The Knowledge Engineering Review, Vol. 19:3, 235–274. © 2004, Cambridge University Press
doi:10.1017/S026988890400013X Printed in the United Kingdom

Literature review of spatio-temporal database models

NIKOS PELEKIS,[1,2] BABIS THEODOULIDIS,[1] IOANNIS KOPANAKIS[1] and YANNIS THEODORIDIS[2]

[1]*Center of Research in Information Management (CRIM), Department of Computation, UMIST, P.O. Box 88, Sackville Street, Manchester M60 1QD, UK; e-mail: pele@ath.forthnet.gr, babis@co.umist.ac.uk, kopanak@csd.uoc.gr, http://www.crim.org.uk*
[2]*Department of Informatics, University of Piraeus, 80 Karaoli and Dimitriou Street, GR-18534 Piraeus, Greece; e-mail: npelekis@unipi.gr, ytheod@unipi.gr, http://www.unipi.gr*

Abstract
Recent efforts in spatial and temporal data models and database systems have attempted 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 characterised by both spatial and temporal semantics. Development and research in this area started decades ago, when the management and manipulation of data, relating to both spatial and temporal changes, was recognised 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 (Tansel *et al*., 1993) and spatial (Guting, 1994) 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, and reviews of this work 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 modelling, and clarifying the direction for further research and development. Standing at this point, the contribution, and simultaneously 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 trends as well as the needs for further research in the area.

Our attempt to classify most of the existing data models was facilitated by previous reviews in the field. One of the most significant contributors to the domain has been Langran (1992) who first looked at the aspects of time in geographic information systems (GISs). Although Langran's work covers many of the most important issues of spatio-temporal systems, a number of new proposals

https://doi.org/10.1017/S026988890400013X Published online by Cambridge University Press

have emerged since. In this survey, earlier review works are considered and included in order to capture all of the trends and ideas proposed in the domain of spatio-temporal database modelling. Frank (1992) was one of the first to attempt to present the poor (until then) theories and methods of reasoning in the time-varying spatial space, while a bibliography on spatio-temporal databases was published until 1994 in Al-Taha *et al.* (1994), which contains interesting pointers for further reference. More fruitful reviews of the domain were available the forthcoming years in Yuan (1996a), Renolen (1997a), Abraham & Roddick (1999), Sellis (1999), Peuqeot (2001), and especially in Pavlopoulos (1998), where the classification of spatio-temporal database models used in the current survey of the area is introduced.

2 Definitions on spatio-temporal data modelling

A delineated in the papers cited above, 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; and (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 (a), rather than the other three areas where secondary implementation issues are outlined.

Spatio-temporal data models are the core of a spatio-temporal information system (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 as a result of spatio–temporal behaviours not anticipated by the application used for the initial model development.

The study of the domain literature highlighted the fact that existing models possess a set of precise characteristics that match the requirements of the spatio-temporal database community. These requirements implicitly form an evaluation norm for spatio-temporal data modelling. From this study, common directions of modelling 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 that we should concentrate on and to choose the routes for subsequent improvements.

The above-mentioned requirements fall into four categories. The first category only deals 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. If, as we believe, these requirements are followed carefully in the process of designing a spatio–temporal database model, then it is possible to create a robust and expandable model that is capable of dealing with most 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 (Kakoudakis & Theodoulidis, 1996b). 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 (1983) was the

https://doi.org/10.1017/S026988890400013X Published online by Cambridge University Press

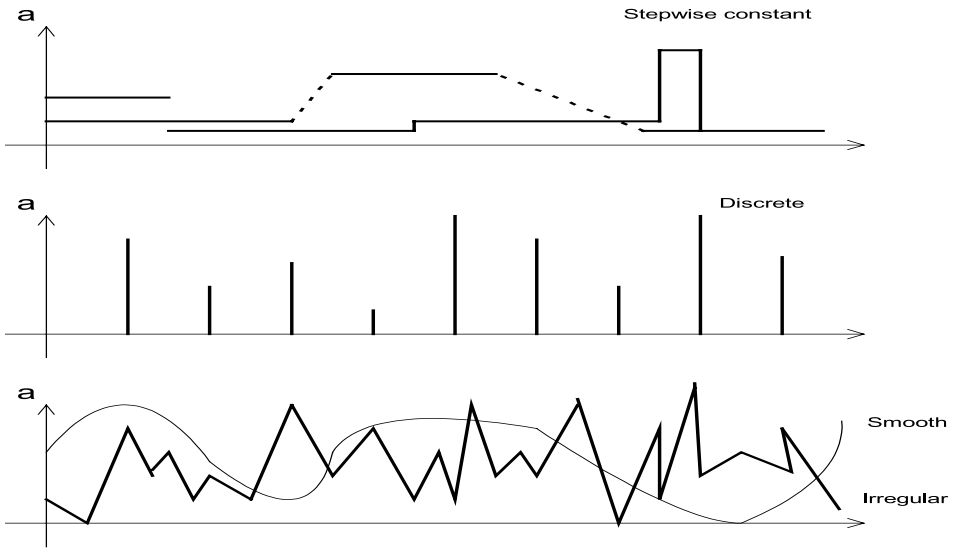


Figure 1 Types of changes according to time density

first to introduce such operations (e.g. timepoint T ''*inside*'' temporal period A, which ''*meets*'' period B).

Time density

This issue arises whether time should be modelled as discrete elements (isomorphic to the integers) or as continuous elements (isomorphic to the 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 change. A third type of entities exhibit *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 (Renolen, 1996).

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 modelling 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. Worboys (1994b) depicted 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 only allows 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.

https://doi.org/10.1017/S026988890400013X Published online by Cambridge University Press

Transaction/valid time
There are two different clauses that a model utilises 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 (Kakoudakis & Theodoulidis, 1996b).

Time order
Hazelton (1992) observed 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 (nonlinear) metaphors, namely branching and multi-dimensional time (Theodoulidis & Loucopoulos, 1991).

Lifespan
This factor shows whether 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 (Worboys, 1994b). *Raster* data are structured as an array of cells, pixels or voxels for two- or three-dimensional representations. 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 utilise 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 between existing spatio-temporal modelling techniques according to whether they support different topological relationships for real-world spatial objects. Hazelton (1992) listed all possible topological relationships between objects of various dimensions in up to four-dimensional space. Figure 2 depicts common 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.

https://doi.org/10.1017/S026988890400013X Published online by Cambridge University Press

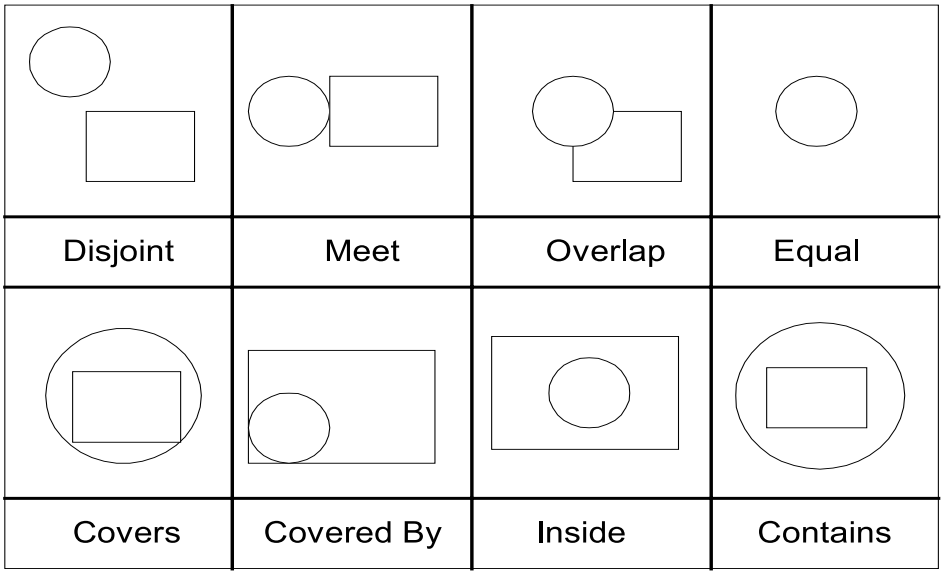


Figure 2 Topological relationships

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 also on the observations and choices of the particular modeller.

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 as to 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).

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 categorises existing approaches that follow the continuous paradigm is

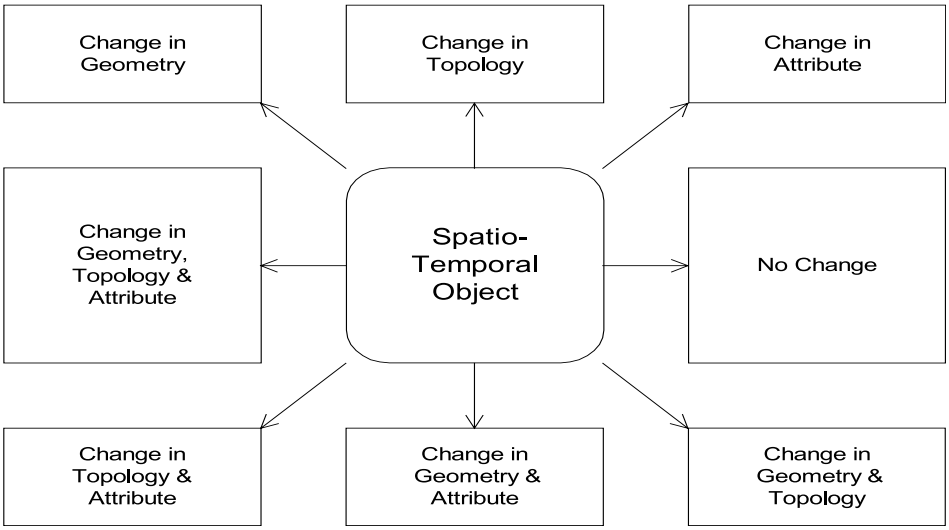


Figure 3 The eight possible types of change of a spatio-temporal object

https://doi.org/10.1017/S026988890400013X Published online by Cambridge University Press

whether the latter can deal with the movement of spatial objects over time. This is an extra decisive factor that differentiates models that support changes in the position and/or the extent of objects in the unified space-time continuum.

Evolution in time and space

This factor shows if there are defined functions such as *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. Continuous slow evolution of an object may also be disrupted by sudden drastic changes, some of which may be permanent, others temporary, and yet more others may even be part of a cycle. To model such evolution, a modeller may choose a discrete representation or alternatively they may use step semantics or some kind of interpolation to represent the unrecorded states. Similarly, extrapolation can be utilised to predict the short-term future.

Space-time topology

This criterion sets a standard concerning whether models can estimate metrics such as 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 (1983) temporal relationships.

Object identities

Another issue that can be employed to evaluate the modelling 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 that we should no longer consider it to be the same object? Sometimes it may be more appropriate to destroy the original instance of an object and recreate a new one, because of extensive changes. Another critical issue is that of splitting or unifying objects.

Dimensionality

With this criterion, models are examined as to whether they support two dimensions to model the spatio-temporal objects, as traditional GISs do. Although ''2.5''-dimensional solutions exist (perspectives, stereo views, etc.), volumetric three-dimensional GISs 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 categorisation is a superset of a similar benchmarking framework presented in (Theodoridis, 2003).

Queries about locations, spatial properties and spatial relationships

Queries of this category involve stationary reference objects. Examples include attributes of entities independent of space and time (e.g. who is the owner of this parcel), as well as point (e.g. where is

https://doi.org/10.1017/S026988890400013X Published online by Cambridge University Press

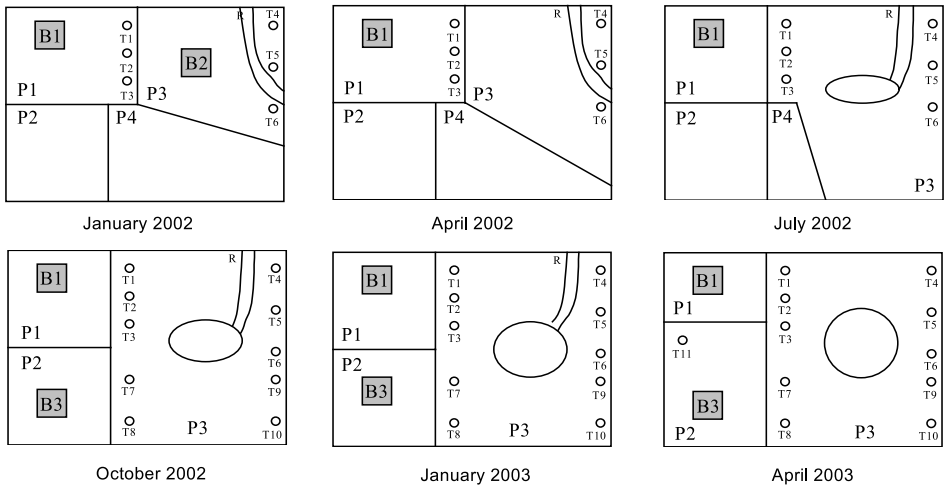


Figure 4 The LIS

this building), range/distance-based (e.g. find gas stations in this rectangular area/in this circle), nearest-neighbour (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 behaviours and relationships

This set of queries is further classified into three sub-categories. (a) *Simple* spatio-temporal queries on discretely changing objects (e.g. what is the state of a parcel at time *t*) or moving reference objects; examples include distance-based (e.g. find humans who 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 *behaviour* queries involving unary operators, such as travelled 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 (Koubarakis *et al.*, 2003), the above queries constitute the minimum functionality a spatio-temporal system should provide and we expect that future releases of commercial database management systems (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. GISs, 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.

https://doi.org/10.1017/S026988890400013X Published online by Cambridge University Press

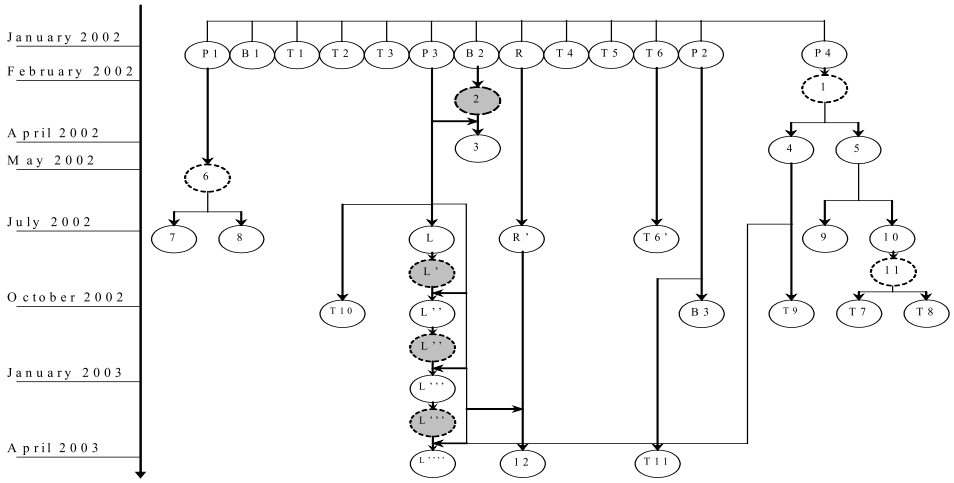


Figure 5 Event diagram for the LIS

- *January 2002*. A land parcel owned by Bob (P1), a land parcel owned by John (P2), a land parcel owned by the council (P3), a parcel owned by Bill (P4), buildings B1 and B2, river R and trees T1–T6.
- *February 2002*. The council makes a proposal to Bill to buy his land to build a park in the future.
- *April 2002*. The council decides to remove building B2 from its parcel and expands the parcel incorporating part of the parcel P4.
- *May 2002*. Bob decides to sell part of his land to the council. The council develops a project to build a park.
- *July 2002*. The council expands its parcel by incorporating part of 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 changes the position of tree T6. John starts the construction of B3.
- *October 2002*. The 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*. The lake continues to grow.
- *April 2003*. The council stops the flow of river R. Parcel P3 becomes a park that includes several trees T1–T10 and lake L. 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 the council and the actual sale happens in July 2002 (valid time). From the information provided, many of the valid times are not exact. For example, it is not clear at 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 a planned project for a park. The actual park can be undeveloped 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 indices used in the event diagram are explained in Table 1. Changes in the identity of spatio-temporal objects such as described by operations such as *create, destroy*, *suspend* and *resume*, in the case of the corresponding creation, deletion, temporal freezing and restoration of an object. *Fission* shows the emerging of new successor objects in the same position as a destroyed object. Consequently, *fusion* shows the creation of a single object on the position of several destroyed objects.

https://doi.org/10.1017/S026988890400013X Published online by Cambridge University Press

Table 1 Event indexes for LIS

Index	Event	Operation	Time
1	The 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 part of P3	Destroy	VT
4	The council gets part of P4	Create	VT
5	Part of P4 still belongs to Bill	Create	VT
6	As a result of Bob's decision to sell 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	The council gets part of P1	Create (fission)	VT
9	The council gets part of P4	Create (fission)	VT
10	P4 is decreased in size, property of Bill	Create (fission)	VT
11	The last part of P4 becomes a council's estate	Suspend	TT
12	The river is stopped and its bank becomes part of P3	Destroy, resume	VT
P1–P4	Parcel 1–Parcel 4	Create	VT
T1–T11	Tree 1–Tree 11	Create	VT
B1–B3	Building 1–Building 3	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

3 Spatio-temporal data models

Throughout the relatively young history of research on spatio-temporal modelling, a substantial number of models have been presented. This section examines most of the spatio-temporal models proposed in the literature over 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 (Langran, 1988). Temporal information has been incorporated into this spatial data model by timestamping 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 between the 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 of answering simple spatial, temporal and spatio-temporal queries, but it is difficult for the model to resolve all 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 has three major disadvantages (Langran, 1992).

- The model is not appropriate for describing changes in space through time. Each snapshot describes what exists at T_i. However, to detect how T_i differs from T_j, the two snapshots must be compared exhaustively.

https://doi.org/10.1017/S026988890400013X Published online by Cambridge University Press

- Regardless of 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 an understanding of the constraints on temporal structure.

Another closely related approach was presented by Armenakis (1992), where spatio-temporal data are looked at with respect to storage, retrieval and update efficiency. He compared three approaches, which he called ''*estimation methods*'' to describe time-varying spatial information. The aim of his investigation was to see whether 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 states.

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 indices 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 the unchanged components instead. In this scenario, when an object changes, its previous version is superseded but fully retained as 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 was suggested by Langran (1988). It is based on the principle that every line in space and time is projected down to the spatial plane and intersected with every other creating a polygon mesh. Each polygon in this mesh has its own attribute history associated with it. Each new amendment is intersected with existing lines, and new polygons are formed with individual histories. The model has been tested with a number of existing methods (Langran, 1992). 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 note 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.

The time dimension of the STC data model is based on a 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 a vector structure of space and the basic data type is the polygon. A STC 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 STC splits an object into two, the old object is effectively replaced by two new objects, each with a new

https://doi.org/10.1017/S026988890400013X Published online by Cambridge University Press

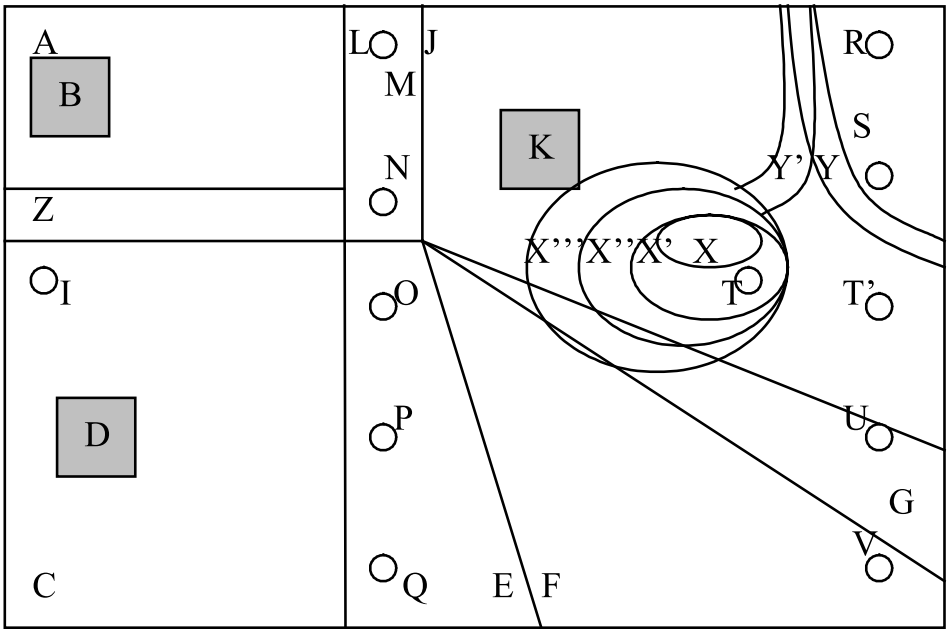


Figure 6 STC model for the LIS

identifier. 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 timestamping

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*'' (Renolen, 1996). In Hunt & Williamson (1990) 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 (Ramachandran *et al.*, 1994) where the *temporal change object* is introduced, which is an object consisting of a set of references to the past (historic), the future (scheduled) and the current version. The time cross section of this model for the LIS is described in Figure 7.

The strength of the simple timestamping 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 such as ''What was the state of . . .'' and weak on queries such as ''What happened . . .''.

https://doi.org/10.1017/S026988890400013X Published online by Cambridge University Press

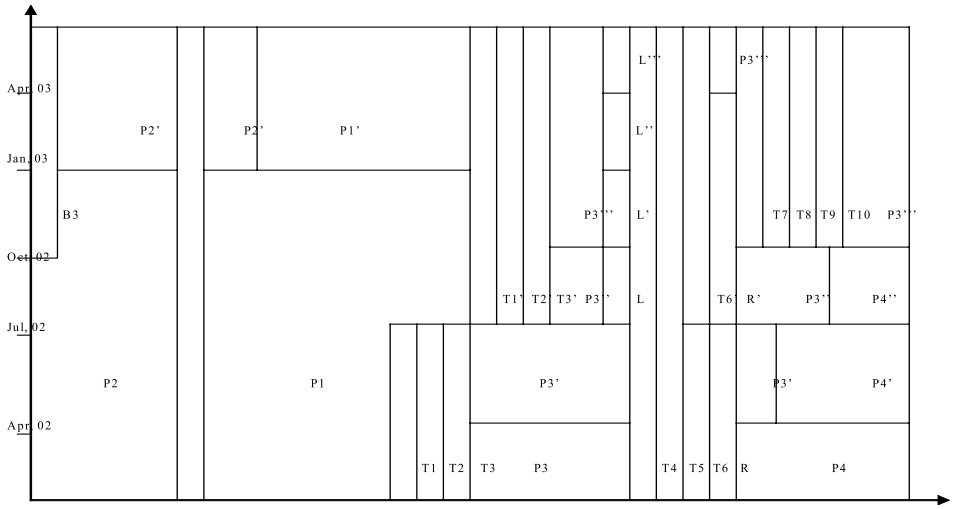


Figure 7 Simple timestamping approach for the LIS

3.4 Event-oriented models

The previous models cannot identify individual changes to, or events that affect, 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 spatial–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 Peuquet & Wentz (1994) such an approach is proposed 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 & Duan (1995) have implemented a raster-based event-oriented approach, called the event oriented spatio-temporal data model (ESTDM). The ESTDM groups timestamped layers to show observations of a single event in a temporal sequence and 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 timestamped 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 that it is capable of supporting both spatial and temporal queries, and does so efficiently. However, the transformation of the

https://doi.org/10.1017/S026988890400013X Published online by Cambridge University Press

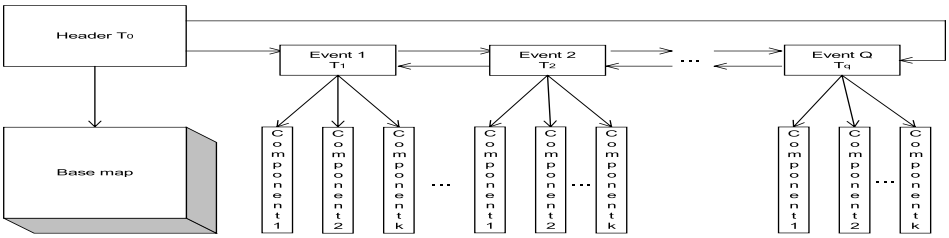


Figure 8 Primary element of an event-oriented model

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.

Langran (1992) described 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. 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 modelling.

3.5 The three-domain model

Yuan (1994, 1996b) described a three-domain model for spatio-temporal modelling. 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 other possible 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.

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

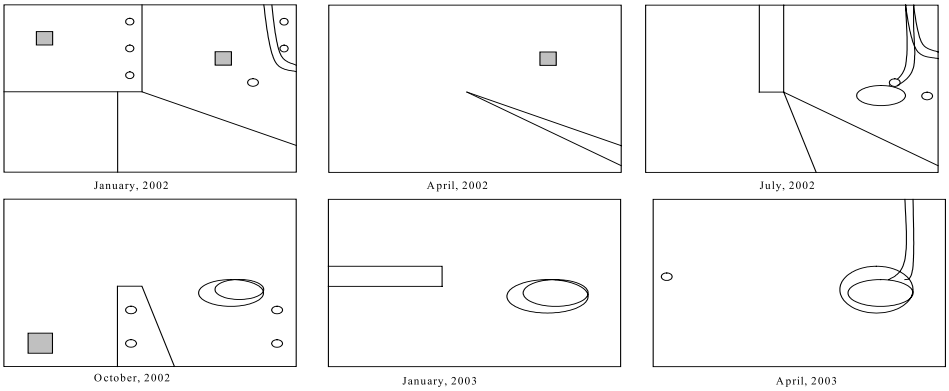


Figure 9 Amendment vector model for LIS

https://doi.org/10.1017/S026988890400013X Published online by Cambridge University Press

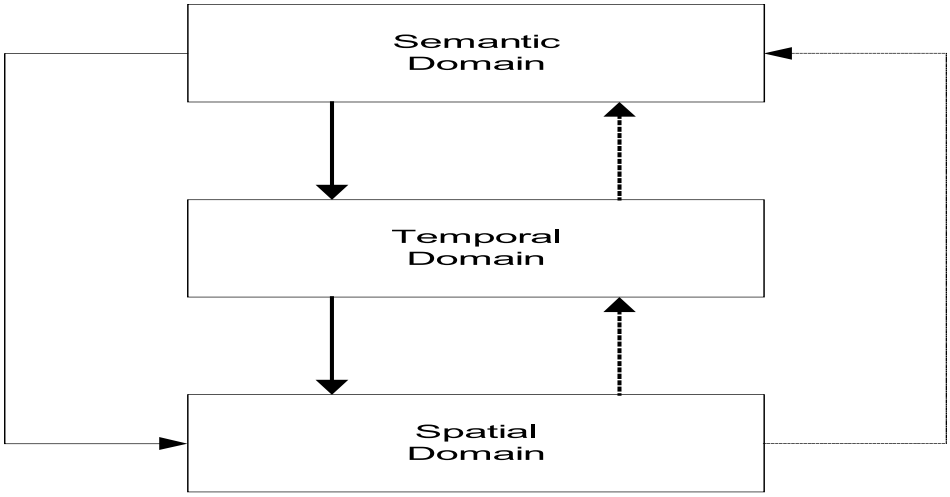


Figure 10 A conceptual framework for the three-domain model

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 (1997) showed that her model can support simple and range temporal and spatio-temporal queries.

In an alternative three-domain method presented in Claramunt & Theriault (1995), 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 the 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 significantly reduces data handling.

This model focuses on spatio-temporal events. Characteristics are categorised 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 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. A versioning

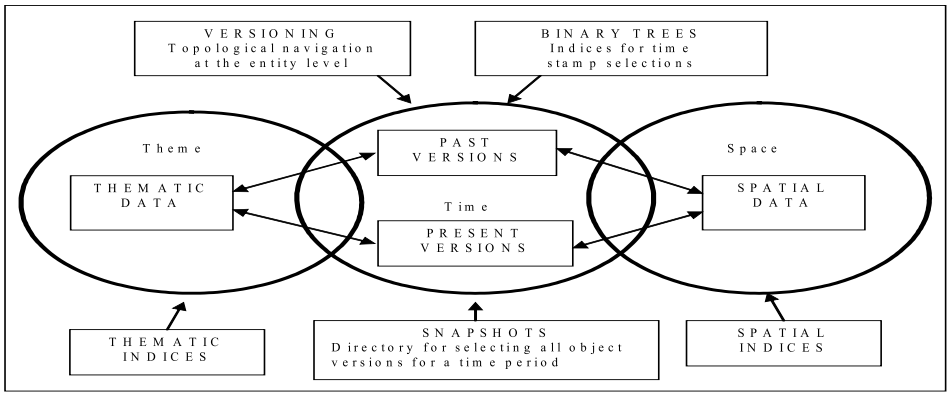


Figure 11 An alternative three-domain model

https://doi.org/10.1017/S026988890400013X Published online by Cambridge University Press

Table 2 Versioning table coming from applying three-domain model into 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		February 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 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 relevant versioning table for the LIS.

The model was a revolution in the development of spatio-temporal databases, 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 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, operators for dealing with the relationships between spatial objects are not defined, and it needs mechanisms to calculate changes.

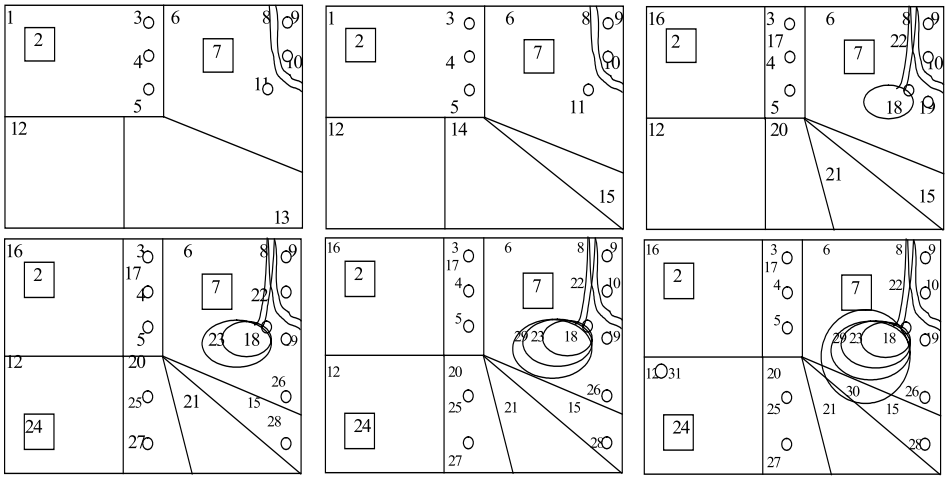


Figure 12 Three-domain spatial transitions for LIS

https://doi.org/10.1017/S026988890400013X Published online by Cambridge University Press

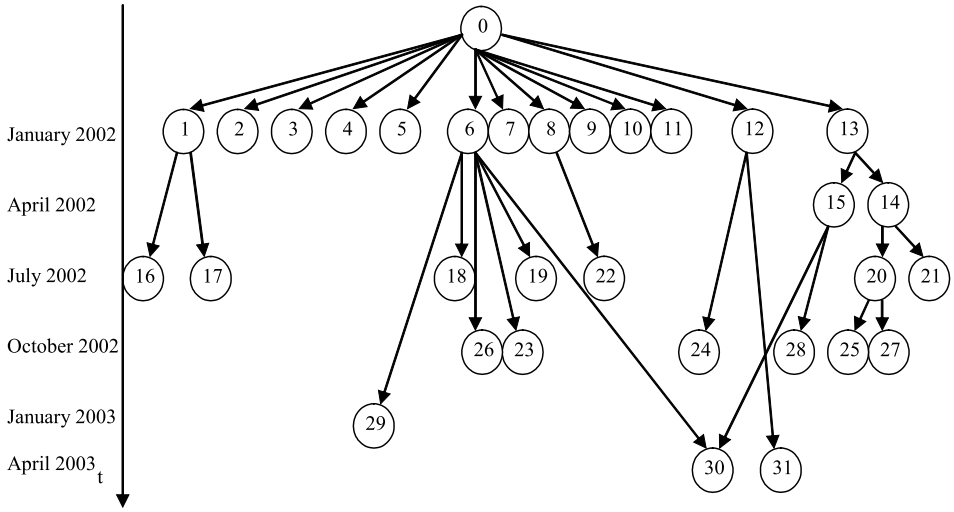


Figure 13 Three-domain spatio-temporal tree for LIS

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 only change by sudden events.

The main purpose of the history graph model (Renolen, 1996) is to identify all types of temporal behaviour and to manage both objects and events. The intention of the history graph notation is to visualise 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 the 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 characterised by two timestamps 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 transition 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, listed below.

- *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 no longer exists in the real world.
- *Reincarnation*: an object that was previously 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.

https://doi.org/10.1017/S026988890400013X Published online by Cambridge University Press

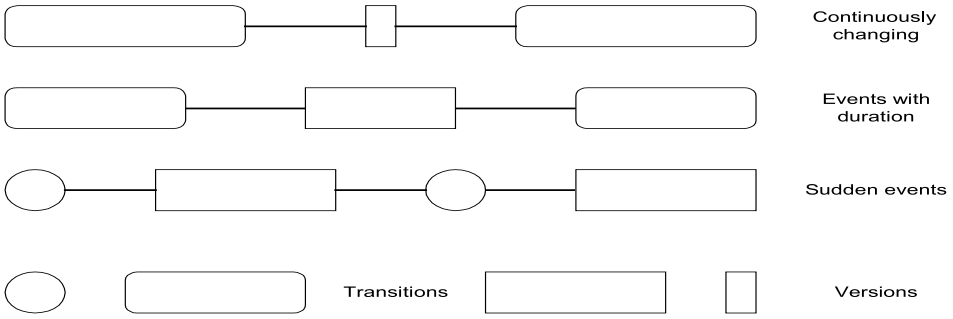


Figure 14 The behaviour types of temporal objects

- *Merge/annexation*: two or more objects are joined together to form a new object, or one or more objects are ''swallowed'' by another object.

Although the event-based models and the data models with simple timestamping 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.

As a whole, history graphs can be thought of 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 (1996) has shown 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 visualised in a conceptual model. Moreover, an implementation based on both processes and states

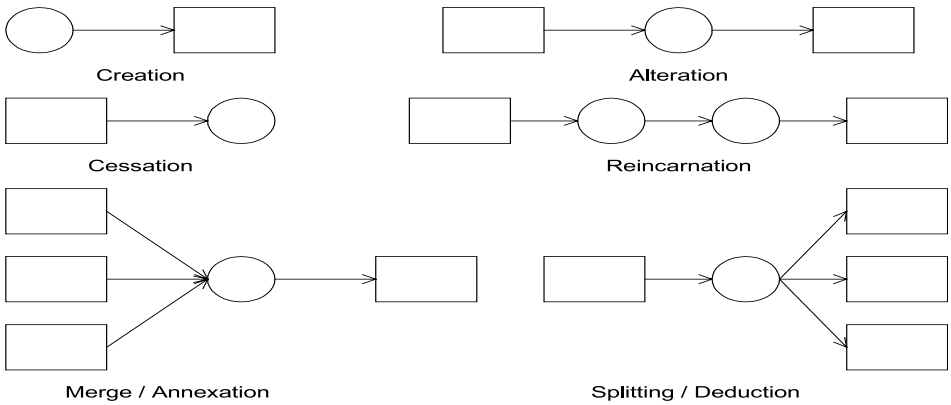


Figure 15 Six basic types of changes in the history graph model

https://doi.org/10.1017/S026988890400013X Published online by Cambridge University Press

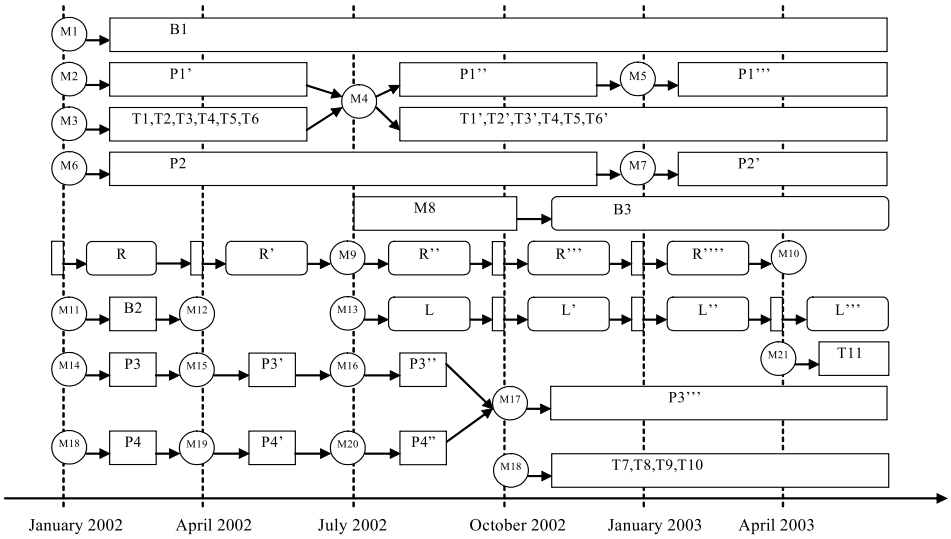


Figure 16 The history graph for the LIS

should be well suited for most types of query, as it will not have the deficiencies of the event-oriented or data-oriented models.

3.7 The spatio-temporal entity-relationship (STER) model

The entity-relationship model is the earliest and best-known conceptual database model, and it has recently been extended to capture spatio-temporal information. For some years the entity-relationship model was used to represent temporally referenced business data (Theodoulidis & Loucopoulos, 1991) or extended in order to deal solely with spatially referenced data (Hadzilacos & Tryfona, 1997). However, none of the proposed models satisfied the demanding requirements of a spatio-temporal information system. Later, Tryfona (1997) proposed an extension of the entity relationship model in order to model the phenomena of the real world and to capture them in 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 the 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 STER model (Tryfona & Jensen 1999; 2000). The careful analysis of spatio-temporal applications and the behaviour of spatial and temporal entities suggested that entity sets with their attributes and

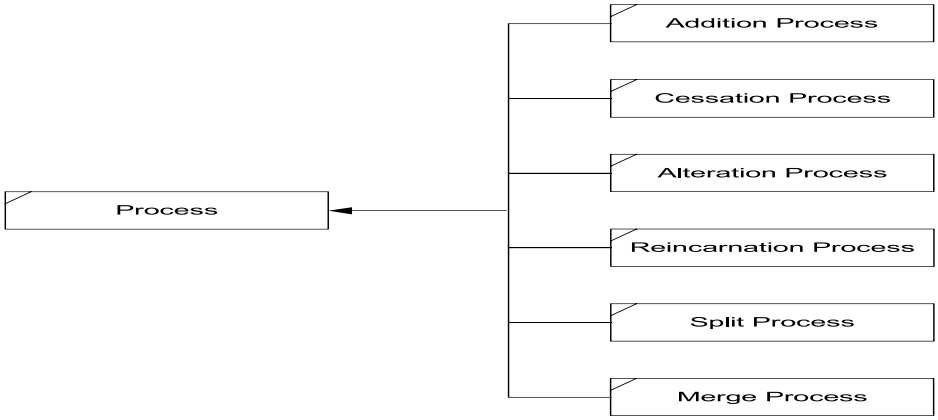


Figure 17 Object-oriented design of history graphs using inheritance

https://doi.org/10.1017/S026988890400013X Published online by Cambridge University Press

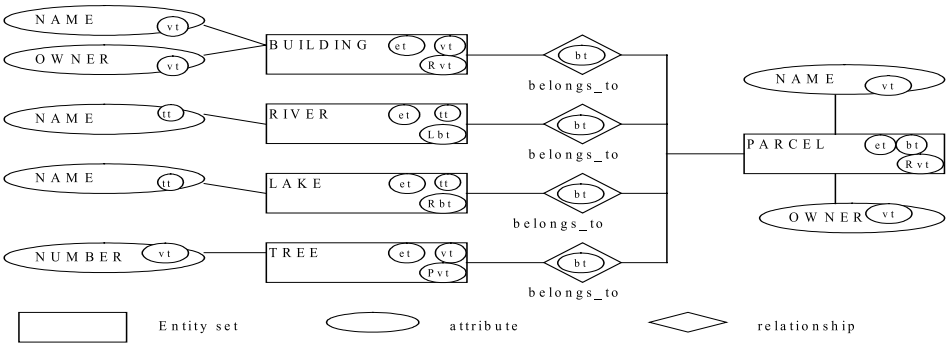


Figure 18 An excerpt of the LIS designed in STER

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 categorised 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. Furthermore, the ontological foundations for designing spatio-temporal databases we 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 we 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 attributes such as ''*ownership*'' and relationships among entity sets such as ''*reincarnation*'', ''*splitting*'' and ''*existence time*'' to be described. 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 in the following syntax of 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

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 whether a spatial object is dynamic or static.

3.8 Object-relationship (OR) 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, Claramunt *et al.* (1998) modelled an application that could capture the natural phenomena of environmental changes. They analysed already proposed models and came to the conclusion

https://doi.org/10.1017/S026988890400013X Published online by Cambridge University Press

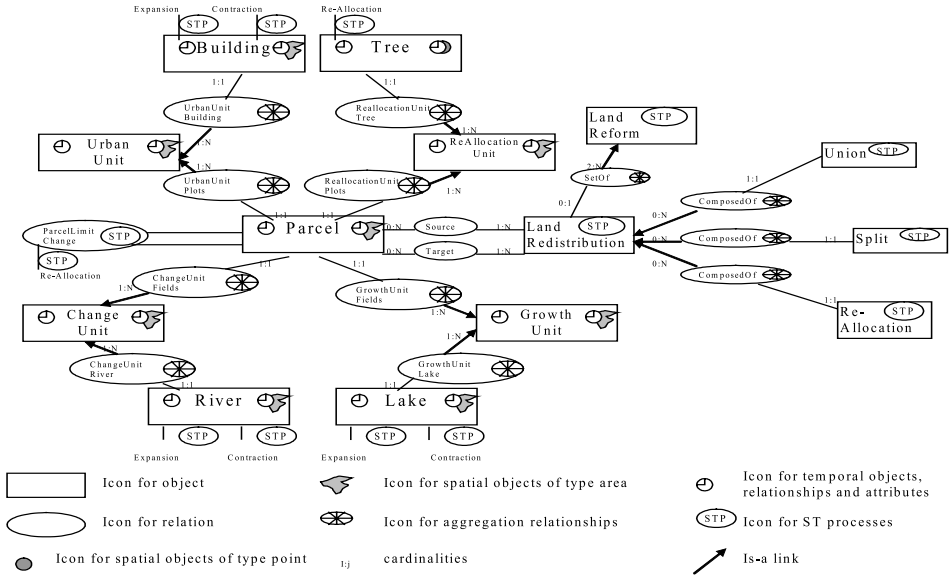


Figure 19 OR model for LIS

that existing models are more concerned with design than the representation of the natural environmental changes, processes and events.

In their introduction, Claramunt *et al.* (1998) reviewed several works on spatio-temporal database modelling. 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 (OR) 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 entity-relationship (Hadzilacos & Tryvona, 1997) and OO techniques. A fourth approach called MADS (modelling application data with spatio-temporal features) (Parent *et al.*, 1999) was also presented, which is implemented following the OR approach. The implementation of OR 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.

MADS incorporates space and time modelling basics into the OR model (Parent *et al.*, 1999). 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, different types of processes were classified and represented as relationships between spatio-temporal objects. At the same time, the properties of objects, 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 best describes 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 visualised through icons in the schema. The proposed design using the LIS case study is illustrated in Figure 19 together with the following 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

https://doi.org/10.1017/S026988890400013X Published online by Cambridge University Press

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 a need to represent processes and changes on the conceptual level, there are no extensive definitions and operations given. The modelling 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.

3.9 Spatio-temporal OO data models

One of the fundamental challenges facing the software industry is dealing with change in databases. OO 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 OO 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, into spatio-temporal database development. As identified by Montgomery (1995), there are four advantages of using the OO approach in spatio-temporal database modelling: (i) a single object can represent the whole history of an entity; (ii) queries are simple, because they deal with each single object of an entity; (iii) temporal data handling is efficient; and (iv) uniform treatment of spatial and temporal data handling is possible.

Worboys *et al* (1990) were the first to merge object orientation into the spatio-temporal modelling. They introduced the concept of the spatio-temporal object and defined 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 (Worboys, 1994a). A finite set of such spatio-temporal simplexes are then defined to form a spatio-temporal complex on which a query algebra is developed. Furthermore, operations such as union, intersection, difference, equality, subset, boundary, spatial projection and temporal projection are provided on this spatio-temporal complex.

The same year, Wachowicz & Healey (1994) introduced an OO 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 just 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: (i) every object must have an initial version; (ii) a hierarchical structure is imposed on the versions of an object; (iii) different versions of an object denote different object instances; (iv) among versions, a current version is always distinguished.

At the version-configuration level, two main OO mechanisms, namely generalisation 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*.

https://doi.org/10.1017/S026988890400013X Published online by Cambridge University Press

Table 3 Update procedures and their respective spatio-temporal changes

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

In another attempt, Bonfatti & Monari (1994) described an integrated approach to model both geographical structures and phenomena. They argued that cross-references between objects to express relationships are ambiguous; hence, better means are needed to characterise object structure and behaviour. Their proposed solution is the use of complex objects, comprising 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 modelling of applications that are space-dominant, such as 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*{	OBJECT *StraightLine*{	OBJECT *Segment*{	OJECT *Line*{
X, Y: *real*;	A, B, C: *real*;	Length: *real*;	Length: *real*;
}	}	Start, End: *Point*;	Start, End: *Point*;
		LyingLine: *StraightLine*;	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*.

- Law 1: $LyingLine.A*Start.X+LyingLine.B*Start.Y+LyingLine.C=0$.
- Law 2: $LyingLine.A*End.X+LyingLine.B*End.Y+LyingLine.C=0$.
- Let $L=Dist(Start, End)$, then Law 3: $Length=L$.

As we mentioned before, the formulation of laws that concern both space and time is not possible for the LIS. Applications listed as spatio-temporal dominant (simulation modelling, electronic charting, environmental management, etc.), would be better 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 OO data model for spatio-temporal data has been proposed by Rojas-Vega & Kemp (1995). They described a structure for distributed multimedia spatial applications and the structure and interface definition language (SIDL) was developed for this purpose. To achieve the full

https://doi.org/10.1017/S026988890400013X Published online by Cambridge University Press

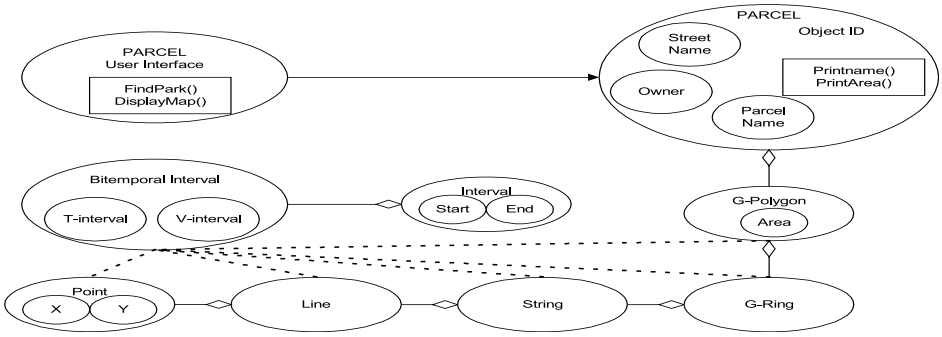


Figure 20 Modelling a Parcel of LIS with an OO model

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 whether 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.

In Becker *et al.* (1996) and Faria *et al.* (1998) the authors proposed OO models based on extensions of ObjectStore and O2, respectively. Becker *et al.* (1996) defined a hierarchy of elementary spatial classes with both geometric and parameterised thematic attributes. Temporal properties are incorporated by adding instant and interval timestamp key words to the query language. In Faria *et al.* (1998), 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 Becker *et al.* (1996) and Faria *et al.* (1998) offer text-based query languages; the non-graphical query languages of these models reduce their suitability as conceptual modelling languages.

Other OO models were proposed based on the object model supported by the Object Modelling Technique (OMT) (Rumbaugh, 1991). The main advantage of this approach is its high-level representation abstraction, which deals with the concepts of modelling rather than with implementation. An interesting approach based on the object model was introduced by Hamre (1994), where a model for representing data varying in three-dimensional space and time (fourth dimension) 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 storing the time when the data is collected (valid time). The values in a data set can be spread either regularly or irregularly within the four-dimensional space by x, y, z and t. For simplicity, it

https://doi.org/10.1017/S026988890400013X Published online by Cambridge University Press

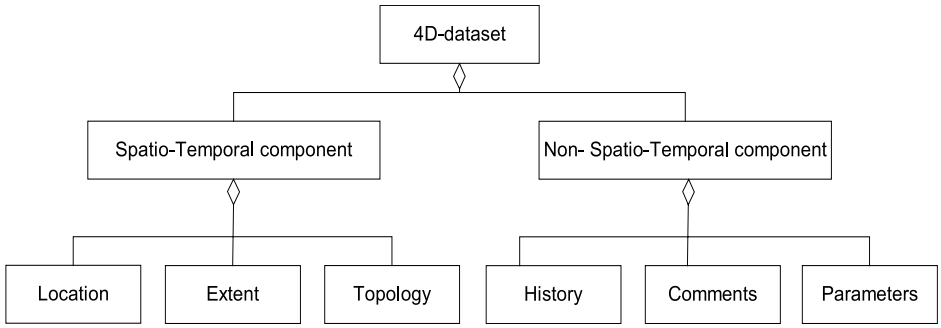


Figure 21 Overview of the components of a four-dimensional dataset

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.

What is more, in this model four-dimensional 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 zero to four 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, RegularLatitude-Series, 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.

Concluding the description of this spatio-temporal OO data model, we must state that operations for creation, updating and deletion of objects are allowed, as well as the ability to specify constraints and relationships between different types of data sets. Finally, in the author's implementation proposal it 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 by Tryfona *et al.* (1998) to develop the geographic object model (Geo-OM). A geographic object's position, space varying dynamic attributes, spatial relationships and operations on objects were 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

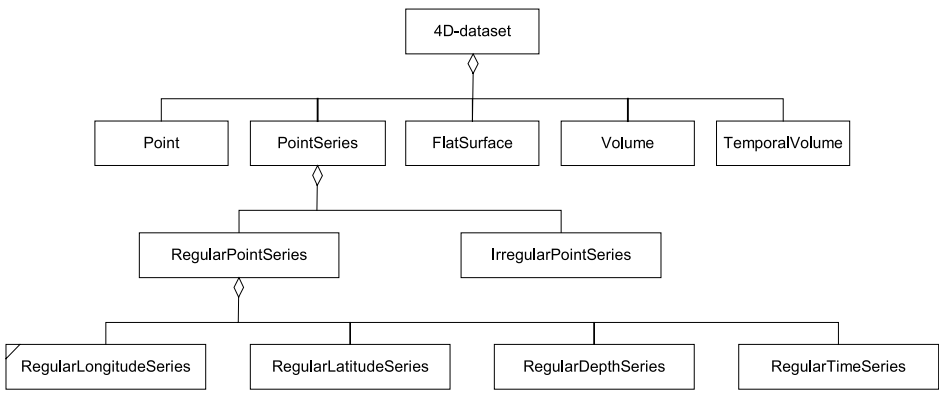


Figure 22 Main classes of four-dimensional datasets

https://doi.org/10.1017/S026988890400013X Published online by Cambridge University Press

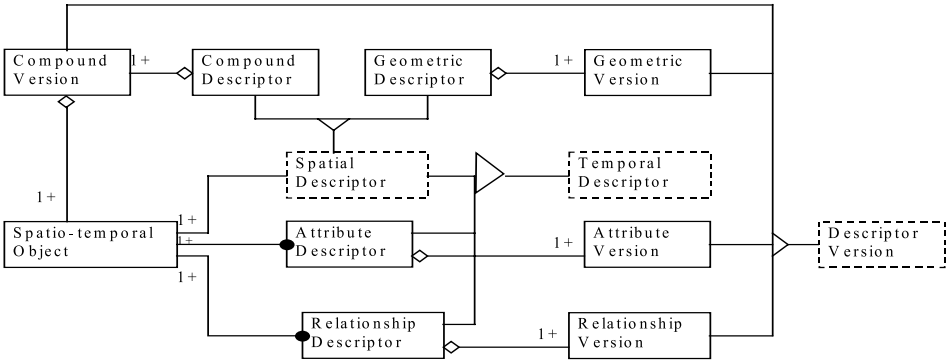


Figure 23 STOM

Geo-OM development showed that the OO approach could be readily applied to capture both spatial static and dynamic properties, which can then also be referred to the temporal semantics.

Renolen (1997b) used the OMT object modelling 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 Renolen (1996). 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.

Each geometric descriptor comprises 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 (Figure 24) to illustrate the spatio-temporal object semantics of the model.

The STOM allows the characteristics and the properties of spatio-temporal objects to be captured 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 OO approach, the notation is not concise, clear and consistent enough to meet the demands of dynamic spatio-temporal modelling.

Worboys (1994a) investigated the possibility of using the OO approach in modelling dynamic spatial systems in order to analyse the fundamental constructs of ''change'', ''event'' and ''process'' in an attempt to develop a foundational model for spatio-temporal databases. In Worboys (1998), comparisons were made between events, change and processes associated with the OO modelling constructs, highlighting the necessity for additional concepts in order to handle the dynamic spatial applications. Raza & Kainz (1999) propose a conceptual data model as an extension to Worboy's model (1994a), to conceptually separate out space, time and attributes, and consolidate them in a modular fashion as distinct elements.

More recently, in Hornsby & Egenhofer (2000), the authors describe a visual language for the explicit description of change relating to objects, called the change description language. This work focused on the appearance/disappearance of entities and particularly of the possible identity transitions from one object to another. 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



Figure 24 Extensions of STOM schema

https://doi.org/10.1017/S026988890400013X Published online by Cambridge University Press

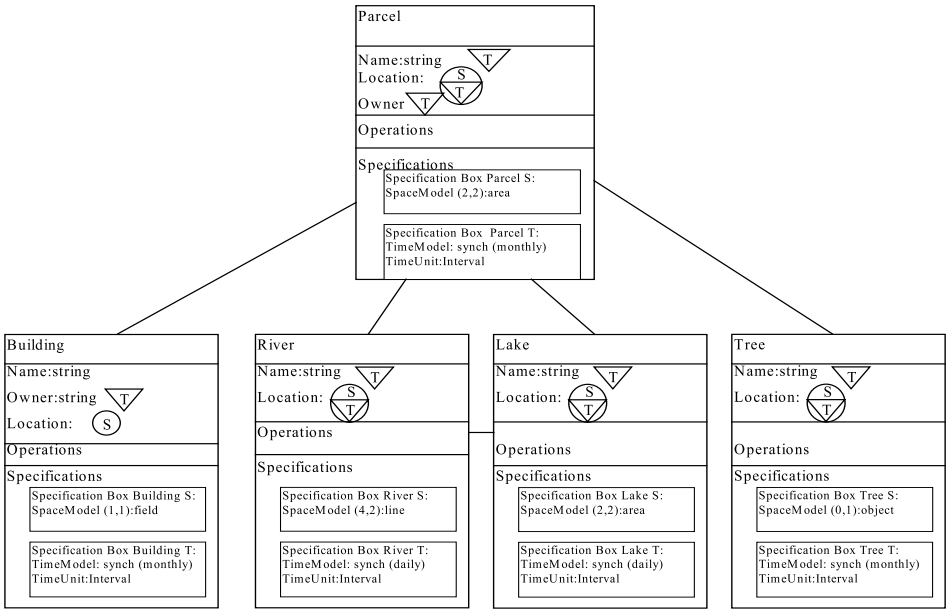


Figure 25 The LIS modelled with STUML

Bonhomme (1999) the Lvis language, originally designed for visual queries of spatial data (Aufaure-Portier & Bonhomme, 1999), 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 Modelling Language (UML) (Fowler & Scott 1998) is another OO standard that is recognised by the Object Management Group (OMG) (OMG, 1995). The UML was extended to give the spatio-temporal UML (STUML) (Price *et al.*, 2000) in an attempt to develop a high-standard spatio-temporal database language, which would be capable of supporting a range of spatio-temporal models and data types using a clear, simple and consistent notation. An extension of the OMG standard for OO modelling 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 by Price *et al.* (1999). The authors defined constructs and corresponding graphical notations that provided support for modelling spatio-temporal properties. However, neither the syntax nor the semantics of the symbols introduced were presented formally.

The extended STUML in Price *et al.* (2000) maintains the clarity and simplicity of the language by introducing a small base set of fundamental modelling constructs: spatial, temporal and thematic. These constructs can then be combined and applied at attribute, attribute group, association and/or class levels of the OO 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 modelling 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.

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 can also change continuously, and in the latter case we are talking about moving objects. If only the position in space of an object is relevant, then a moving point is a basic abstraction; if the extent is also of interest, then a moving region abstraction captures moving as well as growing or shrinking regions. Researchers have tried to model spatio-temporal databases using this concept of a moving object. More specifically, Erwig

https://doi.org/10.1017/S026988890400013X Published online by Cambridge University Press

et al. (1999) proposed a new line of research where moving points and moving regions are viewed as three-dimensional (two dimensions plus time) or higher-dimensional entities whose structure and behaviour are captured by modelling them as abstract data types. Such abstract data types for moving points and moving regions were introduced by Guting *et al*. (2000), together with a set of operations on such entities. The authors argued 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 an 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×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 with 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 work, are dealt with here.

The model presented by Guting *et al*. (2000) lacks design and implementation. As such, in Forlizzi *et al*. (2000), the definition of the discrete representation of the above-discussed abstract data types is presented. 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.

The sliced representation is built by a type constructor ''*mapping*'' parameterised by the type describing a single slice that 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, the unit types *ureal*, *upoint*, *upoints*, *uline* and *uregion* are defined. 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.

https://doi.org/10.1017/S026988890400013X Published online by Cambridge University Press

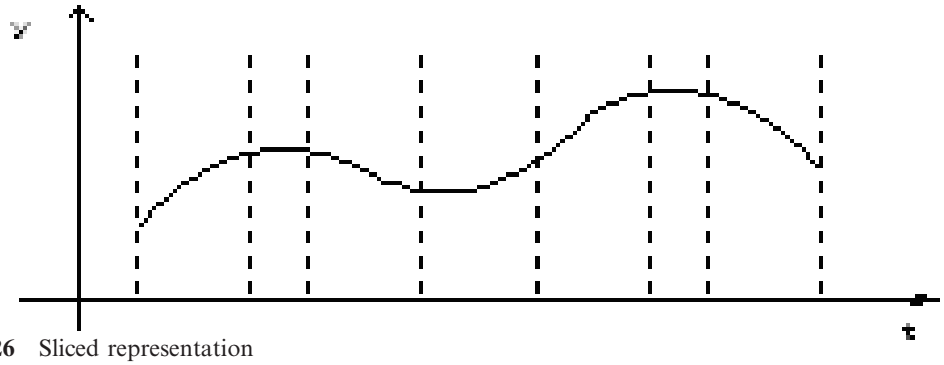


Figure 26 Sliced representation

The next step in this development was the study of algorithms for the rather large set of operations defined in Guting *et al.* (2000). Whereas Forlizzi *et al.* (2000) just provides a brief look at this issue by presenting two example algorithms at the end, in Lema *et al.* (2003) the authors present a comprehensive, systematic study of algorithms for a subset of the operations introduced in Guting *et al.* (2000). Whereas some algorithms are relatively straightforward and simple, there is still a considerable number of quite involved algorithms, whose soundness and ability to be implemented it is hard for the reader to assess, since they are only provided in terms of pseudo-code. In all cases the authors analyse the complexity of the algorithms. In Lema *et al.* (2003) the data structures from Forlizzi *et al.* (2000) 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 (Dieker & Guting, 2000) and as an Informix Datablade.

As an extension to the abstract model in Guting *et al.* (2000), Erwig & Schneider (1999b) 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 Erwig & Schneider (1999a). Further work on modelling includes Su *et al.* (2001), 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 Vazirgiannis & Wolfson (2001) the authors consider movement in networks and some evaluation strategies.

An interesting approach following the paradigm of moving objects was also presented in the STAU project (Pelekis, 2002). This research was focused on the representation and querying of continuously as well as discretely moving objects similar to those presented in Guting *et al.* (2000). From a theoretical point of view the author proposed a datatype-oriented model that supports the representation of objects both under OO and OR 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 (Kakoudakis & Theodoulidis, 1996a). 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 (truck information system (TIS)).

Another model using moving objects was proposed by Wolfson and co-workers (Sistla *et al.*, 1997; Wolfson *et al.*, 1998, 1999). The authors propose a data model called moving objects spatio-temporal (MOST) data model for databases with *dynamic attributes*, i.e. attributes that change continuously as a function of time, without being explicitly updated. This model enables the

https://doi.org/10.1017/S026988890400013X Published online by Cambridge University Press

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 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 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 has identified common directions of modelling in the area and this becomes clearer if there is a certain comparison framework. As such, the following 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 four categories of criteria and requirements that evaluate spatio-temporal modelling techniques 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 incorporate them into existing models. Most of the approaches assume discrete modelling of time, while the rest are continuous models, which include discrete temporal density as a subcase. However, both levels of modelling 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 (Theodoridis, 2003) will be missed. This is because the representational problems might lead us to prematurely discard some options for modelling.

Moreover, in the majority of the models, time is incorporated by timestamping 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 an integral part of the spatial entities in terms of simple continuous functions, is the best method and one that should be adopted 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 for modelling spatio-temporal data should not only include bitemporal time but user-defined time as well. All the models assume that the time axis is linear, and so their behaviour is inadequate in some cases where the nonlinearity 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

https://doi.org/10.1017/S026988890400013X Published online by Cambridge University Press

https://doi.org/10.1017/S026988890400013X Published online by Cambridge University Press

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

Spatio-temporal data models	Temporal granularity	Temporal operations	Temporal density	Representation of time	Transaction (TT) versus valid (VT) versus bitemporal time (BT)	Time order	Lifespan
Snapshot	Multiple	No	Discrete	Modelling: layers–snapshots Time as: attribute of location and absolute reference	VT	Linear	No
STC	Multiple	No	Discrete	Modelling: polygon history Time as: integral part of spatial entities and relative reference	BT	Linear	No
Simple timestamping	Multiple	Yes	Discrete	Modelling: object's creation–cessation Time as: attribute of the object and absolute reference	VT	Linear	Yes
Event-oriented	Multiple	No	Discrete	Modelling: events, change Time as: attribute of an event and relative reference	VT	Linear	Yes
Three-domain	Multiple	Yes	Discrete	Modelling: temporal versions Time as: independent object and absolute/relative reference	BT	Linear	No
History graph	Multiple	No	Discrete/continuous	Modelling: events, processes Time as: attribute of objects, events and absolute/relative reference	BT	Linear	Yes
STER	Multiple	No	Discrete	Modelling: entity change Time as: attribute of entity, relationship and absolute reference	BT	Linear	Yes
OR	Multiple	No	Discrete/continuous	Modelling: ST phenomena Time as: attribute of object and absolute/relative reference	BT	Linear	Yes
OO	Multiple	Yes	Discrete/continuous	Modelling: object change Time as: attribute of object and absolute/relative reference	BT	Linear	Yes
STUML	Multiple	No	Discrete/continuous	Modelling: TimeUnit Time as: via the specification box and absolute reference	BT	Linear	Yes
Moving objects	Single	Yes	Discrete/continuous	Modelling: functions Time as: integral part of spatial entities and absolute reference	VT	Linear	Yes

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

Spatio-temporal data models	Structure of space	Orientation/direction	Measurement	Topographical relationships
Snapshot	Raster	No	No	No
STC	Vector	No	No	No
Simple timestamping	Vector	No	No	No
Event-oriented	Vector/raster	No	No	Yes
Three-domain	Vector	No	No	Yes
History graph	Vector	No	No	No
STER	Vector	No	No	No
OR	Vector	No	No	Yes
OO	Vector/raster	Yes	Yes	Yes
STUML	Vector	No	No	No
Moving objects	Vector/Raster	Yes	Yes	Yes

approaches. Finally, except in the earlier introduced models, there is sufficient behaviour in handling the duration of the events and the history of the spatio-temporal objects (lifespan).

4.2 In terms of spatial semantics

Before stating specific conclusions arising from Table 5, we should first point out that new proposals in the area of spatio-temporal database modelling should be based on spatial data models that present sufficient behaviour in all different spatial requirements. The majority of the models assume a vector structure of space while the raster structure is adopted by four models. Even though there are applications that need the raster structure of space (e.g. multimedia), it is otherwise considered obsolete. Vector representation is much simpler and generic and, what is more, it can easily be transformed to raster structure. The orientation and direction of spatio-temporal objects during their change or movement across space and time dimensions are only discussed in the OO 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 the relative positions of objects. Early approaches such as the snapshot, the space-time composite and the simple timestamping models address none of these issues, while the same also stands for models that are more conceptually oriented such as the STER, the OR model and the STUML.

4.3 In terms of spatio-temporal semantics

The comparison in terms of spatio-temporal semantics is more general and comprises factors related to the model domain. The most commonly used data types are the point, line, region, temporal point and temporal interval, while there are approach-specific data types such as parametric rectangles and moving points. Most of the models capture some aspects of the notion of change as described in the requirements analysis section, but only few of them support all eight possible types of change (see Figure 3). Almost all of the models are capable of handling 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

https://doi.org/10.1017/S026988890400013X Published online by Cambridge University Press

non-interrelated components. Even though moving objects are unified objects, due to their nature there is no meaning in talking about identification of such objects. What is more, only OO 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 with respect to these issues. Finally, most of the models are capable of capturing different dimensionality representations of spatio-temporal objects, while the others can do that by slight changes and extensions.

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 STC and the simple timestamping 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 there is a need 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, such as the history graph, the three-domain and the OO approaches, which support a variety of spatio-temporal queries, but at the end, although they have great potential, cannot cover all possible physical processes and phenomena because of the limited, non-specialised operation set provided by these approaches. Currently, this feature is presented only by moving objects, which can resolve any type of spatio-temporal query.

5 Discussion and conclusions

In this section the synthetic framework presented above is augmented by a discussion declaring the implementation status of each data model. The reviewed models vary as far as completeness, formalisation and implementation are concerned. In addition, some models incorporate different spatial and temporal database models to achieve the final objective, and others have proposed their modelling approach from scratch. In the following we provide a concluding comparative table with the following features.

- *Formalisation*: this factor shows whether the model has been formally defined or not.
- *Implementation*: this criterion depicts whether 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.

As can be seen from Table 8, not all of the models have been formally defined so as to facilitate a spatio-temporal database designer and developer in their tasks. Furthermore, for some of the models (STER and STUML) it remains the case that, even though they have been formalised, their modelling power remains at the conceptual level, and as such the design concepts are not supported and associated with corresponding technical constructs. The problem as to 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

https://doi.org/10.1017/S026988890400013X Published online by Cambridge University Press

https://doi.org/10.1017/S026988890400013X Published online by Cambridge University Press

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

Spatio-temporal data models	Data types	Primitive notions	Type of change All eight types: A/N Discrete: D/N Continuous: C/N Movement: M/N	Evolution in time and 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 timestamping	Parcel-polygon, temporal point	Creation–cessation time of an object	N-D-N-N	No	No	No	2D
Event-oriented	Date, event and raster structures	Event, amendment maps	N-D-N-N	No	No	No	All
Three-domain	Point, line, region, time point, interval	Thematic, spatial and temporal objects	A-D-C-M	No	Yes	Yes	All
History graph	Point, line, region, time point, interval	Object's version and transition	N-D-C-M	No	No	Yes	All
STER	Point, line, region	Entity, relationship	A-D-C-N	No	No	No	2D
OR	Point, line, area, temporal point	Object, event, process, relationship	A-D-C-N	No	No	Yes	All
OO	Simple descriptors, point, line, area	Object, attribute, relationship	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

that the spatio-temporal research community has to focus on this area in the forthcoming years. Finally, the last two columns in Table 8 prove that the existence of a functional, flexible and extensible pure spatial and temporal data model is very significant, not only to be able to introduce an efficient spatio-temporal data model, but also to be able to utilise it separately in the corresponding domains. In other words, it is very important to be able to represent spatial entities and to have complete functionality on them independently of the time dimension and *vice versa*.

The following consider the spatio-temporal research agenda that is important in the context of computing science in depth, in terms of the requirements that determine the priorities of modelling spatio-temporal applications and highlight key issues from those of lesser significance. Undoubtedly, spatio-temporal database modelling requires a specific approach, for reasons that include the following.

- Spatio-temporal databases typically deal with large complex bodies of spatially referenced data, which are required to be readily available (Armenakis, 1992). 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 are either regularly/irregularly updated from external data sources or need to be continuously updated due to evolution of natural processes (Hazelton, 1992).
- 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 (Peuquet, 2001).
- The ontology of space-time deals with the nature of space, time and their unique interactions. Investigation at this highly abstract level is essential in order to develop a common conceptual framework (Tryfona & Jensen, 1999). Issues specific to time, linear versus cyclic views of time, multiple times (world, valid, user-defined time), continuous or discrete change, branching and alternative timelines, etc., seem to be recognised and well understood in a database context (Langran, 1992).
- The spatial and temporal dimensions should be considered separately and incorporated into the database design. Temporal and spatial characteristics of the modelling 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 modelling; however, spatial changes can lead to losing a thematic description of a particular dynamic object (Peuquet & Duan, 1995).
- 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 modelling in the information system.
- Complex functions and calculations involving operators, relationship status (Erwig & Schneider, 1991a) and objects for the prediction of their future motion need to be designed (Wolfson *et al.*, 1998). Research is also needed on applications of newer computational paradigms, such as constraint-based approaches (Grumbach *et al.*, 2001), fuzzy sets (Wang *et al.*, 1990) and rough sets (Ahlquist *et al.*, 2000).
- 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, visualisation methods and approaches to metadata handling (Bonhomme, 1999). Data mining and exploratory uses of information systems hold much promise (Han & Kamber, 2001). In the literature, there exist a number of algorithms and techniques supporting a series of types of knowledge extracted from traditional databases

https://doi.org/10.1017/S026988890400013X Published online by Cambridge University Press

https://doi.org/10.1017/S026988890400013X Published online by Cambridge University Press

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

Spatio-temporal data models	Spatial queries					Temporal queries			ST-queries		
	Attribute	Point	Range/distance-based	Nearest neighbour	Topological	Simple	Range	Relationships	Simple	Range	Behaviour
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 timestamping	Yes	Yes	No	No	No	Yes	Yes	No	Yes	No	No
Event-oriented	Yes	Yes	No	No	Yes	Yes	Yes	No	Yes	No	No
Three-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
OR	Yes	Yes	Yes	No	Yes	Yes	Yes	No	Yes	No	No
OO	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

https://doi.org/10.1017/S026988890400013X Published online by Cambridge University Press

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

Spatio-temporal data models	Formalisation	Implementation	Tool	Application	Spatial model	Temporal model
Snapshot	No	No	None	LIS	GIS	N/A
STC	No	No	None	LIS	GIS	N/A
Simple timestamping	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
Three-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
OR	Yes	Yes	MADS	Rural-urban land use application	MEOSIG	MODUL-R POLLEN
OO	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 and STAU Oracle data cartridge	Multimedia scenario, forest fire control management	Abstractions of spatial data types and Oracle spatial	Abstractions of temporal data types and TAU types

(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 realisation 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 (Renolen, 1997a). This solution needs to be based upon a uniform ontological framework and requires a multi-representational approach.
- The requirement of multidimensional modelling in conjunction with the large bodies of geographically referenced data poses a range of new challenges in the context of the data-warehousing techniques (Jensen *et al*., 2004). Firstly, due to the fact 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 up to now. This is needed to increase the modelling power of the model and to enable new kinds of queries. The above discussion raises specific issues as the normalisation of the hierarchies as well as the handling of imprecision that is introduced by partial containment in the aggregation paths.
- 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 (Parent *et al*., 1999).
- The means to a better ''fit'' is, in part, being provided by the OO (Tryfona & Jensen, 2000) and OR (Guting *et al*., 2000) approaches, which support flexible constructs such as Abstract Data Types (ADTs) and Moving Objects Databases (MODs). However, there remains a lack of both theoretical and technical frameworks that explicitly describe the fundamental spatio-temporal characteristics in a context-independent way from which to draw application and user-oriented views.
- It is not an exaggeration to say that the so-called MODs are (or soon will be) ubiquitous. As the range 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 techniques for identifying their location (e.g. Global Positioning System (GPS)) have emerged, the special category of Location Based Services (LBS) exhibit particular scientific interest.

Concluding, early work on spatio-temporal database modelling attempted to capture the state of real-world objects or the physical events upon them, on the timeline. However, investigations in real-world applications brought new directions and requirements for further development. Thus, the processes of continuous change on the timeline, 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. The 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.

https://doi.org/10.1017/S026988890400013X Published online by Cambridge University Press

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 modelling, multiple timeline representation, complex behavioural spatio-temporal queries and the manipulation of very large data sets are some of the open issues for further investigation and development. Although attempts were made to cover such problems, due to the failure in the selection of appropriate applications carrying all real-world processes, changes, entities and characteristics, models dealing with one issue are not capable of handling others. Due to the complex structure of spatio-temporal databases, the methods able to control intricate entities are the most promising in the field. The OO 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 provided, accentuating 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 modelling has been covered to identify the particular approaches and applications used in the development of each model and to envision the next steps in spatio-temporal database research.

References

Abraham, T and Roddick, JF, 1999, Survey of spatio-temporal databases. *GeoInformatica* **3,** 61–99.

Ahlqvist, O, Keukelaar, J and Oukbir, K, 2000, Rough classification and accuracy assessment. *International Journal of Geographical Information Science* **14,** 475–496.

Allen, JF, 1983, Maintaining knowledge about temporal intervals. *Communications of the ACM* **26**(11), 832–843.

AlTaha, KK, Snodgrass, RT and Soo, MD, 1994, Bibliography on spatio-temporal databases. *ACM SIGMOD Record* **22,** 59–67.

Armenakis, C, 1992, Estimation and organization of spatio-temporal data. *Proceedings of the Canadian Conference on GIS92, Ottawa, Canada, 1992*.

Aufaure-Portier, MA and Bonhomme, C, 1999, A high level visual language for spatial data management. *Proceedings of Visual '99, 3rd International Conference on Visual Information Systems, Amsterdam, 1999*.

Becker, L, Voigtmann, A and Hinrichs, K, 1996, Temporal support for geo-data in object-oriented databases. *Proceedings of Database and Expert Systems Applications*, pp. 79–93.

Bonfatti, F and Monari, PD, 1994, 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. Amsterdam: Elsevier.

Bonhomme, C, 1999, A visual language for querying spatio-temporal databases. *Proceedings of the 7th ACM Symposium on Advances in Geographic Information Systems*, Kansas City, MO.

Claramunt, C, Parent, C, Spaccapietra, S and Theriault, M, 1998, Database modeling for environmental and land use changes. *Geographical Information and Planning*. Berlin: Springer, ch. 20.

Claramunt, B and Theriault, M, 1995, Managing time in GIS: an event oriented approach. In Clifford. J and Atuzhilin. A. (eds.), *Recent Advances on Temporal Databases*. Berlin: Springer, pp. 23–42.

Dieker, S and Guting, RH, 2000, Plug and play with query algebras: Secondo. A generic DBMS development environment. *Proceedings of the International Database Engineering and Applications Symposium (IDEAS)*, pp. 380–390.

Erwig, M, Guting, RH, Schneider, M and Vazirgiannis, M, 1999, Spatio-temporal data types: an approach to modeling and querying moving objects in databases. *GeoInformatica* **3**(3), 265–291.

Erwig, M and Schneider, M, 1999a, Developments in spatio-temporal query languages. *Proceedings of the IEEE International Workshop on Spatio-Temporal Data Models and Languages (STDML)*, Florence, Italy, pp. 441–449.

Erwig, M and Schneider, M, 1999b, Spatio-temporal predicates. *Technical Report 261*, Fern Universität Hagen.

Faria, G, Medeiros, CB and Nascimento, MA, 1998 An extensible framework for spatio-temporal database applications. *Technical Report TR-27*, Time Center, pp. 1–15.

Forlizzi, L, Guting, RH, Nardelli, E and Schneider, M, 2000, A data model and data structures for moving objects databases. *Proceedings of the ACM SIGMOD International Conference on Management of Data*, pp. 319–330.

https://doi.org/10.1017/S026988890400013X Published online by Cambridge University Press

Fowler, M and Scott, K, 1998, *UML Distilled–Applying the Standard Object Modeling Language*. Reading, MA: Addison-Wesley.

Frank, A, 1992, Theories and methods of spatio-temporal reasoning in geographic space (*Lecture Notes in Computer Science*, vol. 639). Berlin: Springer.

Grumbach, S, Rigaux, P and Segoufin, L, 2001, Spatio-temporal data handling with constraints. *GeoInformatica* **5,** 95–115.

Guting, RH, 1994, An introduction to spatial database systems. *VLDB Journal* **4,** 357–399.

Guting, RH, Bohlen, MH, Erwig, M, Jensen, CS, Lorentzos, NA, Schneider, M and Vazirgiannis, M, 2000, A foundation for representing and querying moving objects. *ACM Transactions on Database Systems* **25**(1), 1–42.

Hadzilacos, T and Tryfona, N, 1997, An extended entity-relationship model for geographic applications. *ACM SIGMOD Record* **26**(3), 24–29.

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

Han, J and Kamber, M, 2001, *Data Mining: Concepts and Techniques*. San Mateo, CA: Morgan Kaufmann.

Hazelton, NWJ, 1992, Developments in spatio-temporal GIS. *Proceedings of the 1st Regional Conference on GIS Research in Victoria and Tasmania*.

Hornsby, K and Egenhofer, MJ, 2000, Identity-based change: a foundation for spatio-temporal knowledge representation. *International Journal of Geographical Information Science* **14,** 207–224.

Hunter, GJ and Williamson, IP, 1990 The development of a historical digital cadastral database. *International Journal of Geographic Information Systems* **4**(2).

Jensen, CS, Kligys, A, Pedersen, TB and Timko, I, 2004, Multidimensional data modeling for location-based services. *The VLDB Journal* **13,** 1–21.

Kakoudakis, I and Theodoulidis, B, 1996a, The TAU temporal object model. *Technical Report TR-96–4*, Timelab, UMIST, Manchester.

Kakoudakis, I and Theodoulidis, B, 1996b, The TAU time model. *Technical Report TR-96–5*, Timelab, UMIST, Manchester.

Koubarakis, M *et al*. (eds.), 2003, Spatio-temporal databases: the Chorochronos approach (*Lecture Notes in Computer Science*, vol. 2520). Berlin: Springer.

Langran, G, 1988, A framework for temporal geographic information systems. *Cartographica* **25**(3).

Langran, G, 1992, *Time in Geographical Information Systems*. London: Taylor and Francis.

Lema, JAC, Forlizzi, L, Guting, RH, Nardelli, E and Schneider, M, 2003, Algorithms for moving objects databases. *The Computer Journal* **46**(6), 680–712.

Montgomery, LD, 1994, Temporal geographic information systems technology and requirements: where we are today. *Thesis*, Department of Geography, Ohio State University.

Object Management Group, 1995, *Object Management Architecture Guide*, 3rd edn. Framingham, MA: Wiley.

Parent, CS, Spaccapietra, S and Zimanyi, E, 1999, Spatio-temporal conceptual models: data structures+space+time. *Proceedings of the 7th ACM Symposium on Advances in Geographic Information Systems*, Kansas City, MO.

Pavlopoulos, A, 1998, Evaluation and comparison of spatio-temporal data models. *MSc Dissertation*, UMIST.

Pelekis, N, 2002, STAU: a spatio-temporal extension to ORACLE DBMS. *PhD Thesis*, UMIST.

Peuquet, D, 2001, Making space for time: issues in space-time data representation. *GeoInformatica* **5,** 11–32.

Peuquet, D and Duan, N, 1995, An event-based spatio-temporal data model (ESTDM) for temporal analysis of geographical data. *International Journal of Geographical Information Systems* **9**(1), 7–24.

Peuquet, D and Wentz, E, 1994, An approach for time-based spatial analysis of spatio-temporal data. *Advances in GIS Research, Proceedings* **1,** 489–504.

Price, RJ, Srinivasan, B and Ramamohanarao, K, 1999, Extending the unified modeling language to support spatiotemporal applications. *Asia Technology of Object Oriented Languages and Systems*, 163–174.

Price, RJ, Tryfona N and Jensen, CS, 1999, Extended spatiotemporal UML: motivations, requirements and constructs. *Journal on Database Management, Special Issue on UML* **11**(4).

Ramachandran, A, MacLoad, F and Dowers, S, 1994, Modeling temporal changes in a GIS using an object oriented approach. *Advances in GIS Research, Sixth International Symposium on Spatial Data Handling*.

Raza, A and Kainz, W, 1999, Cell tuple based spatio-temporal data model: an object oriented approach. *Proceedings of the 7th International Symposium on Advances in Geographic Information Systems (ACM-GIS '99)*, Kansas City, MO.

Renolen, A, 1996, History graphs: conceptual modeling of spatio-temporal data. *GIS Frontiers in Business and Science*, vol. 2, International Cartographic Association, Brno, Czech Republic.

Renolen, A, 1997a, *Temporal Maps and Temporal Geographical Information Systems (Review of Research)*. Department of Surveying and Mapping, The Norwegian Institute of Technology.

Renolen, A, 1997b, *Modeling Spatiotemporal Information: The Spatiotemporal Object Model*. Norwegian University of Science and Technology, Trondheim, Norway.

https://doi.org/10.1017/S026988890400013X Published online by Cambridge University Press

Rojas-Vega, E and Kemp, Z, 1995, An object-oriented data model for spatio-temporal data. *Proceedings of the 9th Annual Symposium on Geographic Information Systems*, Vancouver, Canada.

Rumbaugh, J, 1991, *Object Oriented Modeling and Design*. Englewood Cliffs, NJ: Prentice-Hall.

Sellis, T, 1999, Research issues in spatio-temporal database systems. *Proceedings of Advances in Spatial Databases: 6th International Symposium (SSD '99)*, Hong Kong, China.

Sistla, P, Wolfson, O, Chamberlain, S and Dao, S, 1997, Modeling and querying moving objects. *Proceedings of the 13th International Conference on Data Engineering (ICDE13)*, Birmingham, UK.

Su, J, Xu, H and Ibarra, O, 2001, Moving objects: logical relationships and queries. *Proceedings of the 7th International Symposium On Spatial and Temporal Databases (SSTD)*, Redondo Beach, CA, pp. 1–19.

Tansel, AU, Clifford, J, Gadia, S, Jajodia, S, Segev, A and Snodgrass, R, 1993, *Temporal Databases: Theory, Design and Implementation*. New York: Benjamin/Cummings.

Theodoridis, Y, 2003, Ten benchmark database queries for location-based services. *The Computer Journal* **46**(6), 713–725.

Theodoulidis, I and Loucopoulos, P, 1991, The time dimension in conceptual modeling. *Information Systems* **16**(3), 273–300.

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

Tryfona, N and Jensen, CS, 1999, Conceptual data modeling for spatiotemporal applications. *GeoInformatica* **3,** 245–268.

Tryfona, N and Jensen, CS, 2000, Using abstractions for spatio-temporal conceptual modeling. *Proceedings of the 2000 ACM Symposium on Applied Computing*, Como, Italy.

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

Vazirgiannis, M and Wolfson, O, 2001, A spatiotemporal model and language for moving objects on road networks. *Proceedings of the 7th International Symposium On Spatial and Temporal Databases (SSTD)*, Redondo Beach, CA, pp. 20–35.

Wachowicz, M and Healey, RG, 1994, Towards temporality in GIS. In Worboys, M. F. (ed.), *Innovations in GIS*. London: Taylor and Francis.

Wang, F, Hall, GB and Subaryono, 1990, Fuzzy information representation and processing in conventional GIS software: database design and application. *International Journal of Geographical Information Systems* **4,** 261–283.

Wolfson, O, Sistla, AP, Chamberlain, S and Yesha, Y, 1999a, Updating and querying databases that track mobile units. *Distributed and Parallel Databases* **7,** 257–387.

Wolfson, O, Xu, B, Chamberlain, S and Jiang, L, 1998, Moving objects databases: issues and solutions. *Proceedings of the 10th International Conference on Scientific and Statistical Database Management*, Capri, Italy, pp. 111–122.

Worboys, MF, 1994a, A unified model for spatial and temporal information. *The Computer Journal* **37**(1), 27–34.

Worboys, MF, 1994b, Unifying the spatial and temporal components of geographical information. *Information Symposium on Spatial data Handling*.

Worboys, MF, 1998, *Modeling Changes and Events in Dynamic Spatial Systems with Reference to Socio-Economic Units*, Department of Computer Science, Keele University, UK.

Worboys, MF, Hearhshow, HM and Maguire, DJ, 1990, Object-oriented modeling for spatial databases. *International Journal of Geographical Information Systems* **4**(4).

Yuan, M, 1994, Wildfire conceptual modeling for building GIS space-time models. *Proceedings of the GIS/LIS '94*, 860–869.

Yuan, M, 1996a, *Temporal GIS and Spatio-temporal Modeling*. National Center for Geographical Information and Analysis (CD-ROM).

Yuan, M, 1996b, Modeling semantical, temporal and spatial information in geographic information systems. In Craglia, M. and Couclelis, H. (eds.), *Geographic Information Research: Bridging the Atlantic*. London: Taylor and Francis, pp. 334–347.

Yuan, M, 1997, Use of a three-domain representation to enhance GIS support for complex spatio-temporal queries. *Transactions in GIS*.

https://doi.org/10.1017/S026988890400013X Published online by Cambridge University Press