, Lines, and Points in Geographic Databases*, Technical Report, Department of Surveying Engineering, University of Maine, Orono, ME, 1991, 1-33.

[Egen91b] Egenhofer, M. and R. Franzosa. *Point-set Topological Spatial Relations*, International. Journal of Geographical Information Systems, Volume 5, Number 2, 1991, 161-174.

[Egen92] Egenhofer, M.J. and K.K. Al-Taha. *Reasoning about Gradual Changes of Topological Relationships*, Theories and Methods of Spatiotemporal Reasoning in Geographic Space, editor A. Frank, Lecture Notes in Computer Science, Volume 639, Springer-Verlag, 1992, 196-219.

[Egen94a] Egenhofer, M. *Spatial SQL: a Query and Presentation Language*, IEEE Transactions on Knowledge and Data Engineering, Volume 6, Issue 1, 1994, 86-95.

[Egen94b] Egenhofer, M, E. Clementini, and P. Di Felice. *Topological Relations between Regions with Holes*, International Journal of Geographical Information Systems, Volume 8, Number 2, 1994, 129-142.

[Erwig99] Erwig, M., R. H. Güting, M. Schneider, and M. Vazirgiannis. *Spatio-Temporal Data Types: An Approach to Modeling and Querying Moving Objects in Databases*, GeoInformatica, Volume 3, Issue 3, 1999, 269-296.

[Faria98] Faria, G., C. B. Medeiros, and M.A. Nascimento. *An Extensible Framework for Spatio-Temporal Database Applications*, Time Center Technical Report TR-27, 1998, 1-15.

[Form94] Forman, G. and J. Zahorjan. *The Challenges of Mobile Computing*, University of Washington Computer Science Engineering Technical Report# 93-11-03, 1994, 1-16.

[Fowl97a] Fowler, M. and K. Scott. UML Distilled, Addison-Wesley, Reading, M.A., 1997.

[Fowl97b] Fowler, M. Analysis Patterns: Reusable Object Models, Addison-Wesley, Menlo Park, C.A., 1997.

[Gibbs97] Gibbs, S, C. Breiteneder, and D. Tsichritzis. *Modeling Time-Based Media*, Chapter 1 in The Handbook of Multimedia Information Management, editors W. I. Grosky, R. Jain, and R. Mehrotra, Prentice Hall PTR (ECS Professional), New Jersey, 1997, 13-38.

[Gibl77] Giblin, P. J. Graphs, Surfaces and Homology—An Introduction to Algebraic Topology, Chapman and Hall, London, 1977.

[Golsh94] Golshani, F. and N. Dimitrova, *Retrieval and Delivery of Information in Multimedia Database Systems*, Information and Software Technology, Volume 36, Issue 4, 1994, 235-242.

[Gord97] Gordillo, S., F. Balaguer., F. Das Neves. *Generating the Architecture of GIS Applications with Design Patterns*,, Proceedings of the Fifth ACM International Workshop on Advances in Geographic Information Systems, 1997, 30-34.

[Gord98] Gordillo, S. and F. Balaguer. *Refining an Object-Oriented GIS Design Model: Topologies and Field Data*, Proceedings of the Sixth ACM International Workshop on Advances in Geographic Information Systems, 1998, 76-81.

[Greg99] Gregersen, H. and C.S. Jensen. *Temporal Entity-Relationships Models—a Survey*, IEEE Transactions on Knowledge and Data Engineering, Volume 11, Issue 3, 1999, 464-497.

[Gros97] Grosky, W.I., R. Jain, and R. Mehrotra, editors. The Handbook of Multimedia Information Management, Prentice Hall PTR (ECS Professional), New Jersey, 1997.

[Guti91] Güting, R.H. *Geo-Relational Algebra: A Model and Query Language for Geometric Database Systems*, Proceedings of the International Conference. On Extending Database Technology, 1991, 506-527.

[Guti98] Güting, R. H., M. H. Böhlen, M. Erwig, C. S. Jensen, N. A. Lorentzos, M. Schneider, and M. Vazirgiannis. *A Foundation for Representing and Querying Moving Objects*, FernUniversitat Hagen, Informatik-Report 238, 1998, 1-49.

[Hadz92] Hadzilacos, T. and N. Tryfona. *A Model for Expressing Topological Integrity Constraints in Geographic Databases*, Theories and Methods of Spatio-Temporal Reasoning in Geographic Space, Springer-Verlag, Berlin, 1992, 252-268.

[Harar69] Harary, F. Graph Theory, Addison-Wesley, Reading, MA, 1969.

[Haze91] Hazelton, N.W.J. *Integrating Time, Dynamic Modelling and Geographical Information Systems: Development of Four-Dimensional GIS*, Ph.D. Dissertation, Department of Surveying and Land Information, University of Melbourne, 1991.

[Hend98] Henderson-Sellers, B. *OML: Proposals to Enhance UML*, The Unified Modeling Language, UML'98: Beyond the Notation, editors J. Bezivin and P.A. Muller, Springer-Verlag, Berlin, 1998, 349-364.

[Hend99a] Henderson-Sellers, B. and F. Barbier. *What Is This Thing Called Aggregation?*, Proceedings of the International Conference on Technology of Object-Oriented Languages and Systems (TOOLS) EUROPE, 1999, 216-230.

[Hend99b] Henderson-Sellers, B. and F. Barbier. *Black and White Diamonds*, Proceedings of the International Conference on Unified Modeling Language, 1999, 550-565.

[Imiel93] Imielinski, T. and B.R. Badrinath. *Data Management for Mobile Computing*, SIGMOD Record, Volume 22, Issue 1, 1993, 34-39.

[Jähne93] Jähne, Bernd. Spatio-Temporal Image Processing. Theory and Scientific Applications, Springer-Verlag, Berlin, 1993.

[Jain94] Jain, R. *Metadata in Video Databases*, SIGMOD Record, Volume 23, Issue 4, 1994, 27-33.

[Jens98] Jensen, C. and C. E. Dyreson, editors *The Consensus Glossary of Temporal Database Concepts—Feb 1998 Version*. editors O. Etzion, et al, Temporal Databases: Research and Practice, Lecture Notes in Computer Science, Volume 1399, Springer-Verlag, 1998, 367-405.

[Kilo94] Kilov, H. and J. Ross. Information Modeling, An Object-Oriented Approach, Prentice Hall, 1994.

[Lang88] Langran, G. *Temporal GIS Design Tradeoffs*. Proceedings of the Third International Conference of Geographic Information Systems/Land Information Systems (GIS/LIS), Volume 2, 1988, 890-899.

[Lang89] Langran, G. *A Review of Temporal Database Research and its use in GIS Applications*, International Journal of Geographical Information Systems, Volume 3, Number 3, 1989, 214-232.

[Lang93] Langran, G. Time in Geographic Information Systems, Taylor & Francis, London, 1993.

[Leban86] Leban, B., D. McDonald, and D. Forster. *A Representation for Collections of Temporal Intervals*, Proceedings of the Fifth National Conference on Artificial Intelligence, Volume 1, 1986, 367-371.

[Little93] Little, T.D.C. and A. Ghafoor. *Interval-Based Conceptual Models for Time-Dependent Multimedia Data*, IEEE Transactions on Knowledge and Data Engineering, Volume 5, Issue 4, 1993, 551-563.

[MacD91] MacDougall, E. B. *Dynamic Statistical Visualization of Geographic Information Systems*, Proceedings of the International Conference on Geographic Information Systems/Land Information Systems, Volume 1, 1991, 158-165

[Mitas95] Mitasova, H., L. Mitas, W. M. Brown, D. P. Gerdes, I. Kosinovsky, and T. Baker. *Modelling Spatially and Temporally Distributed Phenomena: New Methods and Tools for GRASS GIS*, International Journal of Geographical Information Systems, Volume 9, Number 4, 1995, 443-446.

[Mots99] Motschnig-Pitrik, R. and J. Kaasboll. *Part-Whole Relationship Categories and Their Application in Object-Oriented Analysis*, IEEE

Transactions on Knowledge and Data Engineering, Volume 11, Issue 5, 1999, 779-797.

[Odell94] J.J. Odell. *Six Different Kinds of Composition*, Journal of Object-Oriented Programming, Volume 6, Issue 8, 1994, 10-15.

[OMG99] OMG Unified Modeling Language Specifications, version 1.3. Object Management Group, Needham, M.A., Retrieved May 19, 2000 from World Wide Web: http://www.ftp.omg.org/pub/docs/ad/99-06-08.pdf, 1999, 1-800.

[Oomo97] Oomoto, E. and K. Tanaka. *Video Database Systems—Recent Trends in Research and Development Activities*, Chapter 13 in The Handbook of Multimedia Information Management, editors W. I. Grosky, R. Jain, and R. Mehrotra, Prentice Hall PTR (ECS Professional), New Jersey, 1997, 405-448.

[Paren98] Parent, C., S. Spaccapietra, E. Zimanyi, P. Donini, C. Plazanet, and C. Vangenot. *Modeling Spatial Data in the MADS Conceptual Model*, Proceedings of the Eighth International Symposium on Spatial Data Handling, 1998, 138-150.

[Paren99] Parent, C., S. Spaccapietra, and E. Zimanyi. *Spatio-Temporal Conceptual Models: Data Structures + Space + Time*, Proceedings of the Seventh ACM Symposium on Advances in Geographic Information Systems, Nov 5-6, 1999, 26-33.

[Pavl98] Pavlopoulos, A. and B. Theodoulidis. *Review of SpatioTemporal Data Models*. TimeLab Technical Report TR-98-3, 1998.

[Pequ95] Pequet, D. J. and Duan. N. *An Event-Based Spatiotemporal Data Model (ESTDM) for Temporal Analysis of Geographical Data*, International Journal of Geographical Information Systems, Volume 9, Number 1, 1995, 7-24.

[Price91] Price, R. *Multimedia Information Systems*, Chapter 5 in Multimedia: Technology and Applications, editor J. Waterworth, Ellis Horwood, Chichester, UK, 1991, 114-150.

[Price99a] Price, R., B. Srinivasan, and K. Ramamohanarao. *Extending the Unified Modeling Language to Support Spatiotemporal Applications*, Proceedings of the Tenth International Conference on Technology of Object Oriented Languages and Systems (TOOLS) PACIFIC, 1999, 163-174.

[Price99b] Price, R, K. Ramamohanarao, B. Srinivasan. *Spatiotemporal Extensions to Unified Modeling Language*, Proceedings of the Tenth International Workshop on Database and Expert Systems Applications, 1999, 460-461.

[Price99c] Price, R., K. Ramamohanarao, B. Srinivasan. *A Spatiotemporal Modeling Language for Multimedia Applications*, Advances in Intelligent Computing and Multimedia Systems, editors S. M. Rahman, O. R. Baiocchi, and G. E. Lasker, International Institute for Advanced Studies in Systems Research and Cybernetics, Windsor, Ontario, Canada, 1999, 125-130.

[Price00a] Price, R., N. Tryfona, and C. S. Jensen. *Modeling Part-Whole Relationships for Spatial Data*, Proceedings of the Eighth ACM International Workshop on Advances in Geographic Information Systems, 2000, 1-8.

[Price00b] Price, R., N. Tryfona, and C.S. Jensen. *Extended Spatiotemporal UML: Motivations, Requirements, and Constructs*, Journal of Database Management, Volume 11, Issue 4, Oct-Dec 2000, 14-27.

[Price01a] Price, R., N. Tryfona, and C.S. Jensen. *Modeling Topological Constraints in Spatial Part-Whole Relationships*, Proceedings of the 20th International Conference on Conceptual Modeling (ER2001), Yokohama, Japan, 2001, in press.

[Price01b] Price, R., B. Srinivasan, and K. Ramamohanarao. *Issues in Spatiotemporal Data Modeling*, Design and Management of Multimedia Information Systems: Opportunities and Challenges, editor S. M. Rahman, Idea Group Publishing, Hershey, PA., 2001, 412-433.

[Quat98] Quatrani, T. Visual Modeling with Rational Rose and UML, Addison-Wesley, Reading, Massachusetts, 1998.

[Ratio98a] Using Rational Rose, Rational Software Corporation, 1998.

[Ratio98b] Using Rational Rose98/Oracle8, Rational Software Corporation, 1998.

[Rumb99] Rumbaugh, J, I. Jacobson, and G. Booch. The Unified Modeling Language Reference Manual, Addison-Wesley, Reading, M.A., 1999.

[Saks98] Saksena, M. et al. *Extending Aggregation Constructs in UML*, The Unified Modeling Language, UML'98: Beyond the Notation, Springer-Verlag, Berlin, 1998, 434-441.

[Salt83] Salton, G. and M. J. McGill. Introduction to Information Retrieval, McGraw-Hill, New York, 1983.

[Salt89] Salton, G. Automatic Text Processing, Addison-Wesley Publishing Co., 1989.

[Schw98] Schwalb, E. and L. Vila. *Temporal Constraints: A Survey*, Constraints: An International Journal, Volume 2, 1998, 129-149.

[Sloc93] Slocum, T., J. C. Davis, and S. L. Egbert. *Developing Software for Exploring Temporal Spatial Data*, Proceedings of the International Conference on Geographic Information Systems/Land Information Systems, Volume 2, 1993, 642-653.

[Suth75] Sutherland, W. A. Introduction to Metric and Topological Spaces, Clarendon Press, Oxford, 1975.

[Tans93] Tansel, A.U., J. Clifford, S. Gadia, S. Jojodia, A. Segev, and R. Snodgrass. Temporal Databases Theory, Design, and Implementation, Benjamin/Cummings Publishing Company, Inc., Redwood City, California, 1993.

[Timpf99] Timpf, S. *Abstraction, Levels of Detail, and Hierarchies in Map Series*, Proceedings of International Conference on Spatial Information Theory

(COSIT'99), editors C. Freksa and D. M. Mark, Springer-Verlag, Berlin, 1999, 125-139.

[Tryf97a] Tryfona, N. and M.J. Egenhofer. *Consistency among parts and aggregates: A computational model*, Transactions in Geographic Information Systems, Volume 1, Issue 3, 1997, 189-206.

[Tryf97b] Tryfona, N., D. Pfoser, and T. Hadzilacos. *Modeling Behavior of Geographic Objects: An Experience with the Object Modeling Technique*, Proceedings of the Ninth International Conference CAiSE'97, editors A. Olive and J. A. Pastor, Lecture Notes in Computer Science, Volume1250, Springer-Verlag, Berlin, 1997, 347-359.

[Tryf99] Tryfona, N. and C. Jensen . *Conceptual Data Modeling for Spatiotemporal Applications*, GeoInformatica, Volume 3, Issue 3, 1999, 245-268.

[Wins87] Winston, M.E., R. Chaffin, and D. Herrmann. *A Taxonomy of Part-Whole Relations,* Cognitive Science, Volume 11, 1987, 417-444.

[Worb92] Worboys, Micheal. *Object-Oriented Models of Spatiotemporal Information*, Proceedings of the International Conference on Geographic Information Systems/Land Information Systems, Volume 2, 1992, 825-834.

[Worb95] Worboys, Michael. GIS A Computing Perspective, Taylor & Francis, London, 1995.

[Xiao89] Xiao, Q. and H. Raafat. *A Temporal/Spatial Database Structure for Remotely Sensed Image Data Management within GIS*, Proceedings of the Fourth International Conference on Geographic Information Systems/Land Information Systems (GIS/LIS), Volume 1, 1989, 116-122.

[Yang96] Yang, H. K., M. S. Kim, S. H. Kwon, and C. S. Jeong. *The Object-Oriented Design of KROSS: An Object-Oriented Spatial Database System*, Proceedings of the Seventh International Conference of Database and Expert System Applications, 1996, 603-612.

[Yeh92] Yeh, T.S. and Y.H. Viemont. *Temporal Aspects of Geographical Databases*, Proceedings of the Third International Conference on European Geographic Information Systems, Volume 1, 1992, 320-328.

[Yosh99] Yoshitaka, A. and T. Ichikawa. *A Survey on Content-Based Retrieval for Multimedia Databases*, IEEE Transactions on Knowledge and Data Engineering, Volume 11, Issue 1, 1999, 81-93.

[Zani97] Zaniolo, Carlo et al. Advanced Database Systems, Morgan Kaufmann Publishers, S.F., CA., 1997.

[Zetts97] Zettsu, K., K. Uehara, K. Tanaka, and N. Kimura. *A Time-Stamped Authoring Graph for Video Databases*, Proceedings of the Eighth International Workshop on Database and Expert Systems Applications, 1997, 192-201.

[Zhil00] Zhilin, Li, Li Yongli and Chen Yong-Qi. *Basic Topological Models for Spatial Entities in 3-Dimensional Space*, GeoInformatica, Volume 4, Issue 4, 2000, 419-433.