Literature Review of Spatio-Temporal Database Models

Nikos Pelekis[1,2,†], Babis Theodoulidis[1], Ioannis Kopanakis[1], Yannis Theodoridis[2]

[1]Center of Research in Information Management (CRIM)
Department of Computation, UMIST
URL: http://www.crim.org.uk
E-mail: pele@ath.forthnet.gr, babis@co.umist.ac.uk, kopanak@csd.uoc.gr

[2]Department of Informatics
University of Piraeus
Athens, Hellas
URL: http://www.unipi.gr
E-mail: {npelekis, ytheod}@unipi.gr

Abstract

Recent efforts in spatial and temporal data models and database systems attempt to achieve an appropriate kind of interaction between the two areas. This paper reviews the different types of spatio-temporal data models that have been proposed in the literature as well as new theories and concepts that have emerged. It provides an overview of previous achievements within the domain and critically evaluates the various approaches through the use of a case study and the construction of a comparison framework. This comparative review is followed by a comprehensive description of the new lines of research that emanate from the latest efforts inside the spatio-temporal research community.

1 Introduction

Spatio-temporal databases deal with applications where data types are characterized by both spatial and temporal semantics. Development and research in this area started decades ago, when management and manipulation of data, relating to both spatial and temporal changes, was recognized as an indispensable assignment. However, spatio-temporal data handling was not a straightforward task due to the complexity of the data structures requiring careful analysis in structuring the dimensions, together with the representation and manipulation of the data involved. Therefore, the earlier work in this area began from separate research in both temporal [TCG+93] and spatial [Gut94] databases. This effort later became the basis for spatio-temporal database models.

Since the integration of spatial and temporal database models into spatio-temporal database models, a number of new approaches have been proposed. At the same time, reviews of these works have classified and compared the existing spatio-temporal models. Currently, domain experts are trying to achieve more effective integration of the spatial and temporal aspects providing practical, unified spatio-temporal data modeling, and clarifying the direction for further research and development. Standing at this point the contribution and contemporaneously the aim of this paper is to provide a complete literature review of existing spatio-temporal database models developed or suggested in recent decades and for the first time to critically compare and evaluate them in terms of some universal criteria, in order to identify the trend as well as the needs for further research in the area.

The attempt to classify most of the existing data models was facilitated by previous reviews in the field. One of the most significant contributors of the domain has been Gail Langran who first looked at the aspects of time in Geographic Information Systems (GIS) [Lan92]. Although Langran's work covers many of the most important issues of spatio-temporal systems, a number of new proposals have emerged since. In this survey, earlier review works are considered and included in order to capture all the trends and the ideas proposed in the domain of spatio-temporal database modeling. [Fra92] was one of the first attempts to present the poor (till then) theories and methods of reasoning in the time-varying spatial space, while a bibliography on spatio-temporal databases until 1994 was published in [ASS94], which contains interesting pointers for further reference. More fruitful reviews of the domain were available the forthcoming years in [Yua96a], [Ren97a], [AR99], [Sel99], [Peu01] and especially in [Pav98] where is introduced the classification of spatio-temporal database models used in the current survey of the area.

† Contact author's address: 80 Karaoli-Dimitriou St., GR-18534 Piraeus, Greece. Tel: +30-2104142449, Fax: +30-2104142264.

2 Definitions on Spatio-Temporal Data Modeling

As delineated in the papers just cited, evolving research on space-time representation has focused on a number of specific areas, including: (a) the ontology of space and time and the development of efficient and robust space-time database models and languages; (b) inexactness and scaling issues; (c) graphical user interfaces and query optimisation; (d) indexing techniques for space-time databases. Given the need for a more complete theoretical foundation and the requirement for corresponding implementation solutions, this paper focuses on the area first listed, rather than the other three areas where secondary implementation issues are outlined.

Spatio-Temporal data models are the core of a **S**patio-**T**emporal **I**nformation **S**ystem (STIS); they define object data types, relationships, operations and rules to maintain database integrity. A rigorous data model must anticipate spatio-temporal queries and analytical methods to be performed in the STIS. Spatio-temporal database models are intended to deal with real world applications, where spatial changes occur over the time line. A serious weakness of existing models is that each of them deals with few common characteristics found across a number of specific applications. Thus the applicability of the model to different cases, fails on spatio-temporal behaviours not anticipated by the application used for the initial model development.

The study of the literature of the domain highlighted a set of precise characteristics of existing models that stand for the requirements of spatio-temporal database community. These requirements implicitly form an evaluation norm for spatio-temporal data modeling. From this study common directions of modeling in the area were identified and weak and strong points of different research approaches were also detected. Consequently, this allows us to recognise the achievements of previous works, to identify the issues where we should concentrate to and as such choose the routes for subsequent improvements.

The above-mentioned requirements fall in four categories. The first category deals just with the nature of time including the basic features that are used to describe it. The second category handles the pure spatial aspects of the existing approaches. The third deals with the unified spatio-temporal semantics, while the last category considers the query capabilities of the models. As we believe, if these requirements are followed carefully in the process of designing a spatio-temporal database model, a robust and expandable model can be achieved, capable of dealing with most of the real world spatio-temporal processes.

2.1 Temporal Semantics

***Granularity*:** Granularity is specified by an anchored point on the time axis and a partitioning length [KT96b]. The anchored point denotes where the partitioning begins while the partitioning length denotes the size of each granule. Different applications require different levels of granularity.

Temporal operations: In the literature a series of specific operations describing temporal relationships have been proposed and proved necessary in handing any time-referenced information. Allen [All83] was the first who introduced such kind of operations (e.g. timepoint *T* "*inside*" temporal period *A* which "*meets*" period *B*).

***Time density*:** This issue arises whether time should be modeled as discrete elements (isomorphic to integers) or as continuous elements (isomorphic to real numbers). More specifically, time density is closely related to the types of changes/events that can occur to the value of a thematic or spatial characteristic. Stable features that are exposed to sudden events present *stepwise* constant values (e.g. position of tectonic rocks). Continuously changing attributes can be divided into two sub-categories according to their pattern of change. As such we notice either *uniform* (e.g. ships, airplanes) or *irregular* (e.g. taxis moving in a city centre) types of continuous changes. A third type of entities exhibits *discrete* values (e.g. seismograms that interrelate measurements of earthquakes at different locations), which are collected on a periodical or irregular basis. Additionally some entities are static and never change (e.g. historic battlefields) while other entities may be measured or depend on the time itself [Ren96].

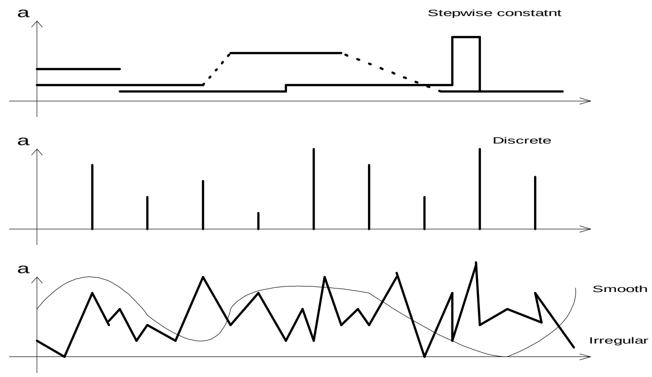

Figure 1 Types of changes according to time density

***Representation of time*:** In a model time is represented by timestamps, where representation methods are different for each model. This criterion allows us to compare each modeling technique, by whether maintaining the duration of the status of an object or recording events that imply status change. A practical question that arises with the representation of spatio-temporal data is what to timestamp or on which level to associate the temporal references. In [Wor94b] Worboys depicts two approaches of amalgamating spatial and temporal references in a computer-based system:

- The first option is to timestamp the entire geographical object. This is likely to be the most inexpensive of the options in terms of computer storage. However, it allows only a limited expression of the temporal properties of the object throughout its life.
- The second option is to fuse time and space at the primitive spatial object level (point, polygon). Although this option leads to a greater storage overhead, it allows a much finer granularity of expression of temporal variability within objects.

***Transaction / Valid time*:** There are two different clauses that a model utilizes to associate time with spatial changes-processes. The transaction time (or registration time) indicates the time an event is actually recorded in the database. The valid time (or real-world time) describes the time that an event actually happened in the real world. A spatio-temporal model that supports both transaction and valid time is said to maintain bitemporal time [KT96b].

***Time order*:** [Haz92] observes two major metaphors/criteria for describing the perspective of time, that of time as an arrow, representing progress, and as a cycle, representing constancy and continuity. They are complemented by two other (non-linear) metaphors, namely branching and multi-dimensional time [TL91].

***Lifespan*:** This factor shows if a model supports and deals with the duration of an event. This also concerns whether a model keeps track of the history of the real world objects, in terms of storing the lifespan of a discrete phenomenon or the temporal differences for a continuous one.

2.2 Spatial Semantics

Structure of space: This criterion represents the two basic approaches for computer storage of geographic data, which are the *raster* and *vector* spatial data models [Wor94b]. *Raster* data are structured as an array of cells, pixels or voxels for 2D or 3D representations respectively. Space is partitioned into grids where each cell is addressed by its position in the raster array. On the other hand, *vector* techniques describe each spatial object in terms of start and end points. Vector representations make more efficient use of computer storage as they utilize only useful data and not the entire plane.

***Orientation/Direction*:** This standard demonstrates whether a model supports the orientation and the direction features that real world objects show in space (e.g. *on the left side of, to the right*).

***Measurement*:** This issue examines whether it is possible to get a value of a spatial object (e.g. *length, perimeter, distance* etc) using a particular model or if a model supports comparative operations such as *bigger, longer*.

***Topology*:** With this criterion we distinguish existing spatio-temporal modeling techniques according to whether they support different topological relationships for the real world spatial objects. [Haz92] lists all possible topological relationships between objects of various dimensions in up to 4-dimensional space. Figure 2 visualizes common topological relationships.

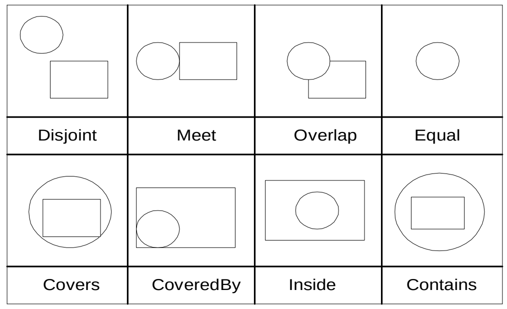

Figure 2 Topological relationships

2.3 The Spatio-Temporal Semantics

Data types: This issue refers to the basic spatial, temporal or spatio-temporal data types adopted by each model. Examples of spatial data types are the point, line and region whereas temporal point and interval are samples of temporal data types. Finally, moving point and moving region are characteristic cases of unified spatio-temporal data types.

Primitive notions: This criterion specifies the abstraction of the real world used by each model. Each model concentrates on different aspects of the real world to represent spatio-temporality in the information system. The notions vary depending not only on the method used but on the observations and choices of the particular modeller as well.

Type of change: This norm compares the models if they are able to deal with changes in shape and size of the objects. Models are also evaluated whether a change in the description of a spatio-temporal object can be combined with a synchronous representation of the change of an object's position. Consequently, the morphology, topology and attributes of a spatio-temporal object may or may not change over time, allowing for eight different scenarios (figure 3).

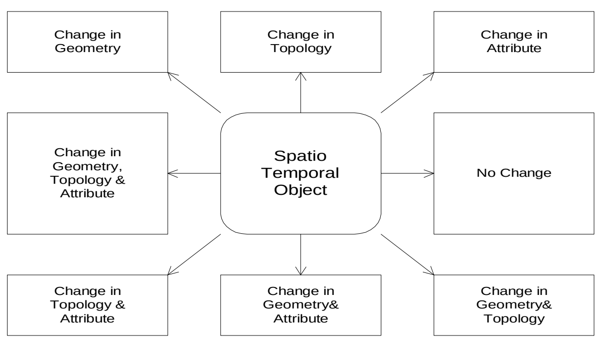

Figure 3 The eight possible types of change of a spatio-temporal object

This norm further considers whether a model supports spatio-temporal real world objects that change continuously or just objects that are subject to discrete changes. An additional emerging criterion that further categorizes existing approaches that follow the continuous paradigm is whether the latter can deal with the movement of the spatial objects over time. This is an extra decisive factor that differentiates models that support changes upon the position and/or the extent of objects in the unified space-time continuum.

Evolution in time & space: This factor shows if there are defined functions like *evolution, creation, fusion etc.* to observe and describe the movement or change of objects in space, independently from their object identification. The norm is also applied to compare models on the existence of operations able to calculate the velocity and/or the acceleration of the movement of spatio-temporal objects. There are specific difficulties presented by the evolution of objects at different speeds. Normally, a physical object is thought to continuously move/change during its lifespan. On the other hand, a re-occurring event must be viewed discretely. Slow continuous evolution of an object may also be disrupted by sudden drastic changes, some of which may be permanent, others temporary, or even part of a cycle. To model such evolution, a modeller may choose a discrete representation or alternatively he may use step semantics or some kind of interpolation to represent the unrecorded states. Similarly, extrapolation can be utilized to predict the short-term future.

Space-time Topology: This criterion sets a standard whether models can estimate metrics like values of distance, direction and change in size of a particular object. It further evaluates the ability of the models to represent topological relationships (figure 2) between (in particular continuously) evolving spatial objects for a certain period of time. In other words, it considers relationships that are defined as the combination of the spatial topological relationships with Allen's [All83] temporal relationships.

***Object identities*:** Another issue that can be employed to evaluate the modeling ability of existing spatio-temporal data models is the manipulation of the identity of an object. In particular, the lifespan of an object is an important application dependant variable. The question is when does "change" affect an object so as not to be called the same object any more? Some times it may be more appropriate to destroy the original instance of an object and re-create a new one, due to an extensive change. Another critical issue is that of splitting or unifying objects.

***Dimensionality*:** With this criterion models are examined whether they support 2 dimensions to model the spatio-temporal objects, as traditional GIS do. Although "2.5" dimensional solutions exist (perspectives, stereo views etc.), volumetric 3 dimensional GIS provide advantages in displaying spatio-temporal data. In more recent approaches, relegating the attribute value associated with grid locations to a fourth dimension, time can be introduced as a fifth.

2.4 Query Capabilities

This section classifies existing spatio-temporal database models in terms of their query capabilities. The proposed categorization is a superset of a similar benchmarking framework presented in [The03].

***Queries about locations, spatial properties, and spatial relationships*:** Queries of this category involve stationary reference objects. Examples include attribute of entities independent of space and time (e.g. who is the owner of this parcel?), as well as point (e.g. where is this building?), range / distance-based (e.g. find gas stations in this rectangular area / in this circle), nearest-neighbor (e.g. find the closest gas station) and topological queries (e.g. find streets crossing a particular area).

***Queries about time, temporal properties, and temporal relationships*:** These queries can be simple temporal queries (e.g. what is the state of a spatial feature at time t?), temporal range queries (e.g. what happens to that feature over a given period?) and temporal relationship queries (e.g. find stadiums that were built in Athens concurrently and their construction took less than six months).

Queries about spatio-temporal behaviors and relationships: This set of queries is further classified into three sub-categories: (a) *Simple* spatio-temporal queries on discretely changing (e.g. what is the state of a parcel at time t?) or moving reference objects; examples include distance-based (e.g. find humans passed close to me yesterday) and similarity-based queries (e.g. find a similar trajectory to the one I followed today). (b) Spatio-temporal *range* queries (e.g. what happens to a region over a given period?) and/or *join* queries; examples include distance-join (find the three closest restaurants to my fleet) and similarity-join queries (find the two most similar pairs of trajectories in month January). (c) Spatio-temporal *behavior* queries involving unary operators, such as traveled distance or speed (e.g. find the average speed on Saturday nights, when/where did the fire reach its maximal rate of spread?).

Based on related research work [KS+03], the above queries constitute a minimum functionality a spatio-temporal system should provide and we expect that soon coming releases of commercial DBMSs will partially support them.

2.5 Case study: Land Information System (LIS)

The dynamic environment of the real world can be represented by spatio-temporal databases incorporating the behaviour of objects in space and time. Geographical information systems, automated mapping facilities, land use information, road or city planning, migration of population and forest fires are some applications where a spatio-temporal data model can be used. Figure 4 illustrates such a spatio-temporal application. The case study deals with objects that change their shape and/or their position discretely or gradually over time. Concurrently it enables the description of the changes that take place upon the thematic properties of the involved objects.

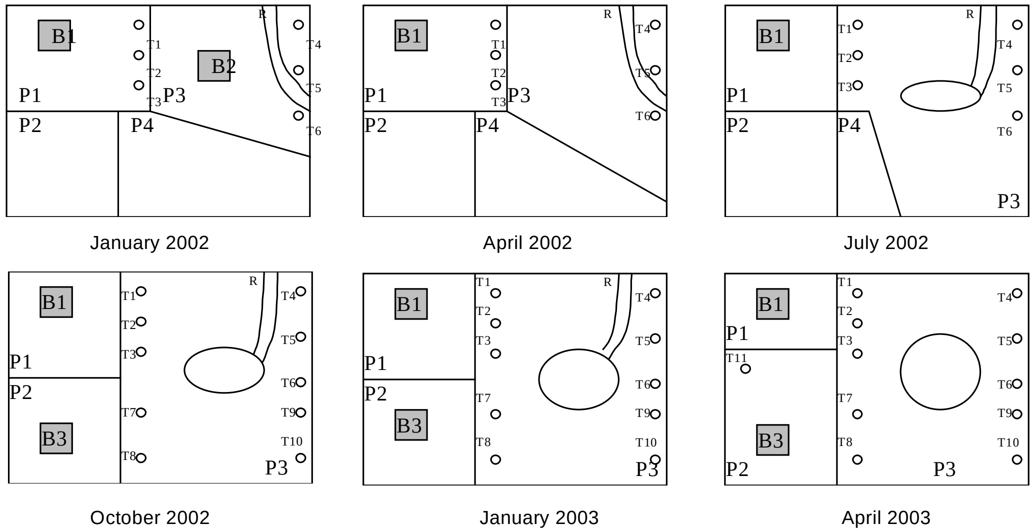

Figure 4 The Land Information System

- **January 2002:** A land parcel owned by Bob (P1), a land parcel owned by John (P2), a land parcel owned by council (P3), a parcel owned by Bill (P4), buildings B1 and B2, river R and trees T1-T6.
- **February 2002:** Council proposes Bill to buy his land to build a park in the future.
- **April 2002:** Council decided to remove building B2 from its parcel and expands the parcel incorporating part of the parcel P4.
- **May 2002:** Bob decides to sell a part of his land to council. Council develops a project to build a park.
- **July 2002:** Council expands its parcel by incorporating part of the P1 with trees T1-T3 owned by Bob and part of the parcel P4 owned by Bill. It further changes the root of the river R and starts to create a lake L. To do so it changes the position of tree T6. John starts the construction of B3.
- **October 2002:** Council gets parcel P4 and plants trees T7-T10. Lake L becomes larger while by this time John completes building B3 on his parcel P2.
- **January 2003:** Lake continues to grow.
- **April 2003:** Council stops the flow of river R. Parcel P3 becomes a park with several trees T1-T10 and lake L among them. P2 is enlarged and T11 is planted inside its extension.

From the chronology of the events and the actual changes shown in figure 5 it is obvious there are references to transaction and valid time. In May 2002 (transaction time) Bob decides to sell part of his land to council and the actual fact happens in July 2002 (valid time). From the information provided many of the valid times are not exact. For example, it is not clear on which exact time point the direction of the river was changed. In figure 5 circles show the events measured or recorded. The dotted circles show the changes made by mutual agreement or contract, which do not change the position of the objects. These changes can be recorded in the database as planned project for park. The actual park can be not developed in a real world, but the planned picture can exist virtually in the database. The filled dotted circles show changes, which take place in the real world but are not recorded or captured by any means.

The indexes used in the event diagram are explained in table 1. Change of spatio-temporal objects' identity is described by operations like *create, destroy, suspend* and *resume*, in case of a corresponding creation, deletion, temporal freezing and restoration of an object. *Fission* shows the emerging of new successor objects on the position of a destroyed object. Consequently, *fusion* shows the creation of a single object on the position of several objects destroyed.

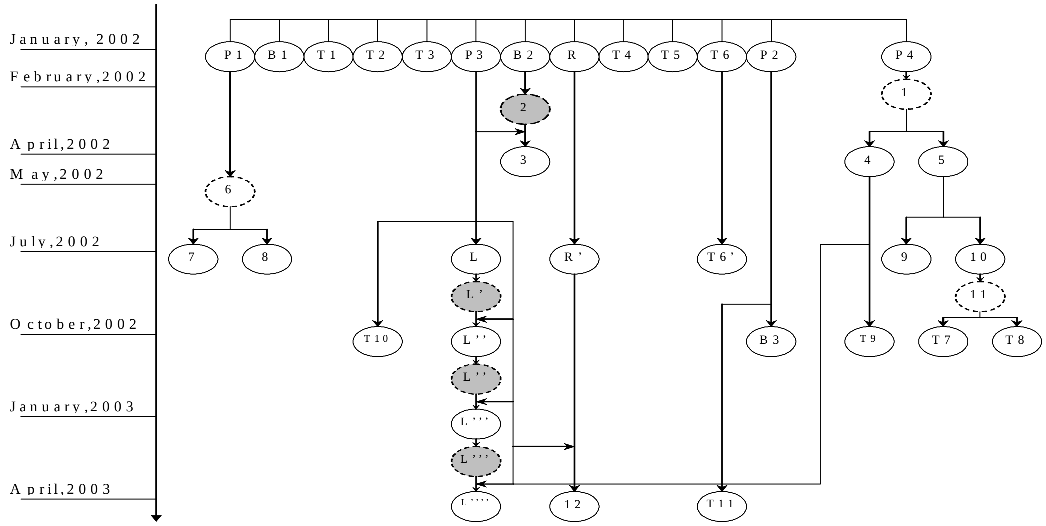

Figure 5 Event diagram for the Land Information System

Index	*Event*	*Operation*	*Time*
1	Council proposes to buy a land from Bill	Suspend	TT
2	Demolishing of the building B2	Suspend	TT
3	B2 is demolished and the land where the building stood is a part of P3	Destroy	VT
4	Council gets a part of the P4	Create	VT
5	A part of P4 still belongs to Bill	Create	VT
6	As a result of Bob's decision to sell a part of his land to council, the project on the new park is developed	Suspend	TT
7	P1 is decreased in size, property of Bob	Create (fission)	VT
8	Council gets a part of P1	Create (fission)	VT
9	Council gets a part of P4	Create (fission)	VT
10	P4 is decreased in size, property of Bill	Create (fission)	VT
11	Last part of P4 becomes a council's estate	Suspend	TT
12	The river is stopped, and its bank becomes a part of P3	Destroy, resume	VT
P1-P4	Parcel 1-Parcel 4	Create	VT
T1-T11	Tree1 – Tree 11	Create	VT
B1-B3	Building1-Building3	Create	VT
R	River	Create	VT
L	Lake	Create	VT
T6'	Tree T6 moved to a new position	Resume (fission)	VT
R'	River R changed its route	Resume (fission)	VT
L'-L'''	Recorded change of lake L.	Resume (fission)	TT

Table 1 Event indexes for LIS

3 Spatio-Temporal Data Models

Throughout the relatively young history of research on spatio-temporal modeling, a substantial number of models have been presented. This section examines most of the spatio-temporal models proposed in the literature the last two decades. Each model is applied to the case study and evaluated with respect to the comparison framework formed by the requirements presented in section 2.

3.1 The Snapshot Model

One of the simplest spatio-temporal data models is the snapshot model [Lan88]. Temporal information has been incorporated into this spatial data model by time-stamping layers. In this model, every layer is a collection of temporally homogeneous units of one theme. It shows the states of a geographic distribution at different times without explicit temporal relations among layers.

The time dimension of the snapshot model is based on the linear, discrete, absolute time model. Only valid time is supported. Time is considered as an attribute of the location. The model is the simplest way to represent spatio-temporal information, but its capability to support complex queries is the most limited. It is therefore capable to answer simple spatial, temporal and spatio-temporal queries but it is difficult for the model to resolve all the other types of queries. The snapshot sequence of time slices for the LIS is given by figure 4 that describes our case study. The snapshot model depicts three major disadvantages [Lan92]:

- The model isn't appropriate to describe changes in space through time. Each snapshot describes what exists at T_i. But to detect how T_i differs from T_j, the two snapshots must be compared exhaustively.
- Regardless the magnitude of changes, a complete snapshot is produced at each time slice, which duplicates all the unchanged data.
- It is very difficult to devise or enforce rules for internal logic or integrity because the model does not provide understanding of the constraints upon temporal structure.

Another closely related approach was presented in [Arm92], where spatio-temporal data are looked at with respect to storage, retrieval and update efficiency. He compares three approaches, which he calls *"estimation methods"* to describe time-varying spatial information. The aim of his investigation is to see if these methods have the ability to store/reconstruct complete geographical states, offer functionality for comparisons between states, and describe the events that lead to changes between states. In *static mode*, snapshots of full states are kept which leads to the storage of redundant information. To detect changes between snapshots, relatively expensive computational algorithms must be used, although this would still not explain the processes leading to the change.

In *differential mode*, only the initial state is fully recorded. Changes are stored in one of two possible kinds of *"delta files"*, which record the differences from either the previous state or the initial one. This reduces storage requirements substantially, and makes the computation of changes between states less costly. However, to reinstate previous states or the current one, a series of delta files must be applied to the initial state, which makes this operation inefficient. Alternatively, the current state can be chosen to be stored in full, keeping delta files to trace back to previous states, which is the preferable solution if the current state is more frequently accessed than historic ones.

This is somewhat similar to the final, *sequential updating mode* that also keeps the current state of the map on record. However, this approach records changes as they happen and not in a snapshot-like fashion, and uses indexes to access previous information, eliminating data redundancy. Another alternative, still following the idea of sequential updating, is to drop the use of delta files and retain unchanged components instead. In this scenario, when an object changes, its previous version is superseded but fully retained accessible by temporal links and indexes, while a new object is created to describe the current state of the component.

3.2 The Space-Time Composite (STC) Data Model

This model has been suggested by Langran in [Lan88]. It is based on the principle that every line in space and time is projected down to the spatial plane and intersected with each other creating a polygon mesh. Each polygon in this mesh has its own attribute history associated with it. Each new amendment is intersected with the already existing lines, and new polygons are formed with individual histories. The model has been tested with a number of existing methods [Lan92]. The results that were obtained looked promising, but only small data sets were tested. The space-time composite for LIS is shown in figure 6. We notice that each change causes the changed portion of the coverage to break from its parent object to become a discrete object with its own distinct history.

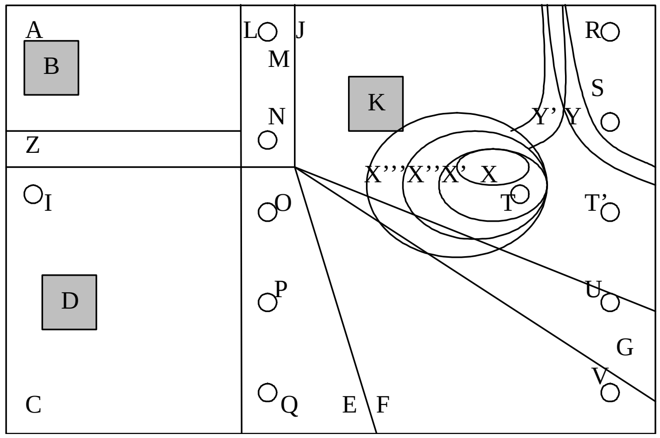

Figure 6 Space-Time Composite Model for the LIS

The time dimension of the space-time composite data model is based on the linear, discrete, relative time model. Both valid and transaction time are supported while the model supports multiple granularities and time is represented as an integral part of spatial entities. The model assumes vector structure of space and the basic data type is the polygon. A space-time composite conceptually describes the change of a spatial object through a period of time. Attribute changes are recorded at discrete times, although its temporal resolution is not necessarily accurate. The STC model is able to record temporality within the largest common units of attribute, space and time but it fails to capture temporality among attributes across space (i.e. movement). In addition, updating a STC database requires reconstruction of STC units.

The model has shown sufficient support for most types of spatio-temporal queries. However, it has difficulty in facilitating queries about spatio-temporal behaviours and relationships. A serious problem with the STC model concerns retroactive changes to identifiers. Each time the space-time composite splits an object into two, the old object is effectively replaced by two new objects with new identifiers. This means that throughout the database, each occurrence of the old object identifier must be replaced with one or both of the new ones.

3.3 Data Models based on Simple Time-Stamping

Another simple approach is to tag every object with a pair of timestamps, one for the time of creation and one for the time of cessation. Current objects have their cessation time given by a special value *“NOW”*, *“CURRENT”*, or *“NULL”* [Ren96]. In [HW90] the authors have implemented such an approach and show that time slices can easily be retrieved by simple queries. They argue that in digital cadastral databases, storing full layers of graphical information for different time periods is impractical, and describe a system that keeps a graphics file of current parcels for day to day use while archiving historical spatial data into a separate file. Reference to this information is still kept in the files that store aspatial information via multiple versioned copies of the same parcel record.

The model is based on the linear, discrete, absolute time model. Only valid time is supported while the model supports multiple granularities. Time is represented as an attribute of the object and vector structure of space is assumed. However, such a model spreads the different versions of the same object over several non-related tuples around the same table. This makes it hard to trace the history of one single object. This deficiency can be resolved by adding explicit references to preceding and succeeding versions of the objects. The idea comes from an object-oriented model [RMD94] where is introduced the *Temporal Change Object*, which is an object consisting of a set of references to past (historic), future (scheduled) and the current version. The time cross-section of this model for the LIS is described in figure 7.

The strength of simple time-stamping approach is that it is relatively easy to obtain states of objects at certain times. The main disadvantage is that it is not possible to obtain direct information of what happened or why it happened. In other words, the changes can be obtained in terms of their effects rather than as explicit information. This means that the model is strong on queries like *“What was the state of…”* and weak on queries like *“What happened…”*.

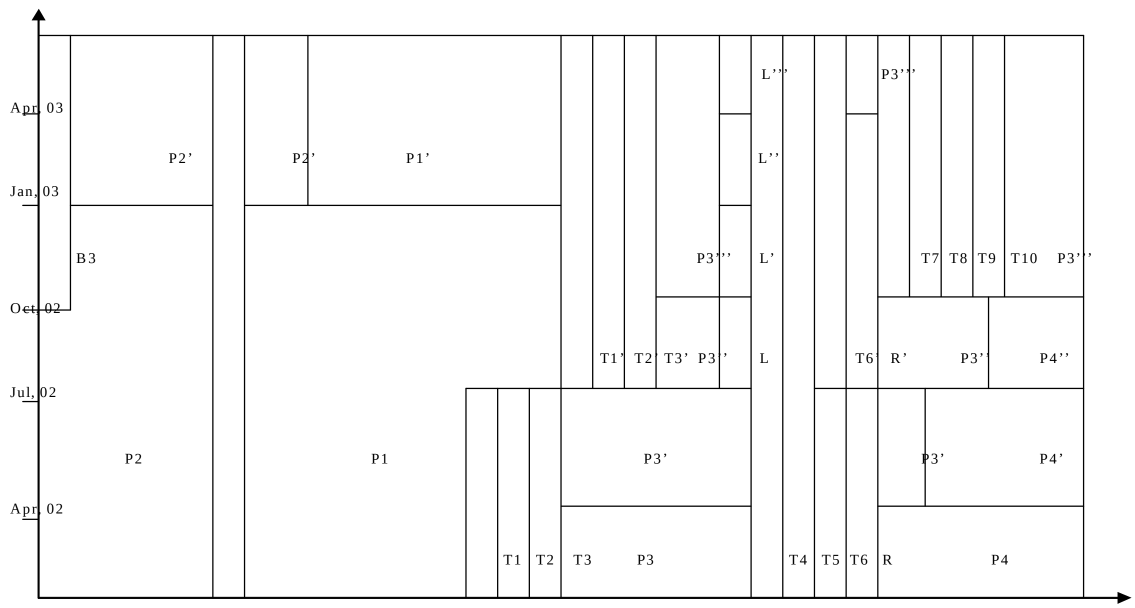

Figure 7 Simple Time-Stamping approach for the LIS

3.4 Event-Oriented Models

The previous models cannot identify individual changes or events to the data set. One way to overcome this is to represent the events explicitly. Imagine a traditional GIS, if all changes that were made to each data set were logged into a transaction log, that log itself would provide all the information needed in a spatio-temporal system. The actual database would then act as a current state database, and in order to obtain historical states of the map, a ***"rewind"*** can be obtained by tracing the transaction log backwards. Thus, the transaction log itself truly is a temporal database.

Data models that have been implemented or even discussed at any level of detail for handling spatio-temporal data are extensions of traditional or vector models. These traditional models can be seen as location-based or object-based. It is now well known that both of these models are required in a GIS if both location-based and object-based queries are to be handled effectively. It follows that neither of these two forms, even when extended to include time would be as effective as a temporal-based representation for handling time-based queries.

In [PW94] is proposed such an approach for the time-based analysis of spatio-temporal data, as an adjunct to location-based and object-based analysis. The design of TEMPEST, a prototype Temporal Geographic Information System (TGIS) that implements this approach, incorporates this time-based data model and associated relational operators. Starting with an initial state (base map), events are recorded in a chain-like fashion in increasing temporal order, with each event associated with a list of all changes that occurred since the last update of the event vector. An event may represent abrupt change or can be triggered when gradual evolution is considered to be significant enough (over some predefined threshold), to register change. Changes can be stored as differences from the previous version, which avoids data redundancy, or if they are considered to be extensive, the full map may be registered.

Additionally, Peuquet and Duan have implemented a raster-based event-oriented approach, called the Event Oriented Spatio-Temporal Data Model (ESTDM) [PD95]. ESTDM groups time-stamped layers to show observations of a single event in a temporal sequence. The ESTDM stores changes in relation to a previous state rather than a snapshot of an instance. A header file contains information about its thematic domain, pointer to a base map, and pointers to the first and last event lists. The base map shows an initial snapshot of a single theme of interest in a geographic area. Every event is time-stamped and associated with a list of event components to indicate where changes have occurred (figure 8). An event component shows changes to a predefined location (a raster cell) at a particular point in time. The ESTDM has shown its capabilities and efficiency to support both spatial and temporal queries. However, the transformation of the ESTDM to a vector-based system requires a substantial redesign of event components. Mechanisms are needed to allow event components to keep track of their predefined entities and locations.

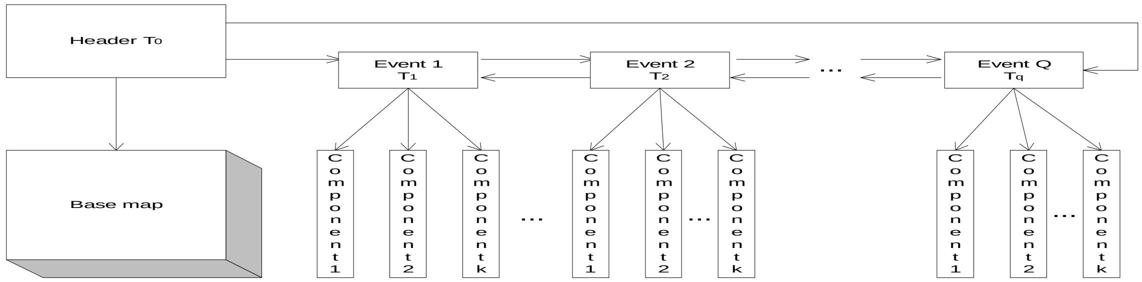

Figure 8 Primary Element of an Event-Orient Model

Langran describes another event-oriented approach, called the amendment vector approach where a base state (or a final state) is overlaid with amendment maps, representing the events in the database [Lan92]. Figure 9 shows this event-based model for the LIS. The advantage of this representation is that information about what happened to the objects is stored in the database. The time domain considers discrete, relative, linear modeling.

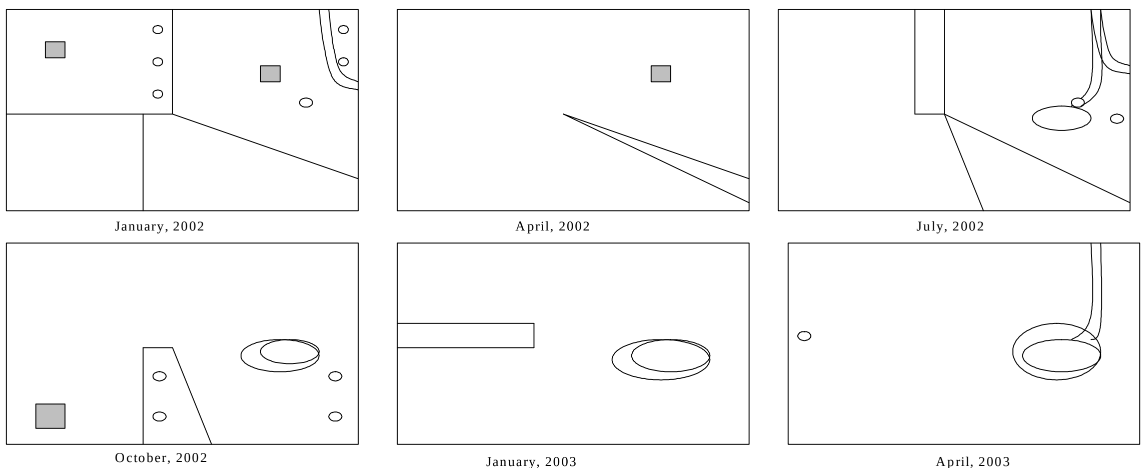

Figure 9 Amendment Vector Model for LIS

3.5 The Three-Domain Model

Yuan in [Yua94] and [Yua96b] describes a three-domain model for spatio-temporal modeling. Her model (figure 10) represents semantics, space and time separately and provides links between them to describe geographic processes and phenomena. The semantic domain holds uniquely identifiable objects that correspond to human concepts independent of their spatial and temporal location. This is in contrast to other models where, for example, a landowner is represented as an attribute of a land parcel. In the three-domain model, the landowner is a semantic entity that is linked to a land parcel (spatial object), with changes to the parcel associated with dates (temporal objects), and possible other land parcels involved in the transformation. Loss of ownership is implemented by linking another semantic entity to the land parcel together with the temporal object representing the date of sale.

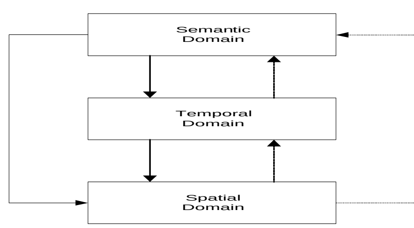

Figure 10 A Conceptual Framework for the Three-Domain Model

The advantage of this vector model is the ability to handle movement as well as change, which is an improvement over many existing models that handle either the former or as in most cases, the latter. A classification of change (as a general term including both “change” and “movement”) for this purpose is given, with two groups identified in each of the three domains. Semantic changes include variations in attributes over time and the static spatial distribution of a geographic phenomenon. Spatial changes may be static, looking at variations of a geographic phenomenon at a snapshot, or transitional, comparing states of an event at different sites. Temporal changes are either spatially fixed mutations of an event or the actual movement of it from one place to another. The model is very applicable in reality because of its highly abstract structure. Moreover it supports both valid and transaction time and time can be modelled either as absolute or relative. Furthermore, the three-domain framework can support a wide range of spatio-temporal queries because of its flexibility in information production by handling semantic, spatial and temporal information separately. Yuan in [Yua97] shows, that her model can support simple and range temporal and spatio-temporal queries.

In an alternative Three-Domain method presented in [CT95], in addition to spatial and temporal domains a thematic domain is added to represent the complete descriptive state of a spatio-temporal object. Recording of descriptive characteristics separately from spatial and temporal attributes allowed the capturing of changes of the aspatial attributes of an object. The temporal domain refers by indexes to spatial and thematic domains, thus making it possible to retain different spatial and thematic characteristics of spatio-temporal objects on the same timestamp. Similarly the relationship between domains (figure 11), allows different versions to refer to similar thematic and spatial descriptions, which reduces data handling significantly.

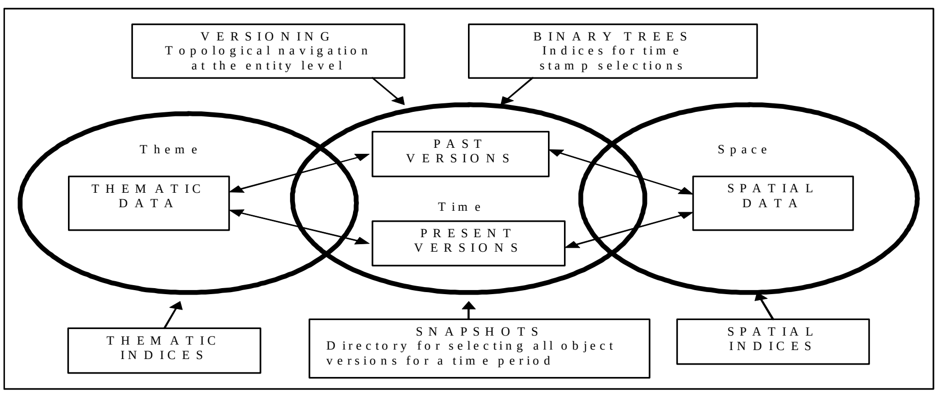

Figure 11 An Alternative Three-Domain Model

This model focuses on spatio-temporal events. Characteristics are categorized as thematic, spatial and temporal and are recorded in corresponding tables every time a change occurs. The temporal domain consists of three version tables, where past, present and future versions are recorded. These tables carry the records relating to attribute and graphics, which consequently derive their records from thematic and spatial domains. Ordering in temporal tables is bi-directional and valid time and transaction time are supported. Complex processes can be described by the extended-versioning diagram, which references multiple entities and links. A form of a spatial tree enables tracing the geometry and the topological relationships of spatial objects on the time line. The spatial changes can be recorded in the spatial domain as shown in table 2. Versioning table consists of tuples containing references to attribute table, spatial table, valid and transaction time. In addition there are columns showing next record number and last record number to perform continuous roll over events and procedures. For simplicity reasons, we omit the attribute and spatial tables and we present an illustration of the transition of the spatial objects (figure 12), the spatial tree indicating the spatio-temporal changes (figure 13) and the produced versioning table for the LIS.

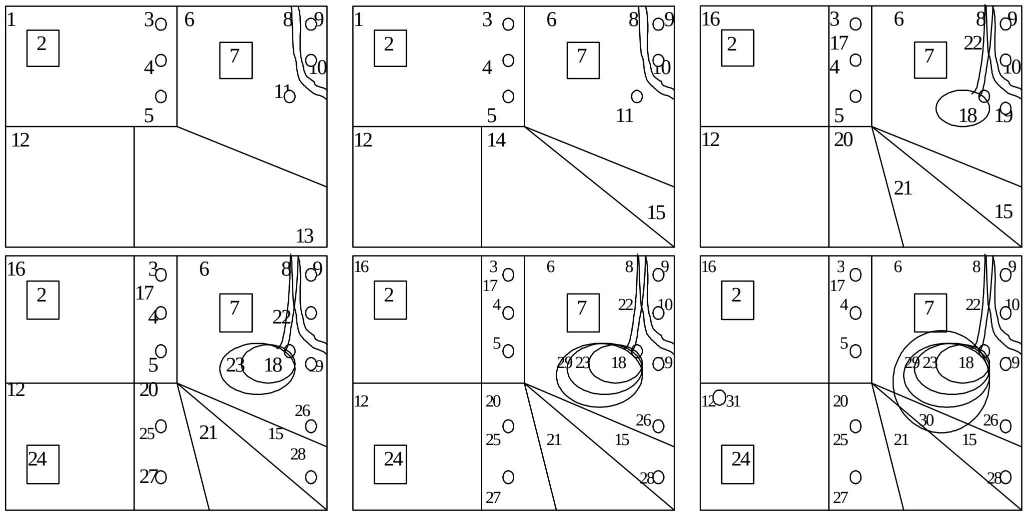

Figure 12 Three-Domain Spatial Transitions for LIS

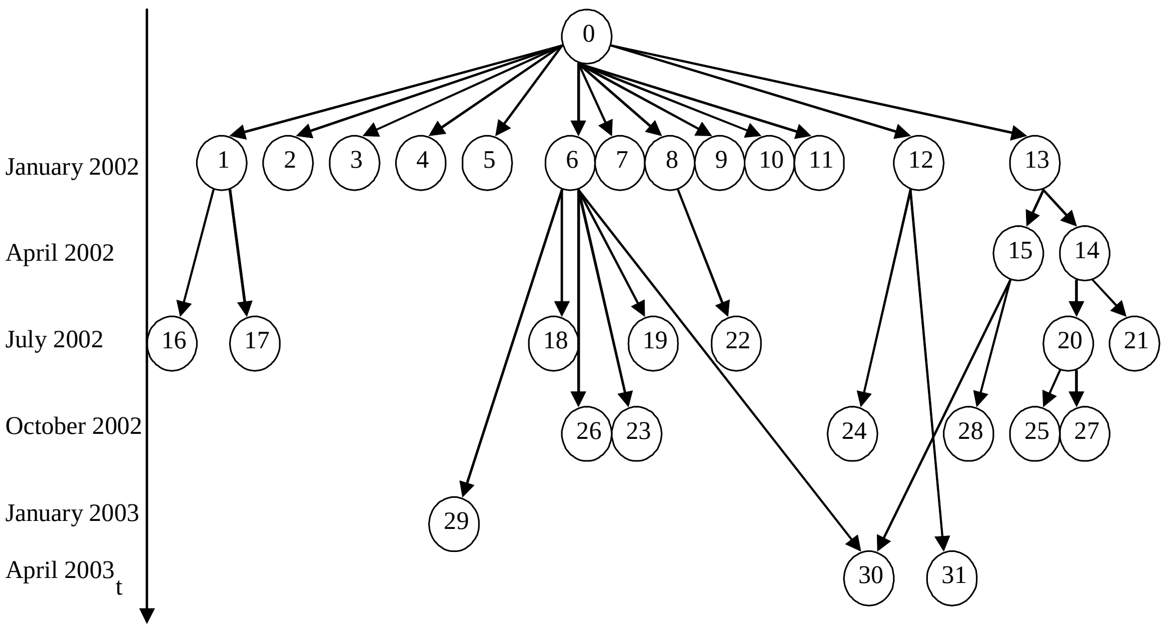

Figure 13 Three-Domain Spatio-Temporal Tree for LIS

ID	Attribute	WD_Time	VL_Time	Graph	Prev	Next	Last
1	1	January 2002		1, 2, 3, 4, 5	0	2	
2	2	January 2002		12	1	3	
3	3	January 2002		6,7,8,9,10,11	2	4	
4	4	January 2002		13	3	5	
5	3		Febr. 2002	15	4	6	
6	3	April 2002		7,15	5	7	
7	4	April 2002		14	6	8	
8	1		May 2002	17	7	9	
9	3	July 2002		17, 18,19,21, 22	8	10	
10	4	July 2002		20	9	11	
11	2		July 2002	24	10	12	
12	3	October 2002		20,25,26,27, 28, 23	11	13	
13	2	October 2002		24	12	14	
14	3	January 2003		29	13	15	
15	3	April 2003		22, 30	14	16	
16	2	April 2003		31	15	17	17

Table 2 Versioning table coming from applying Three-Domain Model into LIS.

The model was a revolution in spatio-temporal database development, because it was the first successful attempt to record individual descriptive characteristics of dynamic objects. At every occurrence of a thematic and/or a spatial change a new version of time is being added to the versions table. The spatial tree allows tracing of the changes in the spatial domain in accordance to the time line. At the same time it captures the events and procedures of the spatio-temporal database together with the spatial coordinates, geometrical and topological properties. However, there are not defined operators for dealing with the relationships among spatial objects and it needs mechanisms to calculate the change.

3.6 The History Graph Model

Understanding temporal behaviour is one of the most fundamental issues in spatio-temporal systems. A simplistic view that many researchers seem to adopt, is to represent objects only in terms of static representations, viewing changes as sudden events. However, we know that many changes in the real world have duration. Actually, features in the real world exhibit a wide range of temporal behaviour. As such, we classify real world objects into three categories: (a) continuously changing objects, (b) objects that are basically static, but they are changed by events that have duration and (c) objects that are always static and change only by sudden events.

The main purpose of the history graph model [Ren96] is to identify all types of temporal behaviour and to manage both objects and events. The intention of the history graph notation is to visualize the temporal element of geographical and other information. It is based on the simple idea that an object may either be in a static, a changing or a ceased state. In the history graph notation, the static states called object versions are shown with rectangular boxes, while the changing states called transitions between versions are shown with round ended boxes (or circles in case of sudden changes).

Each object version is identified by two timestamps describing the interval of time in which the state of the object is valid. Each transition is an entity that relates object versions with its successors or predecessors. It is also characterized by two time stamps describing the period of time in which the transition took place. Links or arrows between states display successor-predecessor relationships between them. An object’s history may be described through a series of consecutive versions and transitions. Objects that change suddenly are described by transitions with zero duration (i.e. events), while objects that change continuously are described by versions with zero duration (i.e. snapshots) describing intermediate states. In general, at least six types of transitions representing different cardinality constraints on the preceding and succeeding object versions may be identified. Figure 14 exhibits the semantics of the history graph notation, while figure 15 illustrates the six different types of change.

- *Creation*: An object is created.
- *Alteration*: An object is changed or modified either by change in attribute or in geometry.
- *Cessation*: An object is destroyed or removed and does no longer exist in the real world.
- *Reincarnation*: An object that previously has been destroyed or removed is reintroduced, possibly with a new state and location.
- *Split/Deduction*: An object is subdivided in two or more new objects or one or more objects are deducted from an existing object.
- *Merge/Annexation*: Two or more objects are joined together to form a new object or one or more objects are “swallowed” into another object.

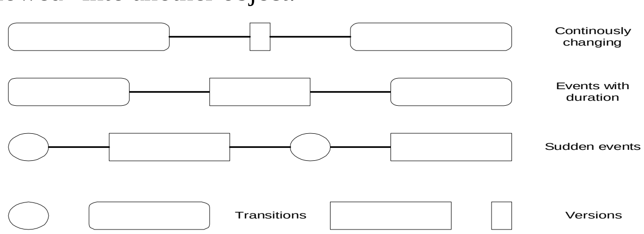

Figure 14 The behaviour types of Temporal Objects

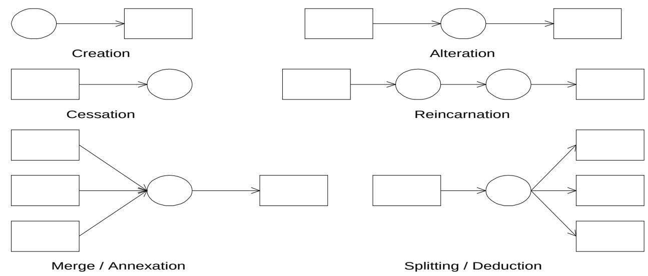

Figure 15 Six Basic Types of Changes in History Graph Model

Although the event-based models and the data models on simple time-stamping have their advantages and their disadvantages, they complement each other. Hence, it is natural to suggest that spatio-temporal databases should manage both events and objects in their data sets. History graphs are the outcome of this procedure. The primary application of a history graph is to describe a limited extent in time and space, called a story. Actually this model does not necessarily have to describe a spatial system, although the concept of splitting and merging is motivated from spatial processes. The obvious way to model LIS with history graphs, is to consider that events are instantaneous and therefore have no duration (figure 16). However, generally speaking this is not the situation. For example the building of a road may take several years, but the road may exist in a database from the day it was opened. The opening of the road may be modelled as a sudden event, while the building and planning of it may take several years. If we assume that events have no duration then the opening date of a road will be its creation date.

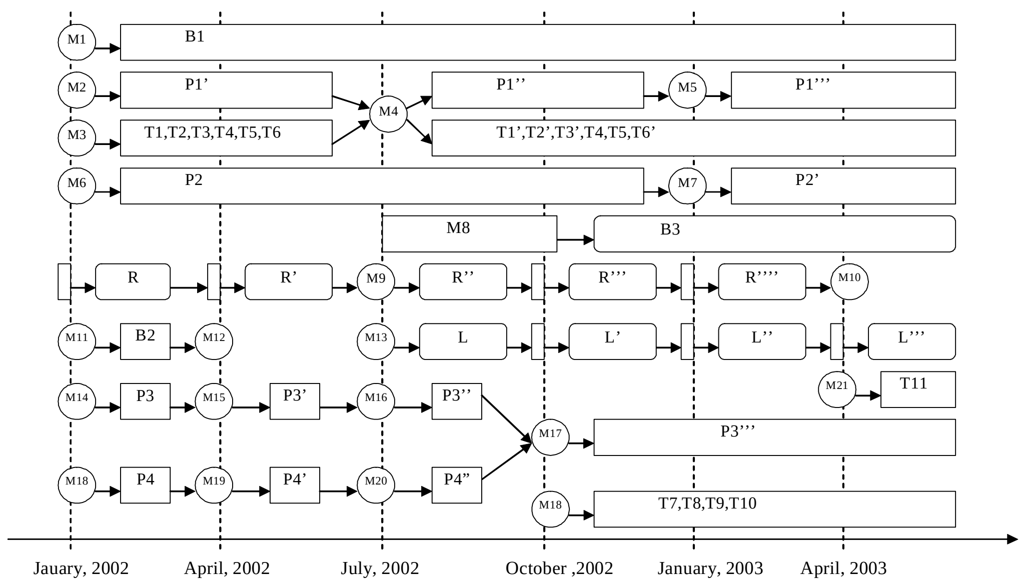

Figure 16 The History Graph for the LIS

As a whole, history graphs can be thought as an extension to the event-oriented way of thinking. Their advantage is that temporal relationships can be derived directly and that time can be modelled both as discrete or continuous as well as absolute or relative. The time axis is linear and both valid and transaction time are supported. The major advantage of this vector model is that history graphs capture all the knowledge we need in order to further develop a spatio-temporal system. Moreover, they capture both the notions of change and movement and they can easily support most types of spatio-temporal queries. Finally, Renolen has shown [Ren96] that it is possible to take advantage of inheritance when implementing the model using object-oriented methodology. The basic idea is to implement the six different types of processes as sub-classes of the process object. He further believes that the programmer’s job is simplified if the data structures can be visualized in a conceptual model. Moreover, an implementation based on both processes and states should be well suited for most types of queries, not having the deficiencies of the event-oriented or data-oriented models.

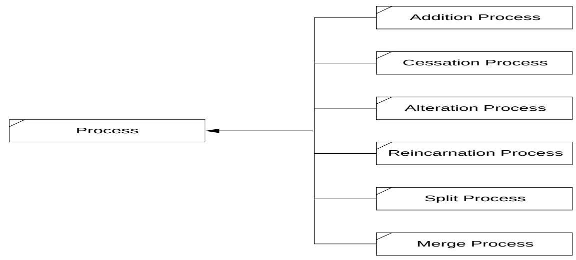

Figure 17 Object-Oriented Design of History Graphs Using Inheritance

3.7 The Spatio-Temporal Entity-Relationship (STER) Model

Entity-Relationship is the earliest and best known among conceptual database models and it has recently been extended to capture spatio-temporal information. For some years the ER model was used to represent temporally referenced business data [TL91] or extended in order to deal with solely spatially referenced data [HT97]. However, none of the proposed models satisfied the demanding requirements of a spatio-temporal information system. Later Tryfona in [Try97] proposed an extension of the entity relationship model in order to model the phenomena of the real world and to capture them into spatio-temporal applications. She extended the ER model with symbols to represent the geometry and temporality of spatio-temporal entity sets. Despite the fact that the model was working on spatio-temporal applications, the temporal semantics of attributes were missing and cardinality of time was not flexible enough to satisfy the requirements of spatio-temporal databases. In addition, a formal definition of the whole model was not demonstrated.

These attempts and investigations provided the foundation for developing the Spatio-Temporal Entity-Relationship (STER) model [TJ99] & [TJ00]. The careful analysis of spatio-temporal applications and behaviour of spatial and temporal entities suggested that entity sets with their attributes and relationships could capture the dynamic nature of spatio-temporal databases. Then, the extensive definition of spatio-temporal objects, attributes and relationships together with the requirements for the model were given. In addition, the applications for spatio-temporal databases are categorized by their data type, highlighting applications dealing with objects with continuous motion, with discrete changes and with objects integrating continuous motion as well as changes of shape. Further more, the ontological foundations for designing spatio-temporal databases are described and the requirements to capture not only the position of an object in space and time, but also their descriptive attributes, relationships and existence aspects are presented. What is more, the STER model, which is able to deal with complex geo-entity sets and interrelations of spatial and temporal semantics, allows description of attributes such as *"ownership"* and of relationships among entity sets such as *"reincarnation"*, *"splitting"*, and *"existence time"*. The model is universal in terms of reusability due to its simple and flexible notation; hence it can be directly used as a framework for other conceptual models.

Figure 18 demonstrates the model applied to the LIS. Subsequently we present the translation of just one of the entities of the graphical diagram following the syntax of STER. It is considered that the attempt to capture the spatio-temporal aspects of information in the LIS using the STER approach was successful. The temporal and spatial changes of the application are identified, while the descriptive attributes are defined on valid and transaction time lines. However, it lacks the ability to capture the actual motion of the process of change and does not indicate if a spatial object is dynamic or static.

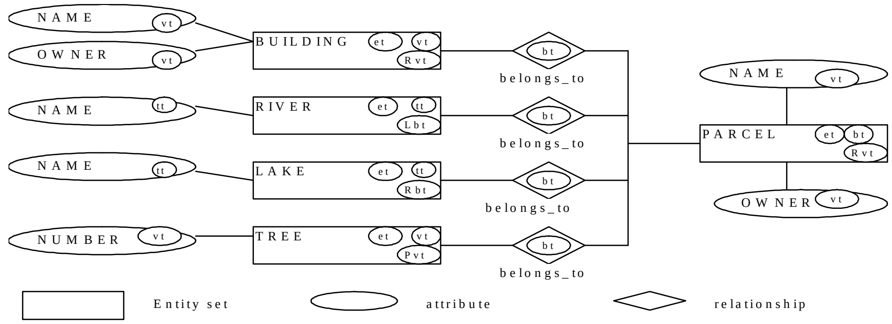

Figure 18 An excerpt of the LIS designed in STER

DEFINE ENTITY SET parcel
ATTRIBUTES
    *(*Name *VALID TIME* valid time*,*
    Owner *VALID TIME* valid time*)*
GEOMETRY REGION
    VALID TIME valid time
EXISTENCE TIME existence time
VALID TIME valid time
TRANSACTION TIME transaction time

3.8 Object-Relationship (O-R) Model

Previous conceptual level development of spatio-temporal database models may have enhanced the current GISs providing a more realistic world description of the changes and processes in space and time. However, none of the above covered the description of change processes. Considering this fact, researchers in [CPS+98] modelled an application, which could capture the natural phenomena of environmental changes. They analyzed already proposed models and came to the conclusion that existing models are more concerned with design rather than on the representation of the natural environmental changes, processes and events.

In their introduction [CPS+98] review several works on spatio-temporal database modeling. They highlighted both the weak and strong points of earlier work in capturing the spatial changes, continuous vision, discrete and linear database representation. They also identified the taxonomy of spatial changes and object oriented database models, connecting spatial entities with temporal attributes. Three different methodologies for spatio-temporal database design are described as the main thrust of their research. These are ***Modul-R***, ***Mecosig*** and ***Pollen***, which extend ER [HT97] and Object-Oriented techniques. A fourth approach called MADS (Modeling Application Data with Spatio-temporal features) [PSZ99] is also presented, which is implemented following object relationship (OR) approach. The implementation of object-relationship models describe *"processes, which act on the geometric attributes of an entity"* and illustrate the importance of capturing the processes, which cause change in connection with space and time.

The MADS incorporates space and time modeling basics into the object-relationship model [PSZ99]. The authors argue that spatio-temporal database processes allow users to handle the complex data models required by higher-level abstracted spatio-temporal applications. Therefore, the types of processes were classified and represented as relationships between involved spatio-temporal objects. At the same time, the object properties, which are required to be captured in the spatio-temporal databases, are stated. These are important to indicate the processes that cause a change. Hence, the description of processes as a relationship type between spatio-temporal objects describes best the spatio-temporal phenomena. The processes influencing a single object's geometry are described as the characteristic of the geometry attribute. Finally, the spatio-temporal processes are visualized through icons in the schema. The proposed design using the LIS case study is illustrated in figure 19 together with the description of one of the objects in the syntax of the definition language of the OR Model.

OBJECT Parcel
TEMPORAL Day
GEOMETRY AREA TEMPORAL DAY
ATTRIBUTES
    Number: INTEGER
    Name: string [1:N] TEMPORAL DAY
    Owner: [1:N] TEMPORAL DAY
END Parcel

The method is a conceptual-level representation of a spatio-temporal database and provides the basis for queries involving actual processes of the application. Although it is obvious that there is need to represent processes and changes on the conceptual level, there are no extensive definitions and operations given. The modeling of the processes is an abstraction of the real world, and a description suitable for one application may be deficient when applied to other real world scenarios.

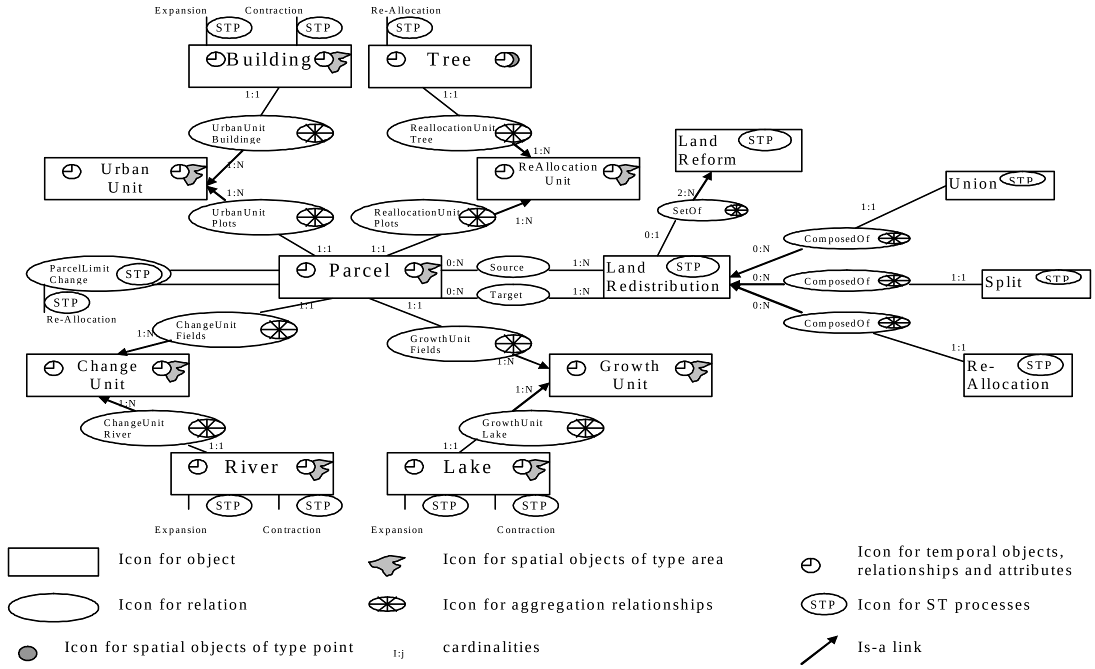

Figure 19 Object-Relationship Model for LIS

3.9 Spatio-Temporal Object-Oriented (O-O) Data Models

One of the fundamental challenges facing the software industry is dealing with change in databases. Object-oriented technology, with its ability to handle independent objects, showed promising results in spatio-temporal database management and attracted many researchers. Attempts were made to incorporate the features of the object-oriented technology such as classes and instances, attributes and abstract data types, operations and methods, classification and encapsulation, aggregation, information hiding, inheritance, polymorphism and dynamic binding, in spatio-temporal database development. As identified in [Mon95], there are four advantages of using the object-oriented approach in spatio-temporal database modeling: a) A single object can represent the whole history of an entity, b) queries are simple, because they deal with each single object of an entity, c) efficient temporal data handling, d) uniform treatment of spatial and temporal data handling.

Worboys is the first one who emerged object orientation in the spatio-temporal modeling [WHM90]. He introduces the concept of the spatio-temporal object (ST-object). He defines a spatio-temporal object as a unified object with both spatial and bitemporal extents. An elemental spatial object (a single point, finite straight line segment or triangular area), also called a simplex, is combined with a bitemporal element to form an ordered pair [Wor94a]. A finite set of such ST-simplexes are then defined to form a ST-complex on which a query algebra is developed. Furthermore, operations like union, intersection, difference, equality, subset, boundary, spatial projection and temporal projection are provided on this ST-complex.

The same year, Wachowicz and Healy [WH94] introduced an object-oriented spatio-temporal model of real-world phenomena and events. In this approach real-world phenomena are represented as complex versioned objects with geometric, topological and thematic properties. A new instance of an object with a different identifier is created for every version of the object establishing a hierarchical structure for the past, present and future of the object. On the other hand, events are manifestations of actions, which invoke update procedures on one or more objects. Time is represented as an independent, linear dimension unlike other representations where the time axis is orthogonal (i.e. modelled together with the spatial dimensions). The time reference is absolute and the time order is linear. Discrete time density is used, while space is conceptualised not only along one, but three linear dimensions.

The model introduces the concept of version management in order to integrate object and event elements. Two main versioning levels can be distinguished: *object version* and *object configuration*. Four basic premises underlie the proposed model at the object version level: a) Every object must have an initial version, b) a hierarchical structure is imposed on the versions of an object, c) different versions of an object denote different object instances, d) among versions, a current version is always distinguished.

At the version configuration level, two main object-oriented mechanisms, namely generalization and aggregation are important tools for the spatio-temporal model. These mechanisms generate an ensemble of version sets, which coexist and interrelate over time. The model supports four main update procedures and their respective spatio-temporal changes (table 3). When an update procedure creates a new object, a new version set is created. Once this version set exists within a version configuration, other update procedures can be carried out over the object at any time *t*.

Spatio-Temporal change	***Update procedure involved***
None	Creation of a new object
Geometry, topology, thematic	Creation of a new object from an existing one
Thematic	Description updating of existing objects
Geometry	Relocation of an object

Table 3 Update procedures and their respective spatio-temporal changes

In another attempt, Bonfatti and Monari [BM94] describe an integrated approach to model both geographical structures and phenomena. They argue that cross-references between objects to express relationships are ambiguous; hence better means are needed to characterize object structure and behaviour. Their proposed solution is the use of complex objects comprising of several components to express structure and relationships. In addition, laws describe the behaviour of the components. Laws are predicates that express regularities in object's states and hence determine the possible states where an object may be. They are formulated without taking into account how to keep them satisfied; each of them is simply a piece of truth relative to the behaviour of the specific object. When mapping laws into an information system they become either constraints against illegal state transformations or actions to undertake to move the system from illegal to legal states. Other laws may describe interdependencies between the components of the object.

Spatio-temporal processes can be modelled quite easily within this framework by attaching timestamps to objects (components) and expressing motion as laws for complex objects. If the motion presents regularities then laws can be described easily. A problem exists with the modeling of applications that are space-dominant like the LIS example where it is difficult to formulate laws to consider how states change. Nevertheless, we use the model to illustrate its different aspects applied to the application domain. This is achieved by decomposing complex structures and phenomena into their constituent elements. A number of objects can then be described as follows:

OBJECT *Point* {
X, Y: *real*;
}

OBJECT *StraightLine* {
A, B, C: *real*;
}

OBJECT *Segment* {
Length: *real*;
Start, End: *Point*;
LyingLine: *StraightLine*;
}

OBJECT *Line* {
Length: *real*;
Start, End: *Point*;
Shape: List of *Segment*;
}

The next step is to define laws to code the behaviour of the complex objects. For example a straight line segment, which is defined as the composition of two points, a length and three coefficients for the line equation, is described by laws stating that both points must be on the line determined by the equation and the length must equal the distance between the two points. Below we describe more formally the above-explained laws on object Segment.

- Law1: LyingLine.A * Start.X + LyingLine.B * Start.Y + LyingLine.C = 0
- Law2: LyingLine.A * End.X + LyingLine.B * End.Y + LyingLine.C = 0
- Let L = Dist(Start, End) then Law3: Length = L

As we mentioned before, the formulations of laws that concern both space and time is not possible for the LIS. Applications listed as spatio-temporal dominant (simulation modeling, electronic charting, environmental management, etc.), would be better to be modelled with the current approach. For these applications laws play an outstanding role since they express behavioural knowledge that is usually neglected by other conceptual models.

Another object oriented data model for spatio-temporal data has been proposed by Rojas-Vega and Kemp [RK95]. They describe a structure for distributed multimedia spatial applications and the

Structure and **I**nterface **D**efinition **L**anguage (SIDL) is developed for this purpose. To achieve full encapsulation necessitated by the distributed nature of the spatial database, the basic object type has a structural and an interface part. The structural part consists of a system generated object identifier, conventional attributes, which form part of the state of the object, an object component grammar and a list of conceptual relationships and constraints with external objects. The object component grammar contains the list of other objects that form part of the object, including compositional semantics that determine if a component is mandatory or optional, shareable or non-shareable, dependent or independent of the existence of the object. The interface part expresses the behavioural semantics of objects in terms of methods encapsulated in an object.

With these parts, complex object structures can be built to fully model real life entities and their interactions. Time is introduced by separate objects that can be related to time-varying components, with the use of separate objects for different models of time. Figure 20 represents these ideas at a conceptual level for an object parcel of the LIS. The object parcel is the core object of our case study. A user interface object provides spatial and non-spatial information about this object. It interacts with object Parcel to find and retrieve specific information associated with a particular parcel. The spatial part of the object Parcel reflects a complex structure and is modelled as a single geometry spatial object G-polygon. In turn, a G-polygon consists of an interior area and a set of G-rings, a G-ring being an aggregation of Strings, which are an aggregation of Lines, which are an aggregation of Points. An object Point must have one ordinate X and one ordinate Y but an ordinate X or Y might belong to several Point objects. The relationships between the objects are compositional relationships. Furthermore two more classes, a temporal class interval and an additional bitemporal interval class are defined, in order to capture temporal requirements.

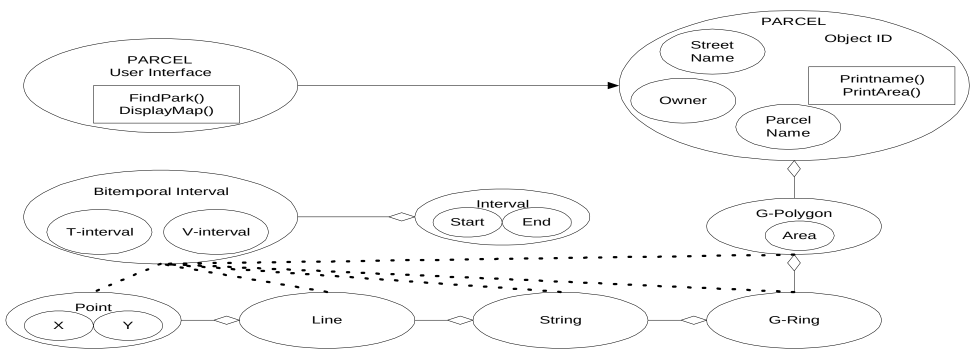

Figure 20 Modeling a Parcel of LIS with an OO Model

In [BVH96] and [FMN98] the authors propose OO models based on extensions of ObjectStore and O2 respectively. [BVH96] define a hierarchy of elementary spatial classes with both geometric and parameterized thematic attributes. Temporal properties are incorporated by adding instant and interval timestamp keywords to the query language. In [FMN98], spatial and temporal properties are added to an object class definition by associating it with pre-defined temporal and spatial object classes. This solution is not suitable for representing temporal or spatial variation at the attribute level, as the timestamp and spatial locations are defined only at the object component level. In addition, both [BVH96] and [FMN98] offer text-based query languages; the non-graphical query languages of these models reduce their suitability as conceptual modeling languages.

Other object-oriented models proposed, were based on the Object Model supported by Object Modeling Technique (OMT) [Rum91]. The main advantage of this approach is its high-level representation abstraction, which deals with the concepts of modeling rather than with implementation. An interesting approach based on the Object Model was introduced in [Ham94], where a model for representing data varying in 3D space and time (4D) is presented for storage of data sets concerning environmental monitoring and simulations.

This model assumes a discrete time axis and stores the absolute time for the data values, while only stores the time when the data is collected (valid time). The values in a data set can be spread either regularly or irregularly within the 4D space by x, y, z and t. For simplicity it is assumed that every parameter in a data set has the same variation pattern and that the set of values associated with a parameter is ordered by an integer index indicating the relative position along each of the locational axes. A data set has attributes defining its location, extent and topology in space and time, as well as attributes describing other thematic properties. Figure 21 depicts the model using the OMT notation.

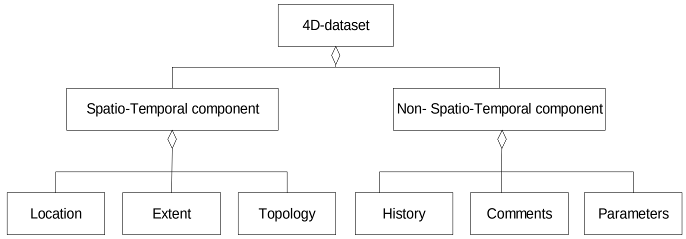

Figure 21 Overview of the components of a 4D dataset

What is more, in this model 4D data sets are classified by examining how their parameter values are spread in space and time. This gives rise to classes termed Point, PointSeries, FlatSurface, Volume and TemporalVolume, where the values are spread in 0 to 4 dimensions, respectively. The dimensionality indicates how many of the locational parameters (x, y, z, t) vary within the data set. All classes except Point can be further divided into subclasses depending on whether successive values are regularly or irregularly distributed. For PointSeries this results in the subclasses RegularPointSeries and IrregularPointSeries on the next level of the class hierarchy. These classes can again be divided into the subclasses RegularLongitudeSeries, RegularLatitudeSeries, RegularDepthSeries and RegularTimeSeries, and similarly for the irregular point series. The other main classes can also be further decomposed, and all classes can be arranged in a hierarchy (an extended version of figure 22) showing how they are related.

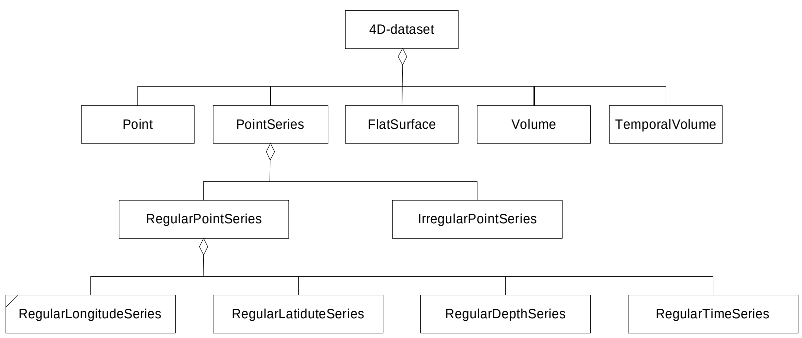

Figure 22 Main classes of 4D datasets

Concluding the description of this spatio-temporal object oriented data model, we must state that operations for creation, updating and deletion of objects are allowed, as well as constraints and relationships between different types of data sets can be specified. Finally in the implementation proposal of the author has been shown that it is possible to integrate both vector and raster data along with non-spatial data in this spatio-temporal model.

The key elements of the Object Model were also used in [TPH98] to develop the Geographic Object Model (Geo-OM). A geographic object's position, space varying dynamic attributes, spatial relationships and operations on objects, are captured here through added object classes and associations, for spatially referenced data. The proposed model allowed the representation of static and dynamic properties of spatial databases. Properties captured were not only the attributes and relationships, but also the operations on objects. The methodology used for the Geo-OM development showed that the object-oriented approach could be readily applied to capture both spatial static and dynamic properties, which can then also be referred to the temporal semantics.

[Ren97b] used the OMT object modeling language to represent the spatio-temporal phenomena. Initially, a temporal base model was developed, to represent the entities and properties through temporal objects and their descriptors. Here, the static states of a descriptor are represented by versions and the dynamic states are represented by transitions. These static and dynamic descriptors are subsequently represented by history graphs, introduced in [Ren96]. Additionally, transitions are defined at the object level, together with their operations, to provide the object links along the time line. The temporal base model was extended to the Spatio-Temporal Object Model (STOM) as shown in figure 23.

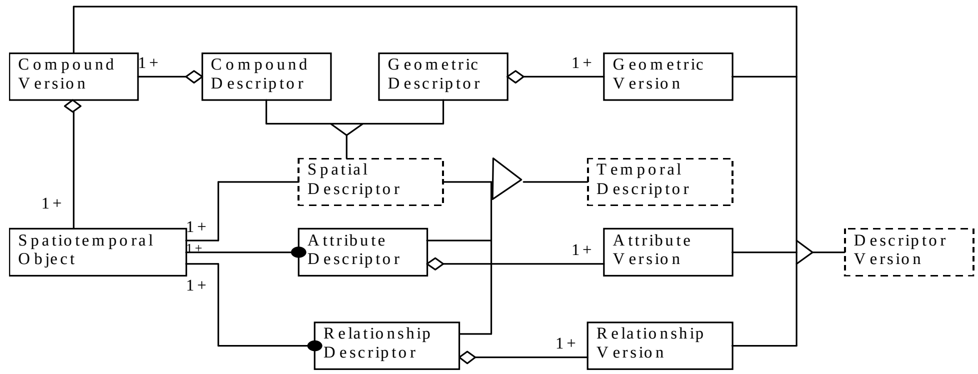

Figure 23 The Spatio-Temporal Object Model (STOM)

Each geometric descriptor comprises of a simple point, line and area object whilst complex objects are represented by a compound descriptor. Object characteristics are represented by attribute and relationship descriptors. The STOM can be extended by the use of enhanced symbols (fig. 24) to illustrate the spatio-temporal object semantics of the model.

Figure 24 Extensions of STOM schema

The STOM allows capturing the characteristics and the properties of spatio-temporal objects independently; hence the different behaviour of objects can be represented. Objects, which are considered temporal, have spatial attributes to describe their geometric shape, their relationship with other objects and their size. Although the model handles spatio-temporal behaviour of the objects, and any behavioural change is readily implemented through the object oriented approach, the notation is not concise, clear, and consistent enough to meet the demands of dynamic spatio-temporal modeling.

Worboys investigated the possibility of using the object-oriented approach in modeling dynamic spatial systems in order to analyse the fundamental constructs of “change”, “event” and “process” in an attempt to develop a foundation model for spatio-temporal databases. In [Wor98] comparisons were made between events, change and processes associated with the object-oriented modeling constructs, highlighting the necessity for additional concepts in order to handle the dynamic spatial applications. Raza and Kainz [RK99] propose a conceptual data model as an extension to Worboy’s model [Wor94a], to conceptually separate out space, time and attributes, and consolidate them in a modular fashion as distinct elements.

More recently, in [HE00] the authors describe a visual language for the explicit description of change relating to objects, called the Change Description Language. This work focuses on the appearance/disappearance of entities and particularly of the identity transitions from one object to another that are possible. Identity is seen as a means of tracking and querying the existence of specific objects and types of objects independent of specific attribute values. In [Bon99] the Lvis language, originally designed for visual query of spatial data [AB99], was extended to provide spatio-temporal visual query capabilities. These visual queries are implemented by translation into an SQL language with spatio-temporal extensions.

The Unified Modeling Language [FS98] is another object-oriented standard that is recognized by the Object Management Group [OMG95]. The UML was extended to give the *Spatio-Temporal UML (STUML)* [PTJ00] in an attempt to develop a high standard spatio-temporal database language, which would be capable to support a range of spatio-temporal models and data types using a clear, simple, and consistent notation. Extending the OMG standard for OO modeling was selected as the best approach given its high level of acceptance, tool support, understandability, and extensibility. An earlier spatio-temporal extension of the UML was proposed in [PSR99]. The authors defined constructs and corresponding graphical notations that provide support for modeling spatio-temporal properties. However, neither the syntax nor the semantics of the symbols introduced were presented formally.

The extended Spatio-Temporal UML in [PTJ00] maintains language's clarity and simplicity by introducing a small base set of fundamental modeling constructs: spatial, temporal, and thematic. These constructs can then be combined and applied at attribute, attribute group, association, and/or class levels of the object-oriented model; where the attribute group is an additional construct introduced for attributes with the same spatio-temporal properties. A formal functional specification of the semantic modeling constructs and their symbolic combinations is given and an application example is used to illustrate the flexibility of this approach. Figure 25 demonstrates the STUML model, applied to the LIS. The simplicity and clarity of the model notation offer advantages over other models that have been proposed.

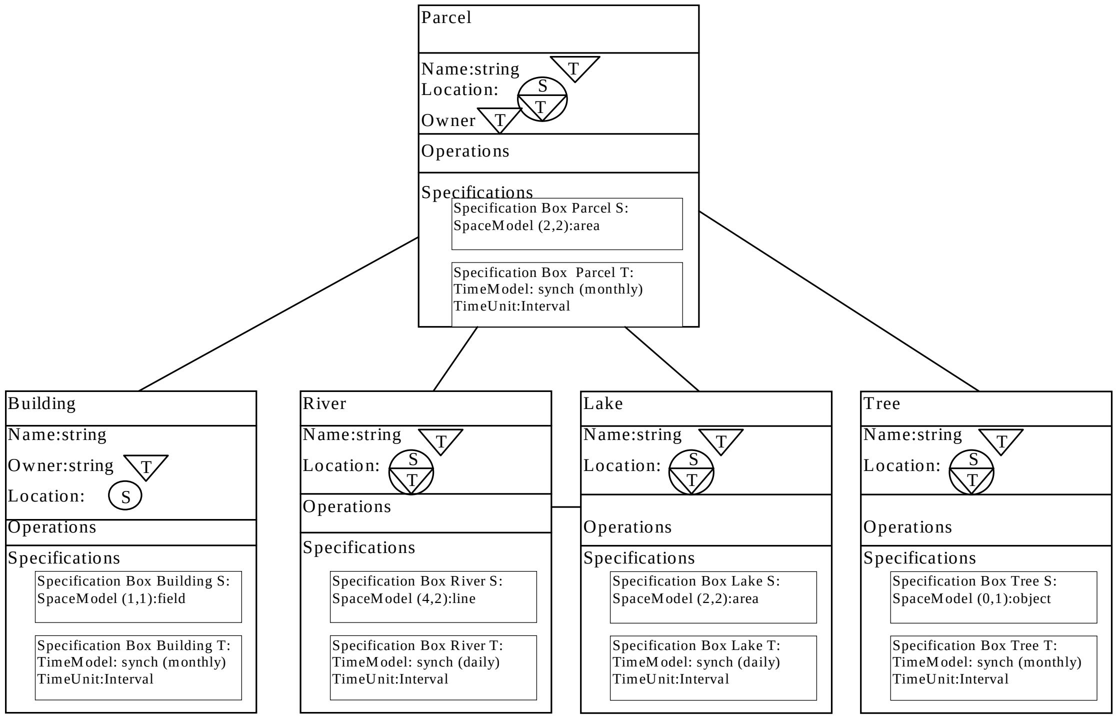

Figure 25 The LIS modelled with STUML

3.10 Moving Object Data Models

Clearly, when we try an integration of space and time, we are dealing with geometries changing over time. In general, geometries cannot only change in discrete steps, but continuously, and then we are talking about moving objects. If only the position in space of an object is relevant, then moving point is a basic abstraction; if also the extent is of interest, then the moving region abstraction captures moving as well growing or shrinking regions. Researchers have tried to model spatio-temporal databases using this concept of moving object. More specifically, in [EGS+99] the authors propose a new line of research where moving points and moving regions are viewed as three-dimensional (2D + time) or higher dimensional entities whose structure and behaviour is captured by modeling them as abstract data types. Such abstract data types for moving points and moving regions have been introduced in [GBE+00], together with a set of operations on such entities. The authors argue that such a collection of types and operations, together with a number of related auxiliary data types, as pure spatial or temporal types and time-dependant real numbers, can be integrated into any extensible DBMS.

This approach models the time as integral part of the spatial entities. The time dimension is based on the linear, discrete/continuous, absolute time model and initially only valid time is considered. The model captures both change and movement and the authors have shown that it is well suited for all types of spatio-temporal queries. Objects within the LIS case study can be represented as follows:

Parcel (name: string, owner: string, area: mregion)
Building (name: string, owner: string, area: mregion)
Lake (name: string, area: mregion)
River (name: string, route: mline)

Tree (name: string, centre: mpoint)

The types defined as mpoint, mline and mregion are closed and consistent, and carry the temporal value together with the spatial value. The operations introduced in the model can be used to describe LIS processes. For example:

Operation (mregion-> mreal) **area** applied to lake
  Returns a time-varying real number representing the size of the lake at all times
Operation (mregion x mpoint -> mboolean) **inside** applied against a parcel and a tree
  Computes a time-varying boolean representing when the tree has been moved or planted in a parcel
Operation (mpointXmpoint -> mreal) **distance**
  Calculates the time-varying distance between two trees at all times

Compared to the models proposed earlier, the representation of moving objects in a spatio-temporal database is data-type oriented, with emphasis on *generality*, *closure* and *consistency*. In addition, the abstraction level is much higher and we consider that this is the first attempt to deal with continuous motion introducing functions dealing with actual values of attribute data types. The definition of continuity is expressed functionally and the operations between spatial and temporal data types not considered in previous works are dealt with here.

The model presented in [GBE+00] lacks design and implementation. As such in [FGN+00] is presented the definition of the discrete representation of the above-discussed abstract data types. The interesting part of the discrete model is how *"moving"* types are represented. The authors describe the *sliced representation* behind which, the basic idea is to decompose the temporal development of a value into fragments called *"slices"* such that within the slice this development can be described by some kind of *"simple"* function. This is illustrated in figure 26 for a time-varying real number.

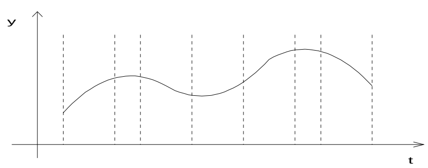

Figure 26 Sliced Representation

The sliced representation is built by a type constructor *"mapping"* parameterised by the type describing a single slice which we call a *unit* type. A value of a unit type is a pair *(I, u)* where *I* is a time interval and *u* is some representation of a simple function defined within that time interval. More specifically, they are defined the unit types *ureal*, *upoint*, *upoints*, *uline* and *uregion*. For values that can only change discretely, there is a *"const"* type constructor, which produces units, whose second component is just a constant of the argument type. This is in particular needed to represent moving integers, strings and boolean values. The *"mapping"* data structure basically just assembles a set of units and makes sure that their time intervals are disjoint.

The next step in this development was the study of algorithms for the rather large set of operations defined in [GBE+00]. Whereas [FGN+00] just provides a brief look into this issue by presenting two example algorithms at the end, in [LFG+03] the authors present a comprehensive, systematic study of algorithms for a subset of the operations introduced in [GBE+00]. Whereas some algorithms are relatively straightforward and simple, there is still a considerable number of quite involved ones, whose soundness and ability to be implemented is difficult to be justified by the reader due to that they are provided in terms of pseudo-code. In all cases the authors analyze the complexity of the algorithms. In [LFG+03] the data structures from [FGN+00] are also refined and extended by auxiliary fields to speed up computations. This paper also offers a blueprint for implementing such a *"moving objects"* extension package for suitable extensible database architectures. More specifically, the details and the current status of a prototypical implementation of the data structures and algorithms described are presented. The prototype is being developed as an algebra module for the experimental database system SECONDO [DG00] and as an Informix Datablade.

As an extension to the abstract model in [GBE+00], [ES99b] introduced the concept of *spatio-temporal predicates*. The goal was to investigate temporal changes of topological relationships induced by temporal changes of spatial objects. A corresponding spatio-temporal query language incorporating these concepts was presented in [ES99a]. Further work on modeling includes [SXI01] where the authors focus on moving point objects and the inclusion of concepts of differential geometry (speed, acceleration) in a calculus based query language. In [VW01] the authors consider movement in networks and some evaluation strategies.

An interesting approach following the paradigm of moving objects was also presented in STAU project [Pel02]. This research was focused on the representation and querying of continuously as well as discretely moving objects similar to those presented in [GBE+00]. From a theoretical point of view the author proposed a datatype-oriented model that supports the representation of objects both under object-oriented and object relational platforms. From a technical point of view, two data cartridges under the Object-Relational DBMS of ORACLE were developed. The first cartridge provides pure temporal functionality implementing TAU Temporal Types [KT96a]. The second cartridge supports the previously mentioned spatio-temporal data types and has been implemented by merging the temporal cartridge with Oracle's Spatial cartridge that offers an integrated set of operations on pure spatial data. The resulted system supports a wide set of object methods that extend Oracle's query language PL/SQL with spatio-temporal semantics. The overall architecture has been successfully tested and demonstrated in a case study related to truck motion analysis (**T**ruck **I**nformation **S**ystem TIS).

Another model using moving objects is proposed by the group of Wolfson in [SWC+97], [WXC+98] and [WSC+99]. The authors propose a data model called **M**oving **O**bjects **S**patio-**T**emporal data model (MOST) for databases with *dynamic attributes*, i.e. attributes that change continuously as a function of time, without being explicitly updated. This model enables the DBMS to predict the future location of a moving object by providing a *motion vector*, which consists of its location, speed and direction for a recent period of time. In the model, the answer to a query depends not only on the database contents, but also on the time at which the query is entered. As long as the predicted position based on the motion vector does not deviate from the actual position more than some threshold, no update to the database is necessary. An important issue here is, to balance the cost of updates against the cost of imprecise information. The authors also offer a query language (Future Temporal Logic FTL) based on temporal logic to formulate questions about the near future movement. The approach is restricted to moving points and does not address more complex time-varying geometries such as moving regions.

The dynamic attribute in the MOST model changes over time according to some given function, even if it is not explicitly updated. For example, consider a moving object whose position in two-dimensional space at any point in time is given by values of the x, y coordinates. Then each one of the object's coordinates is a dynamic attribute. Formally, a dynamic attribute A is represented by three sub-attributes, *A:value*, *A:updatetime* and *A:function*, where *A:function* is a function of a single variable t that has value 0 at $t=0$. The value of a dynamic attribute depends on the time, and it is defined as follows: At time *A:updatetime* the value of A is *A:value* and until the next update of A the value of A at time $A{:}updatetime + t_0$ is given by $A{:}value + A{:}function(t_0)$. The MOST model assumes that valid and transaction time is equal.

4 Comparing Spatio-Temporal Data Models

The review of existing spatio-temporal data models identifies common directions of modeling in the area and this becomes clearer if there is a certain comparison framework. As such, the following sub-sections include comparison tables that provide a general, comparative view of the spatio-temporal data models discussed in this paper. These tables provide an overall view of the presented models and they cover all the four categories of criteria and requirements that evaluate spatio-temporal modeling techniques as these where presented in the beginning of the paper.

4.1 … in terms of temporal semantics

As we can see in table 4 all of the models support multiple granularities except moving objects whose approach is directly associated with the finest granularity. Additionally, the support of special operations describing temporal relationships between spatio-temporal evolving entities is considered insufficient and further technical development is required in order to be incorporated into existing models. Most of the approaches assume discrete modeling of time, while the rest of them are continuous models, which include discrete temporal density as a subcase. However, both levels of modeling are needed even though we cannot store and manipulate continuous models in computers. If we restrict attention directly to discrete models, there is a danger that a conceptually simple, elegant design of up-to-date query operations [The03] will be missed. This is because the representational problems might lead us to prematurely discard some options for modeling.

Moreover, in the majority of the models time is incorporated by time-stamping objects. However other models timestamp events, whereas the history graph model timestamps both objects and events. The tactic that is followed by the moving objects approach where time is incorporated as integral part of the spatial entities in terms of simple continuous functions is the best method that should be followed by a new, generic spatio-temporal model. Most of the models support transaction time while valid time is considered essential for every approach. New proposals of modeling spatio-temporal data not only should include bitemporal time but user-defined time as well. All the models assume that the time axis is linear, while their behaviour is inadequate in some cases where the non-linearity of the time axis is needed (e.g. areas which need to reason in alternative futures). As such, branching, periodic and cyclic time order should be studied in future approaches. Finally, except the earlier introduced models there is a sufficient behaviour in handling the duration of the events and the history of the spatio-temporal objects (lifespan).

Spatio-Temporal Data Models	***Temporal Granularity***	***Temporal operations***	***Temporal Density***	***Representation of Time***	***Transaction TT vs Valid VT vs Bitemporal Time BT***	***Time Order***	***Lifespan***
Snapshot	Multiple	NO	Discrete	Modeling: Layers- Snapshots Time as: Attribute of location & Absolute reference	VT	Linear	NO
STC	Multiple	NO	Discrete	Modeling: Polygon history Time as: Integral part of spatial entities & Relative reference	BT	Linear	NO
Simple Time-stamping	Multiple	YES	Discrete	Modeling: Object's Creation-Cessation Time as: Attribute of the object & Absolute reference	VT	Linear	YES
Event Oriented	Multiple	NO	Discrete	Modeling: Events, change Time as: Attribute of an event & Relative reference	VT	Linear	YES
3-Domain	Multiple	YES	Discrete	Modeling: Temporal versions Time as: Independent object & Absolute/ Relative reference	BT	Linear	NO
History Graph	Multiple	NO	Discrete/ continuous	Modeling: Events, processes Time as: Attribute of objects, events & Absolute/ Relative reference	BT	Linear	YES
STER	Multiple	NO	Discrete	Modeling: Entity change Time as: Attribute of entity, relat/ship & Absolute reference	BT	Linear	YES
O-R	Multiple	NO	Discrete/ continuous	Modeling: ST phenomena Time as: Attribute of Object & Absolute/ Relative reference	BT	Linear	YES
O-O	Multiple	YES	Discrete/ continuous	Modeling: Object Change Time as: Attribute of object & Absolute/ Relative reference	BT	Linear	YES
STUML	Multiple	NO	Discrete/ continuous	Modeling: TimeUnit Time as: Via the Specification box & Absolute reference	BT	Linear	YES
Moving Objects	Single	YES	Discrete/ continuous	Modeling: Functions Time as: Integral part of spatial entities & Absolute reference	VT	Linear	YES

Table 4 Comparing existing spatio-temporal data models in terms of temporal semantics

4.2 … in terms of spatial semantics

Before stating specific conclusions rising form table 5, we should first point out that new proposals in the area of spatio-temporal database modeling should be based on spatial data models that present sufficient behaviour in all different spatial requirements. The majority of the models assume vector structure of space while the raster structure is adopted by four models. Even though there are applications that need raster structure of space (e.g. multimedia), the latter is considered obsolete. Vector representation is much simpler and generic and what is more it can easily be transformed to raster structure. Orientation and direction of spatio-temporal objects during their change or movement across space and time dimensions are only discussed in the object-oriented approaches and in the models with moving objects. These two approaches are the only models that support all types of spatial data handling, in terms of measurement, topological relationships and operations not necessarily concerned with relative positions of objects. Early approaches like the snapshot, the space-time composite and the simple time-stamping models address none of these issues, while the same stands also for models that are more conceptually oriented as the STER, the Object-Relationship model and the STUML.

Spatio-Temporal Data Models	*Structure of space*	*Orientation/ Direction*	*Measurement*	*Topological Relationships*
Snapshot	Raster	NO	NO	NO
STC	Vector	NO	NO	NO
Simple Time-stamping	Vector	NO	NO	NO
Event Oriented	Vector/Raster	NO	NO	YES
3-Domain	Vector	NO	NO	YES
History Graph	Vector	NO	NO	NO
STER	Vector	NO	NO	NO
O-R	Vector	NO	NO	YES
O-O	Vector/Raster	YES	YES	YES
STUML	Vector	NO	NO	NO
Moving Objects	Vector/Raster	YES	YES	YES

Table 5 Comparing existing spatio-temporal data models in terms of spatial semantics

4.3 … in terms of spatio-temporal semantics

The comparison in terms of the spatio-temporal semantics is more general and comprises factors related to the model domain. The most common used data types are the point, line, region, temporal point and temporal interval, while there are approach-specific data types like parametric rectangles and moving points. Most of the models capture some aspects of the notion of change as described in the requirements analysis subsection but only few of them support all eight possible types of change (see figure 3). Almost all models are capable to handle discrete changes, while movement and continuous change in early approaches was not in the priorities of the spatio-temporal database research community. However, the trend in more recent approaches is to support these two concepts and to provide functionality that is directly associated with them, meaning operations related to the evolution of spatio-temporal objects.

In terms of manipulation of the object identification, it is obvious that it is supported by the models that deal with spatio-temporal objects as unified objects and not as entities with non-interrelated components. Even though moving objects are unified objects, due to their nature there is no meaning talking about identification of such kind of objects. What is more, only object-oriented and moving object approaches maintain methods and functions for defining the topology and measuring the evolution of spatio-temporal objects in space and time, while all other models have inadequate behaviour in these issues. Finally, most of the models are capable to capture different dimensionality representations of spatio-temporal objects, while the others can do that by slight changes and extensions.

Spatio-Temporal Data Models	Data Types	Primitive Notions	Type of Change All 8 types: A/N Discrete: D/N Continuous: C/N Movement: M/N	Evolution in Time & Space	Measurement/ Topology	Object ID	Dimensionality
Snapshot	Native	Layer	N-N-N-N	NO	NO	NO	All
STC	Polygon	Polygon mesh	N-D-N-N	NO	NO	NO	2D
Simple Time-stamping	Parcel-Polygon, temporal point	Creation-Cessation time of an object	N-D-N-N	NO	NO	NO	2D
Event Oriented	Date, event & raster structures	Event, amendment maps	N-D-N-N	NO	NO	NO	All
3-Domain	Point, line, region, time point, interval	Thematic, spatial& temporal objects	A-D-C-M	NO	YES	YES	All
History Graph	Point, line, region, time point, interval	Object's version & transition	N-D-C-M	NO	NO	YES	All
STER	Point, line, region	Entity, relation ship	A-D-C-N	NO	NO	NO	2D
O-R	Point, line, area, temporal point	Object, event, process, relat/ship	A-D-C-N	NO	NO	YES	All
O-O	Simple descriptors, point, line, area	Object, attribute, relation-ship	A-D-C-M	YES	YES	YES	All
STUML	SpaceModel of Specification box	Object, Groups, Specification box	A-D-C-M	NO	NO	YES	All
Moving Objects	Moving point, region	Sliced representation	A-D-C-M	YES	YES	NO	2D

Table 6 Comparing existing spatio-temporal data models in terms of spatio-temporal semantics

4.4 … in terms of query capabilities

Comparing existing approaches in terms of their query capabilities we conclude that all models have adequate support for attribute and simple queries in all three categories. The snapshot, the space-time composite and the simple time-stamping models are the simplest ways to represent spatio-temporal information but their capability to support composite queries is the most limited. The problems with the existing approaches begin when dealing with spatial and/or temporal relationships and especially when the need is to describe the complex and evolving spatio-temporal behaviour of real world objects. An obvious weakness where research is currently taking place is on the Nearest-Neighbour type of queries, as well as on queries used in applications where the time dimension is considered to be continuous. What is more, there are models like the history graph, the three-domain and the object-oriented approaches, which support a variety of spatio-temporal queries, but at the end, although they have great potential, they can not cover all possible physical processes and phenomena because of the limited, non-specialized operation set provided by these approaches. Currently, this feature is presented only by moving objects, which can resolve any type of spatio-temporal query.

Spatio-Temporal Data Models	Spatial queries					Temporal queries			ST-queries		
	Attribute	**Point**	**Range / Distance-based**	**Nearest-Neighbor**	**Topo-logical**	**Simple**	**Range**	**Relation-ships**	**Simple**	**Range**	**Behavior**
Snapshot	Yes	No	No	No	No	Yes	No	No	No	No	No
STC	Yes	Yes	No	No	No	Yes	No	No	Yes	No	No
Simple Time-stamping	Yes	Yes	No	No	No	Yes	Yes	No	Yes	No	No
Event Oriented	Yes	Yes	No	No	Yes	Yes	Yes	No	Yes	No	No
3-Domain	Yes	Yes	No	No	Yes	Yes	Yes	Yes	Yes	No	No
History Graph	Yes	Yes	No	No	No	Yes	Yes	Yes	Yes	No	No
STER	Yes	Yes	Yes	No	No	Yes	Yes	No	Yes	No	No
O-R	Yes	Yes	Yes	No	Yes	Yes	Yes	No	Yes	No	No
O-O	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	No
STUML	Yes	Yes	Yes	No	No	Yes	No	No	Yes	No	No
Moving Objects	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes

Table 7 Comparing existing spatio-temporal data models in terms of query capabilities

5 Discussion - Conclusions

In this section the above presented syncretic framework is appended by a discussion declaring the implementation status of each data model. The reviewed models vary on completeness, formalization and implementation. In addition some models incorporate different spatial and temporal database models to achieve the final objective and others have proposed their modeling approach from scratch. In the sequel we provide a concluding comparative table with the following features:

- ***Formalisation***: This factor shows if the model has been formally defined or not.
- ***Implementation***: This criterion depicts if the model has been implemented or not.
- ***Tool***: If the model has been implemented, the name of the tool developed is listed here.
- ***Application***: The case study used to analyse and develop the model.
- ***Spatial model***: The spatial model used as the basis for the development of the corresponding spatio-temporal model.
- ***Temporal model***: The temporal model used to define and develop the spatio-temporal model.

Spatio-Temporal Data Models	*Formal-isation*	*Implem-entation*	*Tool*	*Application*	*Spatial model*	*Temporal model*
Snapshot	No	No	None	LIS	GIS	N/A
STC	No	No	None	LIS	GIS	N/A
Simple Time-stamping	No	No	None	Historical cadastral database	US spatial data transfer standard	N/A
Event Oriented	Yes	Yes	TEMPEST	LIS	GIS with event dates	Ordered time models
3-Domain	Yes	No	None	LIS	Relational Spatial database	Relational Versions Table
History Graph	No	No	None	LIS	N/A	Graphs
STER	Yes	No	None	Cadastral application	Spatial indicators	Temporal indicators
O-R	Yes	Yes	MADS	Rural-urban land use application	MEOSIG	MODUL-R POLLEN
O-O	Yes	Yes	Geo-OM	LIS	ERT model	Temporal base model
STUML	Yes	No	None	Regional health care example	Spatial indicators	Temporal indicators
Moving Objects	Yes	Yes	SECONDO module & STAU Oracle data cartridge	Multimedia scenario, Forest fire control management	Abstractions of spatial data types & Oracle Spatial	Abstractions of temporal data types & TAU Types

Table 8 Comparing existing spatio-temporal data models in terms of implementation state

As it can be implied from table 8, not all models have been formally defined so as to facilitate a spatio-temporal database designer and developer in his tasks. Furthermore, some of the models (the STER and the STUML), even though they have been formalized, their modeling power remains at the conceptual level and as such the design concepts are not supported and associated with corresponding technical constructs. The problem whether a model has been implemented or not and moreover whether there is an appropriate application used to analyse and evaluate each model is very important and is closely related to how expressive and usable a model can be. The fact that more than half of the existing approaches have not been implemented, that some of them that have been implemented are incomplete or in an experimental stage and that some models are evaluated for the first time by the case study introduced in this paper (LIS), shows that the spatio-temporal research community has to focus on this area the forthcoming years. Finally, the last two columns in table 8 prove that a functional, flexible and extensible pure spatial and temporal data model is very significant to exist not only to be able to introduce an efficient spatio-temporal data model, but to be able to utilize it in the corresponding domains separately as well. In other words, it is very important to be able to represent spatial entities and to have complete functionality on them independently to the time dimension and the opposite.

The following paragraphs consider in depth the spatio-temporal research agenda that is important in the context of computing science, in terms of the requirements that determine the priorities of modeling spatio-temporal applications and highlight key issues from those of lesser significance.

Undoubtedly, spatio-temporal database modeling requires a specific approach, for reasons that include:

- Spatio-temporal databases typically deal with large complex bodies of spatially referenced data, which is required to be readily available [Arm92]. As such indexing techniques for space-time databases and more specifically for real-time applications that describe continuously evolving spatial entities; are still an important and open research area.
- A sizeable proportion of the data is either regularly/irregularly updated from external data sources or need to be continuously updated due to evolution of natural processes [Haz92].
- Some of the data are noisy, conflicting and incomplete. More analytically, a major problem with spatial data is the control of error propagation under spatial operations. Further research is needed on finite precision geometry and multiple resolution techniques [Peu01].
- The ontology of space-time deals with the nature of space, of time, and their unique interactions. Investigation at this highly abstract level is essential in order to develop a common conceptual framework [TJ99]. Issues specific to time; of linear vs. cyclic views of time, multiple times (world, valid, user-defined time), continuous or discrete change, branching and alternative timelines, etc., seem to be recognized and well understood in a database context [Lan92].
- The spatial and temporal dimensions should be considered separately and incorporated into the database design. Temporal and spatial characteristics of the modeling reality should be able to capture all the aspects of the real world. What is more, thematic characteristics of spatio-temporal objects should be identified and modelled together with the spatial and temporal characteristics. This issue is not a problem for traditional database modeling; however spatial changes can lead to losing a thematic description of a particular dynamic object [PD95].
- The historic flow of information and the evolution and dynamism of the spatio-temporal objects should be modelled, to establish a realistic representation of the dynamic world over time. Dynamic applications should be modelled in sympathy with the real world, which will simplify its modeling in the information system.
- Complex functions and calculations involving operators, relationship status [ES99a] and objects for the prediction of their future motion need to be designed [WXC+98]. Research is also needed to be carried out on applications of newer computational paradigms, such as constraint-based approaches [GRS01], fuzzy sets [WHS90] and rough sets [AKO00].
- Rapid decision-making must be supported. In this direction, graphical interfaces of the spatio-temporal information systems should be enhanced to help users to take advantage of the newly incorporated operations. Additionally, research must be driven towards the introduction of novel database query languages, metaphors, visualization methods and approaches to metadata handling [Bon99]. Data mining and exploratory uses of information systems hold much promise [HK01]. In the literature, there exist a number of algorithms and techniques supporting a series of kinds of knowledge extracted from traditional databases (e.g. classification, prediction, time series analysis, relevance analysis, sequential patterns, clustering). However, in the field of spatio-temporal databases, related work is limited and novel techniques have to be developed.
- A growing number of researchers in both the DBMS and GIS communities have come to the realization that a general, application-independent solution that allows an optimal combination of simplicity, flexibility and efficiency will require rethinking at an abstract level and new types of implementational data models and associated query languages [Ren97a]. This solution needs to be based upon a uniform ontological framework and requires a multi-representational approach.
- The requirement of multidimensional modeling in conjunction with the large bodies of geographically referenced data poses a range of new challenges in the context of the data warehousing techniques [JKP+04]. Firstly, due to that different dimension values (temporal, spatial, thematic) are partitioned into categories of values, and categories are related via containment relationships, explicit hierarchies would be highly useful in data analysis as they are used for aggregating data to the right level of detail in exploratory analyses that use roll-up/drill-down operations. Such a multidimensional data model should accommodate spatial values that exhibit partial containment relationships instead of the total containment relationships assumed till now. This is needed to increase the modeling power of the model and to enable new kinds of

queries. The above discussion arises specific issues as the normalization of the hierarchies as well as the handling of imprecision that is introduced by partial containment in the aggregation paths.

- Ø The nature of concepts that are inherently both spatial and temporal (e.g. motion) and the interactions of the spatial and temporal dimensions need to be better understood in order to develop truly effective and robust space-time data structures, query languages and user interfaces. Part of the fundamental difficulty that has to be overcome, is that there is an increasing degree of mismatch between the nature of the data as seen conceptually from an application-oriented and implementation-oriented views used in both the DBMS and GIS communities [PSZ99].
- Ø The means to a better "fit" is in part being provided by the object-oriented [TJ00] and object-relational [GBE+00] approaches, which support flexible constructs such as Abstract Data Types (ADT) and Moving Objects Databases (MODs). However, there remains a lack of both theoretical and technical framework that explicitly describes the fundamental spatio-temporal characteristics in a context-independent way from which to draw application and user-oriented views.
- Ø It is not an exaggeration to say that the so-called MODs are (or soon will be) ubiquitous. As the number of mobile commerce or, in general, mobile services, increases rapidly everyday, the need for *powerful conceptual models* and *robust management systems* about continuously time-varying location data are vital. Additionally, the rapid development of mobile devices (e.g. 3G cell-phones, Personal Digital Assistants PDAs) and of techniques for identifying their location (e.g. Global Positioning System GPS) have emerged the special category of Location Based Services (LBS) that exhibit particular scientific interest.

Concluding, spatio-temporal database modeling on earlier stages attempted to capture the state of real world objects or the physical events upon them, on the time line. However, investigations in real world applications brought out new directions and requirements for further development. Thus, the processes of continuous change on the time line, changes in description/size/position/extent of entities were investigated and models to capture them were proposed. At the same time various types of spatio-temporal attributes were defined and the importance of enhanced relationship operations recognised. The models on a later stage are more concerned with conceptual notation of spatio-temporal data. Next step in the spatio-temporal database development is the testing stage, where the models proposed run on different applications, to identify further requirements and research directions.

In this paper, the majority of spatio-temporal database models proposed to model geo-referenced real world concepts, phenomena and processes were studied and essential problems bothering the spatio-temporal research community were identified. Continuous motion modeling, multiple time line representation, complex behavioural spatio-temporal queries and very large data manipulation are some of the open issues for further investigation and development. Although attempts were made to cover such problems, due to failure in selection of appropriate applications carrying all real world processes, changes, entities and characteristics, the models dealing with one issue are not capable to handle the others. Due to the complex structure of spatio-temporal databases the methods capable to control intricate entities are most promising in the field. The object-oriented and moving object methods permit the representation of such complex structures. Thus, the modification of spatio-temporal database models integrating these approaches promises sound and complete real world representation.

A comparison framework of existing spatio-temporal database models was also carried out, accentuating on the specific spatial, temporal and spatio-temporal semantics of the models, while their corresponding query capabilities were also explored. At the same time the state of the art in spatio-temporal database modeling has been covered to identify the particular approaches and applications used at each model development and by that to envision the next steps in spatio-temporal database research.

6 References

[AB99] M.-A. Aufaure-Portier and C. Bonhomme, *"A High Level Visual Language for Spatial Data Management"*, Proceedings of Visual '99, 3rd International Conference on Visual Information Systems, Amsterdam, 1999.

[AKO00] O. Ahlqvist, J. Keukelaar and K. Oukbir, *"Rough Classification and Accuracy Assessment"*, International Journal of Geographical Information Science, Vol. 14: 475-496, 2000.

[All83] J .F. Allen, *"Maintaining Knowledge About Temporal Intervals"*, Communications of the ACM, 26(11):832–843, November 1983.

[AR99] T. Abraham and J.F. Roddick, *"Survey of Spatio-Temporal Databases"*, GeoInformatica, 3:61-99, 1999.

[Arm92] C. Armenakis, *"Estimation and Organization of Spatio-Temporal Data"*, Proceedings of the Canadian Conference on GIS92, Ottawa, Canada, 1992.

[ASS94] K.K. Al-Taha, R.T. Snodgrass and M.D. Soo, *"Bibliography on Spatio-Temporal Databases"*, ACM SIGMOD Record, vol. 22, pp. 59-67, 1994.

[BM94] F. Bonfatti and P. D. Monari, *"Spatio-Temporal Modeling of Complex Geographical Structures"*, Computer Support for Environmental Impact Assessment: Proceedings of the IFIP TC5/WG5.11 Working Conference on Computer Support for Environmental Assessment, CSEIA 93, Como, Italy, Elsevier, 1994.

[Bon99] C. Bonhomme, *"A Visual Language for Querying Spatio-Temporal Databases"*, Proc. of the 7th ACM Symposium on Advances in Geographic Information Systems, Kansas City, MO, 1999.

[BVH96] L. Becker, A. Voigtmann and K. Hinrichs, "*Temporal Support for Geo-Data in Object-Oriented Databases*", Proceedings of Database and Expert Systems Applications, 79-93, 1996.

[CPS+98] C.Claramunt, C. Parent, S.Spaccapietra, M. Theriault, *"Database Modeling for environmental and Land Use Changes"*, Geographical Information and Planning, Chapter 20, Springer-Verlag, 1998.

[CT95] B. Claramunt and M. Theriault, *"Managing Time in GIS: An Event Oriented Approach"*, Recent Advances on Temporal Databases, Clifford. J. and Atuzhilin. A, eds, Springer-Verlag, Zurich, pp.23-42, 1995.

[DG00] S. Dieker and R. H. Guting, *"Plug and Play with Query Algebras: Secondo. A Generic DBMS Development Environment"*, Proceedings of the Int. Database Engineering and Applications Symposium (IDEAS), pages 380-390, September 2000.

[EGS+99] M. Erwig, R.H. Guting, M. Schneider, and M. Vazirgiannis, *"Spatio-Temporal Data Types: An Approach to Modeling and Querying Moving Objects in Databases"*, GeoInformatica, 3(3): 265-291, 1999.

[ES99a] M. Erwig and M. Schneider, *"Developments in Spatio-Temporal Query Languages"*, In IEEE Int. Workshop on Spatio-Temporal Data Models and Languages (STDML), pages 441-449, Florence, Italy, 1999.

[ES99b] M. Erwig and M. Schneider, *"Spatio-Temporal Predicates"*, Technical Report 261, Fern Universitat Hagen, 1999. To appear in IEEE Transactions on Knowledge and Data Engineering

[FGN+00] L. Forlizzi, R. H. Guting, E. Nardelli, M. Schneider, *"A Data Model and Data Structures for Moving Objects Databases"*, Proceedings ACM SIGMOD Intl. Conf. on Management of Data, pages 319-330, 2000.

[FMN98] G. Faria, C.B. Medeiros and M.A. Nascimento, "*An Extensible Framework for Spatio-Temporal Database Applications*", Time Center Technical Report TR-27, 1-15, 1998.

[Fra92] A. Frank, *"Theories and Methods of Spatio-Temporal Reasoning in Geographic Space"*, Lecture Notes in Computer Science 639, Springer Verlag, 1992.

[FS98] M. Fowler and K. Scott, *"UML Distilled – Applying the standard object Modeling language"*, Addison-Wesley, 1998.

[GBE+00] R.H. Guting, M. H. Bohlen, M. Erwig, C. S. Jensen, N. A. Lorentzos, M. Schneider, and M. Vazirgiannis, *"A Foundation for Representing and Querying Moving Objects"*, ACM Transactions on Database Systems, 25(1): 1-42, 2000.

[GRS01] S. Grumbach, P. Rigaux and L. Segoufin, *"Spatio-Temporal Data Handling with Constraints"*, GeoInformatica, 5: 95-115, 2001.

[Gut94] R.H. Guting, *"An Introduction to Spatial Database Systems"*, VLDB Journal 4 (1994), 357-399.

[Ham94] T. Hamre, *"An object-oriented conceptual model for measured and derived data varying in 3D space and time"*, In Advances in GIS Research, Sixth International Symposium on Spatial Data Handling, 1994.

[Haz92] N. W. J. Hazelton, *"Developments in Spatio-Temporal GIS"*, Proceedings of the First Regional Conference on GIS Research in Victoria and Tasmania, 1992.

[HE00] K. Hornsby and M.J. Egenhofer, *"Identity-Based Change: A Foundation for Spatio-Temporal Knowledge Representation"*, International Journal of Geographical Information Science, Vol. 14: 207-224, 2000.

[HK01] J. Han and M. Kamber, "*Data Mining: Concepts and Techniques*", Morgan Kaufmann Publishers,

2001.
[HT97] T. Hadzilacos and N.Tryfona, *"An Extended Entity-Relationship Model For Geographic Applications"*, ACM SIGMOD Record, 26(3), 24-29., 1997.
[HW90] Gary J. Hunter and Ian P. Williamson, *"The Development of a Historical Digital Cadastral Database"*, Int. Journal of Geographic Information Systems, 4(2), 1990.
[JKP+04] C. S. Jensen, A. Kligys, T. B. Pedersen, I. Timko, *"Multidimensional Data Modeling for Location-Based Services"*, The VLDB Journal, 13: 1-21, 2004.
[KS+03] M. Koubarakis, T. Sellis et al. (eds.), *Spatio-Temporal Databases: The Chorochronos Approach*, 2003. Springer-Verlag LNCS 2520.
[KT96a] I. Kakoudakis, B. Theodoulidis, *"The TAU Temporal Object Model"*, TR no. TR-96-4, Timelab, UMIST, Manchester, UK, 1996.
[KT96b] I. Kakoudakis, B. Theodoulidis, *"The TAU Time Model"*, TR no. TR-96-5, Timelab, UMIST, Manchester, UK, 1996.
[Lan88] G. Langran, *"A Framework for Temporal Geographic Information Systems"*, Cartographica, 25 (3), 1988.
[Lan92] G. Langran, *"Time in Geographical Information Systems"*, ed. Taylor & Francis, London, 1992.
[LFG+03] J. A. C. Lema, L. Forlizzi, R. H. Guting, E. Nardelli, M. Schneider, *"Algorithms for Moving Objects Databases"*, The Computer Journal Vol. 46, No. 6, p. 680-712, 2003.
[Mon95] L. D. Montgomery, *"Temporal Geographic Information Systems Technology and Requirements: Where We are Today"*, Thesis, Department of Geography, Ohio State University, USA, 1995.
[OMG95] Object Management Group, *"Object Management Architecture Guide"*, Third Edition, John Wiley & Sons Publishing company, Framingham, MA, 1995.
[Pav98] A. Pavlopoulos, *"Evaluation and Comparison of Spatio-Temporal Data Models"*, MSc Dissertation, UMIST, 1998.
[PD95] D. Peuquet and N. Duan, *"An Event-Based Spatio-temporal Data Model (ESTDM) for Temporal Analysis of Geographical Data"*, Int. Journal of Geographical Information Systems, vol. 9, no. 1, pp. 7-24, 1995.
[Pel02] N. Pelekis, *"STAU: A spatio-temporal extension to ORACLE DBMS"*, PhD Thesis, UMIST, 2002.
[Peu01] D. Peuquet, "*Making Space for Time: Issues in Spase-Time Data Representation*", GeoInformatica, 5: 11-32, 2001.
[PSR99] R. J. Price, B. Srinivasan and K. Ramamohanarao, "*Extending the Unified Modeling Language to Support Spatiotemporal Applications*", Asia Technology of Object Oriented Languages and Systems, 163-174, 1999.
[PSZ99] C.S. Parent, S.Spaccapietra and E.Zimanyi, "*Spatio-Temporal Conceptual Models: Data Structures + Space + Time*", Proceedings of the 7^{th} ACM Symposium on Advances in Geographic Information Systems, Kansas City, MO, 1999.
[PTJ00] R. J. Price, N. Tryfona and C. S. Jensen, *"Extended SpatioTemporal UML: Motivations, Requirements and Constructs"*, Journal on Database Management, Special Issue on UML, 11, 4, Oct.-Dec. 2000.
[PW94] D. Peuquet and E. Wentz, *"An approach for Time-Based Spatial Analysis of Spatio-Temporal Data"*, Advances in GIS Research, Proceedings 1. p. 489-504, 1994.
[Ren96] A. Renolen, *"History Graphs: Conceptual Modeling of Spatio-temporal Data"*, In GIS Frontiers in Business and Science, Vol. 2, International Cartographic Association, Brno, Czech Republic, 1996.
[Ren97a] A. Renolen, *"Temporal Maps and Temporal Geographical Information Systems (Review of Research)"*, Department of Surveying and Mapping, The Norwegian Institute of Technology, 1997.
[Ren97b] A. Renolen, *"Modeling Spatiotemporal Information: The Spatiotemporal Object Model"*, Norvegian University of Science and Technology, Trondheim, Norway, 1997.
[RK95] E. Rojas-Vega and Z. Kemp, *"An Object-Oriented Data Model for Spatio-Temporal Data"*, Proceedings of the Ninth Annual Symposium on Geographic Information Systems, Vancouver, Canada, 1995.
[RK99] A. Raza and W. Kainz, *"Cell Tuple Based Spatio-Temporal Data Model: An Object Oriented Approach"*, ACM-GIS'99: Proceedings of the 7^{th} International Symposium on Advances in Geographic Information Systems, Kansas City, MO, 1999.
[RMD94] A. Ramachandran, F. MacLoad and S. Dowers, *"Modeling Temporal Changes in a GIS using an Object Oriented Approach"*, In Advances in GIS Research, Sixth International Symposium on Spatial Data Handling, 1994.
[Rum91] J. Rumbaugh, *"Object Oriented Modeling and Design"*, Prentice Hall, Englewood Cliffs, NJ, 1991.
[Sel99] T. Sellis, *"Research Issues in Spatio-temporal Database Systems"*, Proceedings of Advances in Spatial Databases: 6^{th} International Symposium, SSD '99, Hong Kong, China, 1999.
[SWC+97] P. Sistla, O. Wolfson, S. Chamberlain, S.Dao, *"Modeling and Querying Moving Objects"*,

Proceedings of the 13th International Conference on Data Engineering (ICDE13), Birmingham, UK, 1997.

[SXI01] J. Su, H. Xu and O. Ibarra, *"Moving Objects: Logical Relationships and Queries"*, Proceedings of the 7th Int. Symp. On Spatial and Temporal Databases (SSTD), pages 3-19, Redondo Beach, California, 2001.

[TCG+93] Tansel, A.U., J. Clifford, S. Gadia, S. Jajodia, A. Segev and R. Snodgrass, *"Temporal Databases: Theory, Design and Implementation"*, Benjamin/Cummings Publishing Company, 1993.

[The03] Y. Theodoridis, *"Ten Benchmark Database Queries for Location-based Services"*, The Computer Journal, 46(6):713-725, October 2003.

[TJ00] N. Tryfona and C. S. Jensen, *"Using Abstractions for Spatio-Temporal Conceptual Modeling"*, Proceedings of the 2000 ACM Symposium on Applied Computing, Como, Italy, 2000.

[TJ99] N. Tryfona and C. S. Jensen, *"Conceptual Data Modeling for Spatiotemporal Applications"*, GeoInformatica, Vol. 3: 245-268, 1999.

[TL91] I. Theodoulidis and P. Loucopoulos, *"The Time Dimension in Conceptual Modeling", Information Systems,* vol. 16, no. 3, pp. 273-300, 1991.

[TPH98] N. Tryfona, D. Pfoser and T. Hadzilacos, *"Modeling Behaviour of Geographic Objects: Experience with the Object Modeling Technique"*, University of Maine, Orono, USA, 1998.

[Try97] N. Tryfona, *"Modeling Phenomena in Spatiotemporal Databases: Desiderata and Solutions"*, Department of Computer Science, Aalborg University, Denmark, 1997.

[VW01] M. Vazirgiannis and O. Wolfson, *"A Spatiotemporal Model and Language for Moving Objects on Road Networks"*, Proceedings of the 7th Int. Symp. On Spatial and Temporal Databases (SSTD), pages 20-35, Redondo Beach, California, 2001.

[WH94] M. Wachowicz and R. G. Healey, *"Towards Temporality in GIS"*, In Worboys M. F. editor, Innovations in GIS, Taylor & Francis, 1994.

[WHM90] M. F. Worboys, H. M. Hearhshow, D. J. Maguire, *"Object-Oriented Modeling for Spatial Databases"*, Int. Journal of GIS ol. 4, No. 4, 1990.

[WHS90] F. Wang, G.B. Hall, and Subaryono, *"Fuzzy Information Representation and Processing in Conventional GIS Software: Database Design and Application"*, International Journal of Geographical Information Systems, Vol. 4: 261-283, 1990.

[Wor94a] M. F. Worboys, *"A Unified Model for Spatial and Temporal Information"*, The Computer Journal, Vol. 37, No. 1, pp. 27-34, 1994.

[Wor94b] M.F. Worboys, *"Unifying the Spatial and Temporal Components of Geographical Information"*, Information Symposium on Spatial data Handling, 1994.

[Wor98] M. F. Worboys, *"Modeling Changes and Events in Dynamic Spatial Systems with Reference to Socio-economic Units"*, Department of Computer Sciemce, Keele University, UK, 1998.

[WSC+99] O. Wolfson, A. P. Sistla, S. Chamberlain and Y. Yesha, *"Updating and Querying Databases that Track Mobile Units"*, Distributed and Parallel Databases, 7: 257-387, 1999.

[WXC+98] O. Wolfson, B. Xu, S. Chamberlain, L. Jiang, *"Moving Objects Databases: Issues and Solutions"*, Proceedings of the 10th Int. Conference on Scientific and Statistical Database Management, pages 111-122, Capri, Italy, 1998.

[Yua94] M. Yuan, *"Wildfire Conceptual Modeling for Building GIS Space-Time models"*, Proceedings of the GIS/LIS'94, pp. 860-869, 1994.

[Yua96a] M. Yuan, *"Temporal GIS and Spatio-Temporal Modeling"*, Published on CD-ROM by the National Center for Geographical Information and Analysis, 1996.

[Yua96b] M. Yuan, *"Modeling Semantical, Temporal, and Spatial Information in Geographic information Systems"*, M. Craglia and H. Couclelis eds. Geographic Information Research: Bridging the Atlantic, Taylor & Francis, pp. 334-347, 1996.

[Yua97] M. Yuan, *"Use of a Three-Domain Representation to Enhance GIS Support for Complex Spatio-Temporal Queries"*, Transactions in GIS, 1997.