Design Patterns for Spatio-temporal Processes

Christophe Claramunt[1], Christine Parent[2], Marius Thériault[3]

[1] The Nottingham Trent University
Department of Computing
Burton Street
Nottingham NG1 4BU, UK
clac@doc.ntu.ac.uk

[2] Swiss Federal Institute of Technology
Computer Science Department
Database Laboratory
Lausanne, CH-1015, Switzerland
christine.parent@di.epfl.ch

[3] Laval University
Department of Geography
Planning and Development Research Centre
Quebec, G1K 7P4, Canada
marius.theriault@ggr.ulaval.ca

Abstract

Time is an essential dimension to analyse and understand real-world evolution. Although many temporal extensions to spatial models have been proposed, there is still a need to define modelling methods to describe and represent real-world phenomena. This paper presents a set of design patterns modelling spatio-temporal processes expressed in an object-relationship data model. The proposed framework is based on an analysis of spatio-temporal processes and on properties of object-oriented and entity-relationship data models.

1. Introduction

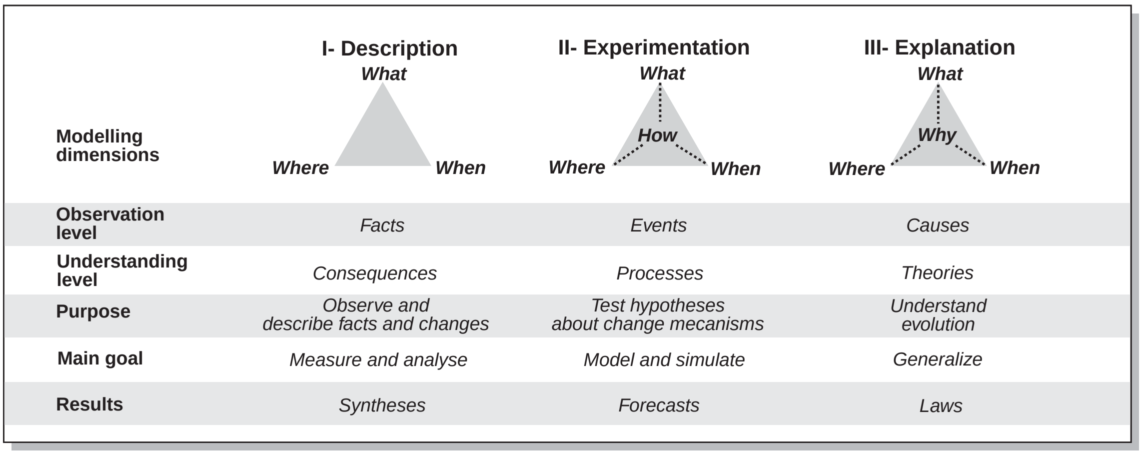
Geographical processes involve entities and changes that are located in both space and time. Therefore, building a taxonomy of spatio-temporal processes (STP) implies an analysis of the representation of facts and events within a space-time framework. According to Peuquet (1994), scientists retain two complementary ontological views to define space-time structures. The absolute approach (introduced by Newton) identifies space as a collection of points and time as a set of instants that exist by themselves. Those dimensions allow the objective measure of entity locations within an independent space-time framework. In contrast, the relative space-time approach (introduced by Leibniz) focuses on real-world entities and uses their mutual relationships to define a subjective space-time canvas. For space, this leads to the distinction between Euclidean geometry (absolute location) and topological properties (relative position). For time, it makes the difference between measured time (absolute chronology) and ordered events (historical sequences). These two complementary paradigms are mandatory to enable the full potential of temporal geographical information systems (TGIS) for real-world phenomena studies (Beller 1991, Langran 1992, Flewelling 1992, Frank 1994, Claramunt 1996).

Providing an efficient system to operate simultaneously on absolute and relative views of space and time implies that geographical (where), temporal (when) and thematic (what) components are implemented using an homogeneous data model (Peuquet 1994, 1995). Because it relates what, where and when predicates in a common three-fold structure, the Peuquet's triad framework can handle complex queries about geo-historical facts and changes. It tracks evolution and describes their consequences. Despite its powerful capabilities this framework does not carry explicit information on how and why changes happen. It records temporal and locational facts that can be used to analyse spatio-temporal patterns and infer underlying processes and relationships. However, it does not provide mechanisms to explicitly describe events and processes and relate changes of one specific entity to actions of other known entities.

As shown in Figure 1, we postulate three levels of scientific knowledge that lead to description, experimentation and explanation. The purpose of description is to represent and analyse facts in order to discover trends and make syntheses. Experimentation goes further, the ultimate goal being to forecast future trends. There is a need to model dynamics behind changes in order to test hypothesis about their action. This problem must be addressed at the process level to discover how things happen and how entities are related into spatio-temporal interaction networks. In this context, the purpose is not to explain why events happen (i.e. discover the causal relationships) but to identify significant properties about the transformation mechanisms and to explicitly record relationships among entities involved in real-world processes. Accordingly, our model objective is not to replace specific simulation models developed in specialised disciplinary contexts but to describe data about processes and their action in order to feed specialised analysis, modelling and simulation tools with appropriate information.

Describing real-world evolution is a complex task. One may observe the status of entities before and after a change occurs, these are facts and consequences. An event is a set of related changes leading to a new status. Events may be observed without knowledge about the mechanisms leading to change. However, we postulate that change happen when a set of active entities or forces transform their environment. Most of the time these transformations do not occur at random because they are constrained to previous status and obey to evolution laws (causal relationships). Discovering these laws is the ultimate goal of science (explanation).

A process is a concept developed by scientists to understand and relate changes occurring in nature (e.g., soil erosion, orogenic, growing processes). It is an intricate mix of facts (status of entities) and transformation mechanisms (ordered changes) that must be considered to structure knowledge about evolution, build models and forecast future situations. Therefore, to describe processes in a data model while keeping the information needed to test hypotheses about evolution mechanisms, we must accumulate evidence that a set of entities are linked into an active system of transformations that is consistent in both space and time. During the data gathering step, there is no immediate need to identify the causal relationships linking events (why). However, one must stay at the experimentation level and record which entities (active and passive) are involved in the transformation, what are the differences between their previous and final status (changes), where (absolute location or relative position) and when (measured time or historical sequence) these changes occur, and, if possible, the mechanisms involved (how). All these facts are clues for scientists in their attempt to understand evolution (test hypotheses, formulate theories and discover the evolution laws).



	I- Description	II- Experimentation	III- Explanation
Observation level	*Facts*	*Events*	*Causes*
Understanding level	*Consequences*	*Processes*	*Theories*
Purpose	*Observe and describe facts and changes*	*Test hypotheses about change mecanisms*	*Understand evolution*
Main goal	*Measure and analyse*	*Model and simulate*	*Generalize*
Results	*Syntheses*	*Forecasts*	*Laws*

Figure 1. Three levels of scientific knowledge

Processes may be modelled using a top-down or a bottom-up approach. Users may describe evolution using high semantic level concepts based on an a priori taxonomy of processes that is application or domain dependent (e.g. meteorological processes). For instance, they identify a global process (e.g. warm front) and relate it to specific events (rain, thunderstorm, etc.). However, processes definitions and laws must be provided by system designers to permit this deductive modelling approach. On the other hand, the bottom-up procedure leads to inductive analysis. In this context, theoretical requirements are kept minimal and users provide only a description of observed evolution. Typically, they want to accumulate data about changes and transformation mechanisms to permit statistical and systemic analysis of spatio-temporal patterns. The ultimate goal being to test hypotheses, the data gathering procedure must avoid any a priori that can introduce some bias in the analyses.

The specific purpose of this paper is to propose a standard way of designing STPs, valid for every data model that supports the time and space dimensions. It presents a taxonomy of basic STPs in Section 2. The database modelling issues are introduced in Section 3. Three design patterns expressed in a spatio-temporal object-relationship data model are presented in Section 4. Section 5 surveys related work and Section 6 concludes the paper.

2. A Taxonomy of Basic Spatio-temporal Processes

Describing geometric transformations of an independent spatial entity implies change on four orthogonal attributes (Bertin 1983, Clementini 1993, Edwards 1993, Claramunt 1995): **shape** (form of its boundary), **size** (area of its interior), **orientation** (compass direction of its major and minor axes) and **location** (position of its gravity centre measured with geographic or Euclidean co-ordinates). As shown in Figure 2, these four geometric parameters lead to five combinations corresponding to basic STPs: no change (STABILITY), change of shape (DEFORMATION), size (EXPANSION and CONTRACTION), orientation (ROTATION) or location (TRANSLATION). All these basic processes apply to any geometric type on 2- or 3-dimensional spaces, except for points since they are dimensionless and cannot shrink, rotate or deform.

Basic STP	**Geometric types**	**Shape**	**Size**	**Orientation**	**Location**
Stability	any	constant	constant	constant	constant
Deformation	any but point	***changed***	constant	constant	constant
Expansion	any	constant	***growing***	constant	constant
Contraction	any but point	constant	***shrinking***	constant	constant
Rotation	any but point	constant	constant	***changed***	constant
Translation	any	constant	constant	constant	***changed***

Figure 2. Basic evolutions of a spatial entity

Considering geometric relationships, there is a need to extend the previous model in two directions. Firstly, by considering overlapping constraints on the directional (shape, size and orientation) and locational changes. Secondly, by introducing a mechanism to handle simultaneous shape and size transformations of spatial entities exchanging their land coverage while they maintain topological consistency. The first extension constrains the movement (isomorphic changes) of an entity (or set of entities):

- it is not constrained and the entity can occupy any position or orientation in space (TRANSLATION or ROTATION),
- it can occupy any free position or orientation in space (**exclusive** TRANSLATION or ROTATION),
- it may/must conquer a position or orientation presently/previously held by an entity(ies) of any type (**elective/mandatory immediate/delayed** SUCCESSION),

- it may/must conquer a position or orientation presently/previously held by an entity(ies) of the same type (**elective/mandatory immediate/delayed homogeneous** SUCCESSION),
- it may/must exchange its position or orientation with an other set of entities of any type (**elective/mandatory immediate/delayed** PERMUTATION),
- it may/must exchange its position or orientation with an other set of entities of the same type (**elective/mandatory immediate/delayed homogeneous** PERMUTATION).

The second extension constrains deformation, expansion and contraction of adjacent entities' territory by enforcing transitive topological relationships among their boundaries with/without exhaustive or exclusive land coverage rule:

- moving the boundary of an entity must be synchronised with similar modification of adjacent entities of the same type (**homogeneous** RE-ALLOCATION within a coverage),
- moving the boundary of an entity is synchronised with similar modification occurring on entity(ies) of different type (**heterogeneous** RE-ALLOCATION between coverages),
- adding an interior boundary into a topological coverage can split an entity that disappears and is replaced by new ones (SPLIT),
- removing a boundary from a topological coverage will join adjacent entities that disappear and are replaced by a new entity (UNION).

All these basic STPs involve at least one entity that has its own spatial attributes transformed by the process while the effects may also concern other entities of the same or different types. However, in most applications, spatial entities are inter-dependent and it is then necessary to handle inter-relation networks formed by their joint spatial transformation. While the evolution of a single independent entity is modelled using an endogenous approach (process in which the entity changes by itself), modelling transformations based on the interaction of many entities involves some extensions. These exogenous processes must express the intrinsic link between individuals' evolution while they offer a mechanism to distinguish between the agent(s) provoking the change (active entities) from those being modified by its action (passive entities). It is not possible to establish an exhaustive list of these functional processes because there is no universal criteria to classify them. Nevertheless three supplementary basic STPs can handle most situations related to natural, economical and social sciences:

- A set of active entities produces a set of new entities (appearing passive entities) while consuming an other set of components entities (disappearing passive entities). The PRODUCTION process is necessary to carry the systemic association between all involved entities and relate their simultaneous appearance and disappearance to the action of producers.

- A first set of entities creates a new set of entities of the same type. In biological sciences, the REPRODUCTION process is used to link parents and children even if the detailed mechanisms of life transmission remain unknown.
- The TRANSMISSION process occurs when a set of receiver entities (passive) has its attributes modified by some contact with a set of transmitter entities (active). This kind of relationships has obvious applications in epidemiology and communication or may as well be used to model transmission of forces between moving balls over a billiard table.

Complex real-world processes are described by combining this minimal set of general low-level evolution mechanisms (basic STPs) to define sequences, conjunctions, disjunctions or cycles of events. These complex processes can be described using the Event Pattern Language (Gehani 1992, Motakis 1995) to link many basic STPs and define composite STPs (Claramunt 1996). The EPL language is already used for active database applications and provides logical constructors for the representation of composite STPs, it includes an order syntax and coordinating rules. Together, STPs and EPL, provide a standardised and flexible data model that can enable processes analysis while putting minimum constraint over their expression.

3. Issues in Database Modelling of Spatio-temporal Processes[1].

Every dynamic process in the real world happens in space and time. A loose definition of spatio-temporal process may therefore include all processes modifying some property of an object (Figure 3):

- its existence: appearance and disappearance of an object (e.g. a new building is built, another one is destroyed), temporary suspension of an object activity and/or accessibility (e.g. skating rings are closed during summer),
- its classification: an object instance may "move" from one object class to another one (an agricultural piece of land becomes an industrial lot, a village becomes a city),
- the values of its thematic attributes (e.g. the price of a building changes),
- the values of its spatial attributes (i.e. geometric or topologic features),
- the values of its time-related attributes (e.g. the delivery date for an order is delayed),
- its thematic relationships to other objects (e.g. the owner of a building changes),

---

[1] As this section develops the database perspective, we shall use the database terminology, i.e. we use the term "data model" to refer to the abstract modelling concepts (e.g. object type, etc., otherwise termed the meta-model) and the term "schema" to refer to a description of application specific object types, etc. for a specific database (otherwise termed the data model). We also use the term object, instead of the term entity, as we refer to the database representation of a real world entity.

- its spatial relationships to other objects (e.g. a village adjacent to a city becomes included into this city boundaries after an administrative regrouping).

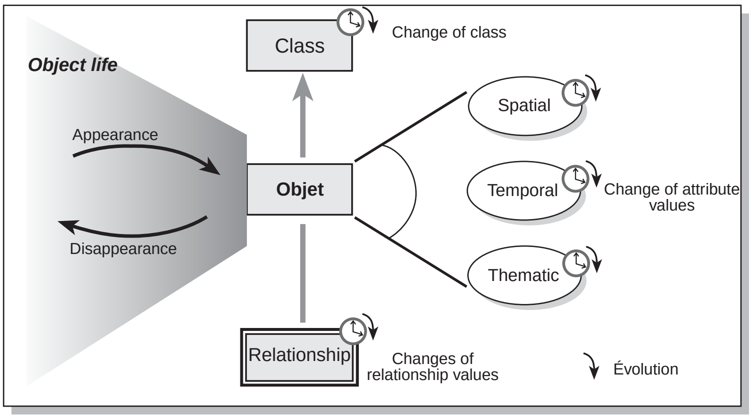


Figure 3. Object life and changes

A generic model for the description of processes would, for instance, represent the spread of a disease using the same modelling features as for another process causing the value of stock items to increase or decrease. To the best of our knowledge, such a generic change data model does not exist yet.

A major contribution in terms of representing change is currently provided by the temporal database approach (Tansel 1993, Snodgrass 1995). Temporal databases associate to each item, designated as temporal, a set of time intervals which specify the item temporal validity. Time intervals represent changes in the real world (valid time) or the transcription of these changes in the database (transaction time). Thus defining an object as temporal allows the recording of the corresponding entity life cycle (appearance, disappearance, suspension). Classification changes are recorded by time-stamping the object memberships in each corresponding class. However, facilities for describing relevant classification changes are still limited. Temporal databases similarly represent the history of changes in attribute values and, at least partially, in relationships. Note that from the database perspective defining an attribute on a DATE domain is not a temporal specification: DATE-type attributes are seen and managed as thematic attributes.

While temporal databases are designed to store historical data (the past) as well as present and possible future data (e.g. for planning purposes), they are not designed to record which processes activate a change. Additional concepts are thus needed to model the processes which cause the changes. We focus hereinafter on those processes which act on the geometric features of an entity (its spatial type or location) or on its spatial relationships. Consistently with the discussions in previous sections, we refer to these as spatio-temporal processes, adopting a more restricted definition of this term than the looser one mentioned above.

Previous sections have shown that different applications are interested in recording ST processes at very different levels of abstraction, from the most basic ones (e.g. a translation) to complex ones which mostly are specific to a given application domain. In terms of data modelling, it does not make sense to enrich the data model with a new modelling concept for each type of STPs required by the application. This would end up in an unlimited amount of modelling primitives, i.e. a data model of a complexity unmanageable by database users. On the other hand, users should be able to define application specific STPs, beyond the standard ones embedded into the system. This results in an extensible hierarchy of process type definitions, organised as a generalisation/specialisation hierarchy as shown in Figure 4.

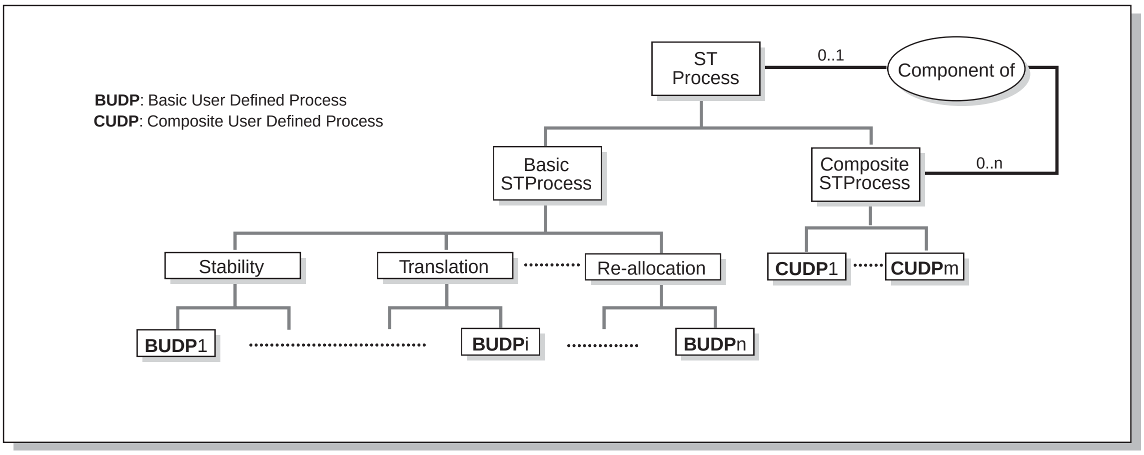


Figure 4. The inheritance hierarchy of STP types

As shown in the Figure 4, STPs are split into basic ones (the basic processes identified in Section 2) and composite ones, whose composition has to be explicitly defined. The latter are by definition user-defined. The former includes the set of system defined STPs, but can be extended, through specialisation, by designers as appropriate to their needs (for instance, to add application specific attributes to the description of a system defined process). The definition of a STP type should include the following properties:

- its name,
- its process type (one of the basic types STABILITY, EXPANSION, CONTRACTION, DEFORMATION, TRANSLATION, ROTATION, SUCCESSION, PERMUTATION, PRODUCTION, REPRODUCTION, TRANSMISSION, SPLIT, UNION, RE-ALLOCATION, or the composite type),
- its temporal data, i.e. the time when the process takes place. It can be an instant, a time interval, a set of time intervals or a set of instants. The time granularity (the unit of measure) needed to express time must be defined.

- its composition, if any. This composition is expressed through a control language, e.g. the EPL language (Gehani 1992), which provides constructs for defining sequences, conjunctions and disjunctions of processes.
- the objects on which the process is operating. These objects are:
  - in case of an endogenous process, a characteristic of a single object (its life cycle, its classification or its geometry),
  - in case of an exogenous process, the set of involved objects,
- the thematic and spatial attributes describing the process, if there is any,
- integrity constraints, if any. For example, a constraint may restrict the geometry of the target objects related to one of the source objects.

The easiest way of materialising the description of ST processes in a database is to define object types as suggested in Figure 4. For each class of similar processes of interest to the application, the database designer can define a special object type to represent this class. Actual processes will be materialised as object instances of the corresponding type. However, to cope with the diversity of application requirements, we propose to add a STP dimension as well to the other usual modelling primitives used in classical data models, i.e. to relationship types and to attributes. Adding the STP dimension to relationship types is of particular interest for the description of exogenous processes, in particular for processes which involve only one object in each source and target type (i.e. one real world process is represented as one relationship instance). Their representation as relationship types between the involved object types provides the best visibility in the schema. Thus, for instance, the possibility for objects of type A to permute with objects of type B can be explicitly defined as a special STP relationship type between A and B (PERMUTATION). In the same way, a cyclic STP relationship type, linking A to itself, will be defined if objects of type A can permute with other objects of the same type (homogeneous PERMUTATION).

Finally, we propose to extend the description of the geometry attribute with an STP clause. This allows both the recording of the different STPs which acted on the geometry of the object and the enumeration of the STPs which are allowed to act on objects of the type. This provides an interesting feature for the description of integrity constraints.

In terms of diagrammatic standards, which are highly appreciated by users, at least two options may be considered. Basically, the STP dimension may be visualised either through the use of dedicated symbols or through the embedding of specific icons. The former would lead, for instance, STP object types to be represented by say diamonds where normal object types are represented by rectangles. In the same case, the latter would show STP object types as rectangles with a STP icon. The same applies to STP relationship types and attributes. The actual choice is a matter of preference and of diagram readability, not of scientific relevance.

4. Patterns for Designing Spatio-temporal Processes

This section stems from the discussion in Section 3 and proposes a standard way for representing the STPs in a database. Three design patterns (i.e. generic schematic structures) are designed, which can be used for any application requiring the modelisation of STPs. In order to be generic, the proposed patterns are expressed using a conceptual data model (logical models imply implementation choices which would bias the conceptual perspective).

Thus, the underlying data model has to be:

- conceptual,
- spatial, providing a discrete view, in order to describe spatial entities,
- and temporal, in order to represent evolving entities.

Many conceptual spatial data models, e.g. (David 1993, Clementini 1994, Camara 1994, Tryfona 1995, De Oliveira 1997) and temporal data models, e.g. (Tansel 1993, Snodgrass 1995) have been defined. Recent proposals integrate the time dimension within spatial data models (Allen 1995, Story 1995, Bedard 1996). Nonetheless, these spatio-temporal models do not provide any support for modelling real-world processes. We propose to build the process modelling facilities on top of the spatio-temporal MADS object-relationship data model (Parent 1997). MADS aims at the same goals as OGIS, the geodata specification currently proposed as a standard for data exchange between GIS (OGIS), while being simpler, more orthogonal and allowing explicit description of spatial relationships. Nevertheless, the patterns we propose for modelling spatio-temporal processes are not specific to the MADS model; similar patterns can easily be built for any object oriented or entity relationship spatio-temporal data models.

The following sequel introduces a short description of the MADS model, and then discusses the representation of spatio-temporal processes.

MADS represents spatiality at different levels: object type, attribute and relationship type. A spatial object type is characterised by the inclusion of a specific attribute, named geometry, whose domain is one of the of supported abstract spatial types: point, line, area and their specialisations or generalisations (MADS supports an extensible hierarchy of 11 spatial types). Spatial types are represented in schema diagrams by specific icons, as illustrated by the object type Parcel in Figure 5. Spatial relationships may be defined among spatial objects to express integrity constraints on their geometries or the application spatial semantics. The type of the spatial relationship (i.e. inclusion, adjacency...) is represented in schema diagrams by specific icons. Any object type or relationship type can have spatial attributes whose domain is one of the abstract spatial types.

Defining an object type (or a relationship type or an attribute) as temporal means that the designer wants to represent its history: the database will represent all the different object instances/values, associated with their corresponding valid times. For example, defining the object type Parcel as temporal means that the creation and deletion time of each parcel will be stored in the database. Defining the geometry of Parcel as temporal means that for each update of the geometry, the system will add the new geometry into the database while keeping the previous ones with their updated valid time.

Let us illustrate some of the MADS features through an example. In Figure 5, Parcel is a spatio-temporal object type whose textual description is:

```
OBJECT Parcel
  TEMPORAL DAY ,
  GEOMETRY AREA TEMPORAL DAY ,
  ATTRIBUTES: (
    number: STRING [1:1] ,
    use: STRING [0:1] TEMPORAL DAY )
  END Parcel
```

Comment: The temporal clause specified at the object level instructs the system to keep track of the life cycle of the parcel: when it was created, when its existence was suspended or reactived, when its was deleted in the real world. The geometry clause specifies that parcels are areas. The temporal specification associated to geometry instructs the system to keep track of the values of the geometry over time, with a time granularity of a day. Parcel has two monovalued attributes, *number* and *use*. *Number* is mandatory (cardinality is [1:1]), while *use* is optional (cardinality is [0:1]). *Use* is temporal: the system will keep track of its values.

To describe spatio-temporal processes we may use any of the three main concepts of the data model, object type, relationship type and attribute. Moreover, each concept can be spatial or not, temporal or not. As previously mentioned, the easiest - and also the more efficient - solution is to use object types. Thus, in a first step we propose a description of processes as object types and in a second step we discuss in which cases and how this description can be simplified using relationships or attribute concepts.

4.1. Describing processes as object types

Each set of similar processes is described as an object type whose characteristics implement the properties defined in Section 3:

- name: a property describing the type -not the instances-. It is the name of the object type.
- temporal data: the description of the process life cycle. Thus the object type will be temporal. Temporal integrity constraints bind its life cycle to objects related to this process.
- process type: a property describing the type. In object oriented models supporting the class attribute concept, process type is a class attribute with an enumerated domain. The MADS

model should be extended, or an integrity constraint should be attached to the object type asserting that the value for the process type attribute is constant.

- composition: if the application needs a detailed description of composite processes with their component processes, component processes are also represented as objects. Composite processes and component processes will be linked through aggregations, which are a specific type of relationships of MADS, with a "part-of" semantics. Moreover the EPL description of the generic composition of all the processes of this type should be kept. Thus the system will be able to check the composition of the process instances (i.e. the aggregation instances). This EPL description is a property of the object type. It will be a constant attribute with the set of EPL sentences as domain.
- the objects on which the process is operating: a binary relationship is defined for each role assumed by one of those objects. For example, in a split process, there are two roles, the source object, and the target objects.
- thematic attributes: descriptive attributes of the object type.
- spatial attributes: there are two solutions: describing the process as a spatial object type, or adding one (or several) spatial attribute(s) to the object type. The choice depends upon the semantics of the process and attribute. Describing a process as a spatial object type relate it to other spatial (process) objects through spatial relationships.
- integrity constraints: they are attached to the object type.

Let us use an example to illustrate the different ways of recording processes. Assume a land management application which records processes modifying parcels:

- the processes (called UpdateGeometry) that change the geometry of a parcel,
- the processes (called LimitChange) that modify the limit between two contiguous parcels due to its correction or to the fact that one owner is selling a strip of his/her parcel to his/her neighbour,
- the processes (called ParcelsRegrouping) that restructure the land in a given area. They act on many parcels simultaneously, modifying the boundaries between parcels, uniting parcels and splitting parcels.

In Figure 5, ParcelsRegrouping processes are described as an object type, the textual description is given below.

```
OBJECT ParcelsRegrouping
  TEMPORAL INTERVAL MONTH ,
  STP COMPOSITE ,
  COMPOSITION = ( Re-Allocation(Parcel) & Split(Parcel) & Union(Parcel))
  GEOMETRY AREA  */ The area involved by the grouping /* ,
  ATTRIBUTES: (
    comment: STRING )
  END ParcelsRegrouping
```

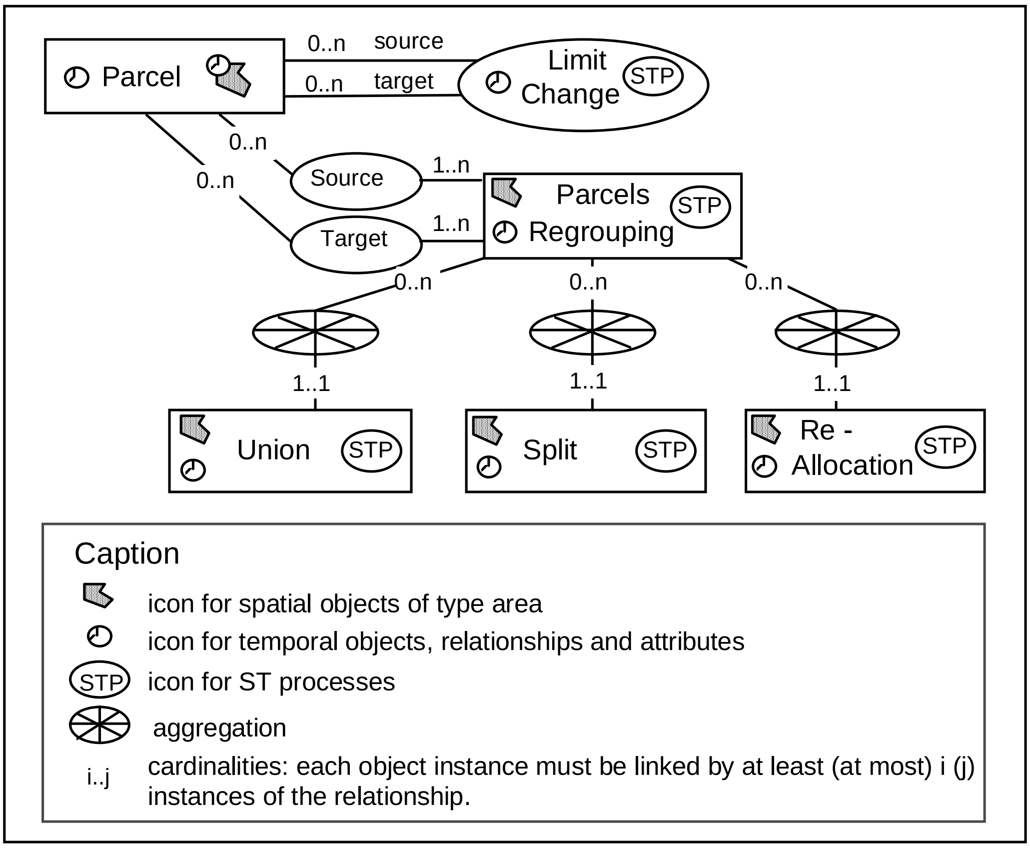


Figure 5. Land management application example

Comment: ParcelsRegrouping is a composite process, composed of Re-Allocation, Split and Union basic processes as shown by the three aggregation relationships in Figure 5. The composition clause of the textual description of ParcelsRegrouping specifies that the component processes run simultaneously (conjunction operator denoted as &). ParcelsRegrouping is linked to Parcel through the relationships, Source and Target, which describe the source and target parcels of the process.

One classic way of implementing such process descriptions as object types in an object oriented database management system is to create a subclass, say STPobject, of the generic class Object, with attributes describing the temporal, STP and composition clauses.

4.2. Describing processes as relationship types

The description of a set of processes as an object type, plus its relationships to the spatial object types involved in the processes, can be simplified if it satisfies the following two conditions:

- the process type must be a basic one (in entity relationship models a relationship cannot be linked to another relationship),
- for each instance of process, the number of involved spatial objects is the same (in entity relationship models each role of a relationship instance is connected to one and exactly one object instance).

In this case the object type and the relationship types can be merged into a single relationship type. Such a transformation is a well known equivalence of two entity relationship schemas (Hainaut 1996).

In the land management example, ParcelsGrouping cannot be simplified, while LimitChange, which is a specialisation of the basic process Re-Allocation involving exactly two parcels, can either be described as an object type and two relationships, or as a unique relationship. This second solution is illustrated by the Figure 5.

```
RELATIONSHIP LimitChange
  TEMPORAL INSTANT DAY ,
  STP RE-ALLOCATION ,
  ROLE SOURCE Parcel [0:N] ,
  ROLE TARGET Parcel [0:N] ,
  ATTRIBUTES: (
    deed-number INTEGER [0:1] ,
    lawyer [0:1] (
       lawyer-name STRING [1:1] ,
       lawyer-address STRING [1:1] ) )
  END LimitChange
```

4.3. Describing processes as attributes

The description of basic processes that model the evolution of a single entity (e.g. expansion, contraction, deformation), can even be simpler than a relationship type. A mere attribute is sufficient, as these processes are linked to one (and exactly one) entity (i.e. endogeneous processes involving an independent entity according to our process classification). In this case, the generic description as an object type plus a relationship type reduces to an attribute. More precisely, as each of these processes acts on the geometry of an entity, the data describing the process should be associated to the corresponding modified value of the geometry. In other words, the geometry of the spatial entity is a complex attribute with two components: the geometry itself and the data describing the process.

The new definition of the Parcel object type contains a STP clause as part of the geometry description:

```
OBJECT Parcel
  TEMPORAL DAY ,
  GEOMETRY ( AREA TEMPORAL DAY ,
             STP ( DEFORMATION ,
                   TEMPORAL DAY ,
                   comment STRING [1:1] ) )
  ATTRIBUTES: ( number: STRING [1:1] ,
                type: STRING [0:1] TEMPORAL DAY )
  END Parcel
```

Comment: The description of the STP clause of geometry contains three kinds of information:

- the process type (deformation [2]),
- the temporal data which states that the process time granularity for recording the ST processes of geometry is the day,
- thematic and spatial attributes. In the above example, *comment* is a thematic attribute.

These three patterns describes ST processes as object types, relationship types and attributes. They can be used in any application, spatial or aspatial, that needs to record the processes involving entity evolutions, whether spatial or aspatial.

5. Spatio-temporal Related Work

We compare our spatio-temporal model principles to some of the researches developed so far. Those approaches can be classified in two main categories:

- Those that consider the time dimension as a fourth spatial dimension (Langran 1992, Hazelton 1992, Worboys 1994),
- Those that consider time as an independent dimension (Peuquet 1994, Claramunt 1995).

Cartographic-oriented views integrate the time dimension as a fourth spatial dimension. They represent a first set of attempts to introduce the time component into Geographical Information Systems. Successive temporal spaces are merged into a same cartographic structure (i.e. concept of amendment vectors, Langran 1992). New geometric primitives are successively recorded within the spatial database (e.g. a parcel limit change leads to the creation of a new time-stamped line in the database).

---

[2] In reality, these UpdateGeometry processes are more sophisticated. They can be deformation, contraction or expansion ones. But, for the sake of the example, let us consider only basic deformation processes.

However, these cartographic views of time lead to a series of limitations:

- Spatio-temporal entities are not explicitly and homogeneously represented as the time dimension is integrated at the geometrical primitive level. This is not satisfactory both from the conceptual level and from the implementation point of views. Furthermore, as the geometrical primitives are time-stamped, spatial data may be duplicated when a same geometric primitive is shared by distinct spatial entities overlapping in time. We may consider the example of a line geometrical primitive shared by a parcel and a building, if the building spatial entity is deleted while the parcel one is still existing, the line primitive will be duplicated. Finally, linking the spatial database to a non spatial database may be a complex task if spatial entities are not homogeneously represented.
- Composing a time slice from these cartographic approaches is not straightforward as the visualisation of a map view for a specific time-stamp imposes to recompose the set of spatio-temporal objects included in the considered time-stamp from the geometrical level.
- Another important limitation of using time as a space dimension is the additional resulting topology complexity. The topological set of relationships is modified for each spatial change. That may lead to cumbersome situations for retroactive spatial changes.
- This cartographic view of time will finally emphasise the difficulty to integrate additional space dimensions.

A second class of proposals consider time as a specific and independent dimension. A triad representation develops an integrated approach of time, space and object-related components (Peuquet 1994). The triad framework integrates three fundamental and complementary views: location-based (e.g. a series of spatial snapshots), time-based (e.g. an ordered list of events), and object-based (e.g. the evolution of a spatial entity). As previously mentioned, this paper retains this conceptual framework. An object-relationship model represents spatial entities and allows to model changes at the object, relationship or attribute levels depending on the application requirements. This approach avoids most of the limitations of the cartographic view as time and space are considered as two independent dimensions.

Current spatio-temporal models are oriented toward the representation of the evolution of spatial entities. However, none of them provides basic constructs to specify the underlying knowledge describing the processes occurring in the real-world. Although event representations and languages have been recently integrated in conceptual spatial models (Frank 1994, Peuquet 1995), they do not address the representation of spatio-temporal processes. Using the same object-oriented approach, we define design patterns to describe spatio-temporal processes.

6. Conclusion

Modelling both space and time allows to cover a large area of applications related to the management, the analysis and the understanding of natural and anthropic phenomena. Although spatial information systems provide new perspectives to real-world phenomena analysis and understanding, they are still lacking of spatio-temporal modelling methods. This paper proposes solutions for representing spatio-temporal processes.

To accommodate the complementary absolute and relative views of time, we have presented the principles of an integrated spatio-temporal model. The framework provides a modelling support to the description and experimentation of real-world phenomena. It is based on a taxonomy of spatio-temporal processes which leads to the definition of the basic evolution of a spatial entity and of the constrained evolution of several spatial entities. In order to materialize the description of processes, their properties are analysed and a generalisation/specialisation hierarchy of spatio-temporal process types is proposed. The hierarchy is extensible, allowing designers to define new process types either as specializations of the basic predefined process types or as new composite process types whose composition has to be described explicitly. Three generic design patterns are defined, allowing to describe processes as object types, relationship types and attributes. The patterns are defined in a spatio-temporal data model that supports time at the life cycle level for objects and relationships and at the value level for attributes. The patterns can easily be adapted to any other data model offering similar features. Further developments include the application of the method to environmental and urban studies.

Acknowledgements

We gratefully acknowledge Stefano Spaccapietra for his valuable comments on earlier versions of this paper.

References

ALLEN, E., EDWARDS, G. and BÉDARD, Y., 1995, Qualitative Causal Modelling for Temporal GIS. In *Spatial Information Theory: A theoretical basis for GIS*, Frank, A. U. and Kuhn, W. Eds., Berlin, Springer-Verlag, 397-412.

BÉDARD, Y. et al., 1996, Adapting Data Models for the Design of Spatio-temporal Databases, in *Computer, Environment and Urban Systems*, Vol. 20, N°1, 19-41.

BERTIN, J., 1983, *Semiology of Graphics*, Madison, Univ. of Wisconsin Press, 1983.

CAMARA, G. et al., 1994, A Model to Cultivate Objects and Manipulate Fields, *Proceedings of the 2nd ACM Workshop on Advances in Geographic Information Systems*, N. Pissinou, K. Malki, Eds., Gaithersburg, Maryland, USA, December 1-2, 1994.

CATTELL, R. G. C., 1994, *The Object Database Standard: ODMG-93*, Morgan Kaufmann, 169p.

CHEYLAN J. P. and LARDON, S., 1993, Toward a Conceptual Model for the Analysis of Spatio-temporal Processes. In *Spatial Information Theory*, Frank, A. U. and Campari, I. Eds., Springer-Verlag Berlin, 158-176.

CLARAMUNT, C. and THÉRIAULT, M., 1995, Managing Time in GIS: An Event-oriented Approach. In *Recent Advances in Temporal Databases*, Clifford, J. and Tuzhilin, A. Eds., Berlin, Springer-Verlag, 23-42.

CLARAMUNT, C. and THÉRIAULT, M., 1996. Toward Semantics for Modelling Spatio-temporal Processes within GIS. In *Advances in GIS Research,* Kraak, M. J. and Molenaar, M. Eds., Delft, Taylor and Francis, 47-63.

CLEMENTINI, E., DI FELICE, P. and VAN OOSTEROM, O., 1993. A Small Set of Topological Relationships Suitable for End-user Interaction. In Abel D. J. and Ooi D. C. Eds. *Advances in Spatial Databases*. Singapore, Springer-Verlag, 277-295.

CLEMENTINI, E. and DI FELICE, P., 1994, Object-Oriented Modeling of Geographic Data, *Journal of the American Society for Information Science*, Vol.45, N°9.

DAVID, B. et al., 1993. GeO2: Why Objects in a Geographical DBMS ? Abel D. J. and Ooi D. C. Eds. *Advances in Spatial Databases*. Singapore, Springer-Verlag, 264-276.

DE OLIVEIRA, J. L., PIRES, F. and MEDEIROS, C. B., 1997, An Environment for Modelling and Design of Geographic Applications, *Geoinformatica,* Kuwer Academic, 1(1): 1-24.

EDWARDS, G., GAGNON, P. and BÉDARD, Y., 1993. Spatio-temporal Topology and Causal Mechanisms in Time-integrated GIS: from Conceptual Model to Implementation Strategies. *Proceedings of the Canadian Conference on GIS'93*. Ottawa, Canada, 842-851.

FLEWELLING, D., EGENHOFER, M. J. and FRANK A. U., 1992. Constructing Geological Cross Section with a Chronology of Geologic Event. In *Proceedings of the 5th International Symposium on Spatial Data Handling. Charleston, South Carolina. IGU Commission* on GIS, 544-553.

FRANK, A. U., 1994, Qualitative Temporal Reasoning in GIS - Ordered Time Scales. In *Proceedings of the Sixth International Symposium on Spatial Data Handling,* Waugh, T. C. and Healey, R. C. Eds., London, Taylor and Francis, 410-430.

GEHANI, N. H., JAGADISH, H. V. and SHMUELI, O., 1992, Composite Event Specification in Active Databases: Model and Implementation. In *Proceedings of the 18th VLDB Conference*, Vancouver, Canada, 327-338.

HAINAUT, J.-L., 1996, Specification Preservation in Schema Transformation Applications to Semantics and Statistics. *Data & Knowledge Engineering*, 16 (1), Elsevier Publications.

HAZELTON, N. W. J., 1992, Developments in Spatio-temporal GIS. In *Proceedings of the First Regional Conference on GIS Research in Victoria and Tasmania* (Ballarat: Ballarat University College).

LANGRAN, G., 1992, *Time in Geographic Information Systems*. Taylor & Francis, 189 p.

MOTAKIS, I. and ZANIOLO, C., 1995. Composite Temporal Events in Active Databases: A Formal Semantics. In Clifford J. and Tuzhilin A. Eds., *Recent Advances in Temporal Databases*, Berlin, Springer-Verlag, 332-354.

OGIS, http://www.opengis.org

PARENT, C., SPACCAPIETRA, S. and ZEMANYI, E., 1997, Conceptual Modeling for Federated GIS over the Web. In *Proceedings of the ISPJ International Symposium on Information Systems and Technologies for Network Society,* Fukuoka, Japan, September 24-26.

PEUQUET, D. J., 1994, It's About Time: A Conceptual Framework for the Representation of Temporal Dynamics in Geographic Information Systems. *Annals of the Association of the American Geographers*, 84 (3), 441-461.

PEUQUET, D. J., and DUAN N., 1995, An Event-based Spatiotemporal Data Model (ESTDM) for Temporal Analysis of Geographical Data. *International Journal of Geographical Information Systems*, 9 (1), 7-24.

SNODGRASS, R. T. et al., 1995, *The TSQL2 Query Language*. The TSQL2 Language Design Committee, Kluwer Academic Publishers, 674 p.

STORY, P. A. and WORBOYS, M. F., 1995, A Design Support Environment for Spatio-Temporal Database Applications. In *Spatial Information Theory: A theoretical basis for GIS*, Frank, A. U. and Kuhn, W. Eds, Berlin, Springer-Verlag, 413-430.

TANSEL, A. U. et al., 1993, *Temporal Databases: Theory, Design and Implementation*, Benjamin Cummings Publishing Cie, 633 p.

TRYFONA, N. and HADZILACOS, T., 1995, Geographic Applications Development: Models and Tools for the Conceptual Level. In *Proceedings of the ACM GIS'95 Advances in Geographic Information Systems*, Baltimore, Maryland, USA, December 1-2, 19-28.