

A Multiple Layered Functional Data Model to Support Multiple Representations and Interoperability of GIS: Application to Urban Management Systems

Mohamed El Adnani
Département Informatique
Faculté des sciences
Marrakech, Maroc
md-
eladnani@ucam.ac.ma

Kokou Yétongnon
Laboratoire LE2I
Université de Bourgogne
Dijon, France
kokou@u-bourgogne.fr

Djamal Benslimane
Laboratoire LISI
Université de Lyon 1
Villeurbanne, France
benslima@u-bourgogne.fr

ABSTRACT

This paper presents a multi-layered functional data model to support multiple representation and information sharing among distributed spatial information systems. The key idea is that a geographical space (for example an urban space) can be viewed as a set of abstract spatial functionalities on which different application classes can be derived to manipulate objects of the same geographical space and share the same abstract generic functionalities. The object classes define multiple views or representations of a domain. For example in urban management, transportation systems, water supply and sewage systems applications can share generic network functionalities defined by graph related terminology and the corresponding graph traversal operations. The multi-layered functional data model consists of two types of generic functional levels. First, a generic functional level consisting of primitive geometrical types and one or more generic layers. The generic layers provide application oriented abstract functions that are based on the primitives geometrical types. The second level consists of one or more domain specific functionalities that specialize and refine the abstract functions of the first level.

Categories and Subject Descriptors

H.2.8 [DataBase Management]: Database Applications
▪ Spatial databases and GIS; **H.2.1 [Database Management]:** Logical Design
▪ Data models; **H.2.4 [Database Management]:** Systems
▪ Object-oriented databases

General Terms

Management Design Languages Theory

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. To copy otherwise, or republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee
GIS '01, November 9-10, 2001, Atlanta, Georgia, USA
Copyright 2001 ACM 1-58113-443-6/01/0011...\$5.00.

Keywords

Spatial data model, Multiple representation, Interoperability

1. INTRODUCTION

Spatial information systems are increasingly used to support applications in which group of users or applications can interact and cooperate. The requirements and needs of these applications may create different representations or view points over a common geographical space. For example, urban management decision support systems often involve making various information systems such as traditional databases, file based systems and GIS cooperate and interact. Often, these information systems model and record objects of a common geographical space differently. For instance, a street can be modeled as a single segment in one application (e.g. a map based management system) and as a set of nodes and their topological relationships or arcs in another application (e.g. a traffic management systems). Multiple representations of objects in a common application space can be defined and stored in the same database. In traditional database systems, multiple representations of objects are maintained by 1) multiple views over the same sets of data, 2) inheritance hierarchy that allows the definition multiple structures over the same set of objects or 3) dynamic objects modeling concepts that allow objects to change their structure and behavior over time. With the emergence and development of network environment, application needs may span several interconnected heterogeneous information systems with different data models and semantics. Multiple representations or view points of the same entities can exist in the different information systems. The need to allow multiple representations of objects and to enable interaction and cooperation among different applications or users have created new spatial information systems challenges and criteria:

- support for multiple representations of geographical or spatial objects.
- support for dynamic objects that can change type and have multiple instances or versions.
- support for heterogeneous data sets and semantic conflict resolution of conflicts.

- support for complex queries that span multiple information systems, thus allowing data sharing and exchange among the participating systems.

1.1 Background

Multiple representations or viewpoints over a common space can arise in different ways. Different properties, set of attributes can be defined over common objects. Moreover, the semantics associated with **common** objects by different groups of users can vary. Abstraction mechanisms can be used to derive specialized viewpoints from generic views of data sets. Furthermore, In spatial information systems, different geometry, resolution and scale can be associated with the same object, creating multiple instances of the same entities.

Many research studies have been aimed at addressing some of these issues. One goal is how to reconcile, integrate or combine different aspects of a given object. Multiple representations definition and management can be achieved at the schema and instance levels [11].

Solutions at the schema level are in general based on spatial views, schema integration and spatial ontologies. Traditionally, in centralized database, schema integration is used to define a single representation of a system by resolving differences in the perception of real world entities by different designers [3]. View mechanisms are then used to create multiple representations or external views on the integrated schema. Inter-views relations are maintained by links such as the 00 generalizations. View mechanism have been extended to define spatial views in geographical databases [2]. Multiple representations in spatial information systems may arise from the resolution and geometry of spatial objects. For example, an accurate map of a space shows the exact locations of the objects that compose the map while a schematic map of the same space may focus only on the objects and their interconnections. Moreover, in the spatial realm, objects can be represented differently according to dimensions such geometry, time and semantics. In distributed environment where objects can be shared and exchanged, management of multiple or similar representations of objects may require defining and using spatial ontologies [1, 4]. A spatial ontology provides reference descriptions and associated semantics of the objects of a domain of interest.

At the instance level, real world objects can have multiple instances or belong to more than one class. Each instance is a different modeling dimension or aspect of the object. In traditional centralized database, the class hierarchy allows different instances. In distributed systems, instances of semantically similar objects can be in several information systems. In both cases the multiple instances of an object may be organized in hierarchical structure that reflects the links and inter-relation among the representations of the object. Multiple representations of an object can be implemented by classical role concepts. A role is an object like structure with set of properties, behaviour and semantics. An object can belong to different classes corresponding to its roles [6, 7, 9, 12]. Consequently, an object is represented by a single structure that include specific information related to its different facets or roles [6].

1.2 Objective and contribution

We present a multi-layered functional data model to support multiple representations and information sharing among

distributed spatial information systems. The key idea underlying our work is that a geographical space (example of an urban domain) can be viewed as consisting of abstract spatial functionalities that are used to derive different application classes that manipulate objects of the same geographical space. Applications can be viewed as classes of concepts that share the same abstract generic functionalities. These classes define multiple views or representation of a domain or space. For instance, in an urban management applications domain, city and highway maps, transportation systems, water supply and sewage systems applications can share the generic functionalities of a network structure defined by graph concepts and the corresponding graph traversal operations. Urban management applications that share the same functionalities must agree on a limited number of spatial concepts and operations on networks. These concepts can be specialized to take into account the details of specific applications to construct a precise interpretation of the common geographical space.

To achieve this goal, we propose a **multi-layered functional data model** consisting of generic functional layers defined over two levels: generic and domain levels. A generic functional level contains a set of primitive geometrical types and one or more generic layers. It layers provides application oriented abstract functions that are based on the primitives geometrical types. The abstract functions can be specialized and refined to derive one or more domain specific functionalities. For example, the network generic layer defines a network as a set of nodes, the relations among the nodes and the usual graph operations including graph traversal functions. The proposed model gives two main criteria for the spatial data bases: the data base consistency and the modeling of all aspects of a real object. These two criteria are an immediate consequence of keeping a structural link between different interpretations that a given real entity can have. The remainder of the paper is organized as follows. Section 2 presents the model. Section 3 is devoted to inter-domains and inter functional layer relationships. Section 4 presents an example of application. Finally, section 5 concludes the paper and discuss ongoing research.

2. MULTI LAYER FUNCTIONAL MODEL OF SPATIAL INFORMATION SYSTEMS

Figure 1 presents the functional layers of the data model. A functional layer is defined over two levels: generic functional and application. The generic functional level consists of a geometric model which provides primitive geometric types and generic layers which are defined by specialization of a subset of the primitive geometric types. Generic layers provide abstract common functionalities shared by different applications. The application level is composed of domains, each consisting of domain specific functionalities that are derived from the abstract functionalities of the generic layers. The derivation process provides class and types definitions for the abstract functions of the generic layer.

Two types of objects can be distinguished in spatial information systems: spatial and geographical objects. Spatial object are represented by geometric properties such as size or position (e.g. Point or Line). They are described by the primitives of the geometric model. Geographical objects extend the geometric properties of an object with other properties. For instance, a road object can be de-

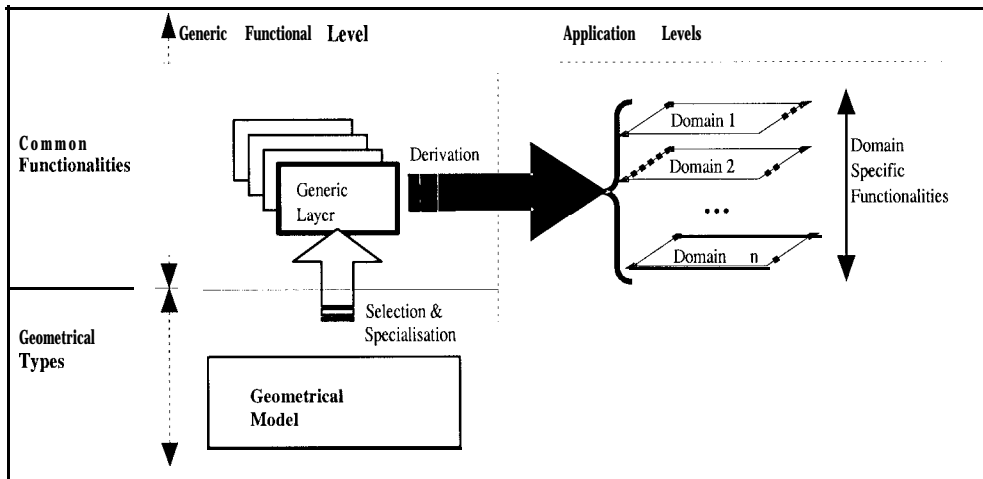


Figure 1: The functional layer.

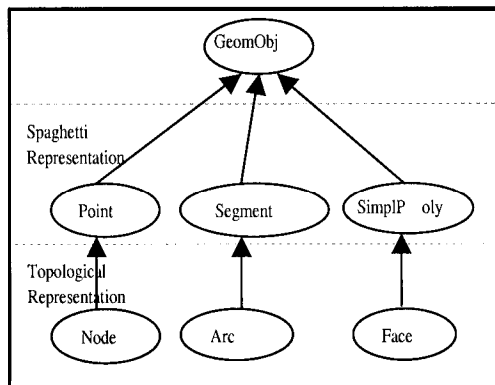


Figure 2: The functional layer.

scribed by geographical properties such as category or type of (state, interstate, departmental, etc), status, construction date, etc. Geographic objects are described by the functional and domain layers of our model. In the following sections, we give detailed description of the different components of the model.

2.1 Geometric Model

At this level, only the geometric properties of an object are described with no regard domain or application specific properties which are described by the generic layer (properties shared by the group of applications) and the domain layers (geographical properties) levels. Several geometrical models are proposed in the literature [8, 10]. They are typically based on three geometric primitives: Point, Line and Zone. Geometric primitives can be modelled in two ways: Vector or Spaghetti representation and Topological representation (Figure 2).

2.2 Functional Layer

A functional layer describes a set of applications objects

that share common abstract functionalities. It is composed of a generic layer and one or more domain layers.

2.2.1 Generic layer

A generic layer describes a set of functionalities shared by applications of some domain (e.g. network infrastructure domains). It is composed of three elements: the generic types, the generic classes and the classes of derivation.

Generic types are defined by specializing a subset of the geometric types of geometric model (see below). This allows a generic type to inherit the geometric properties of geometric types. In order to complete the definition of the generic types, some other specific properties are added. For instance, in the generic networks layer, the generic types *PointNet* and *SegmentNet*, which correspond to network point and segment respectively, are defined by specializing the geometric types: *Point* and *Segment*. A set of operations and axioms are included in the definition of a functional layer to model the characteristic features of the components of the layer. The combined types, operation and axioms are defined like abstract data types. In the definition of a layer, operations and axioms are introduced by the keywords "OP" and "AX" respectively. The generic Network layer is defined in Table 1.

Generic classes correspond to generic types. There are key elements of the derivation process. Their structures and operations are used to define the derived domain classes. Generic classes are used to define and implement objects corresponding to the abstract generic types. They provide an implementation for the structure, operations and axioms of the generic types. A generic class can have two types of associations:

- associations with other generic classes: a generic class can be linked to other generic classes belonging to different functional layers. These inter functional layer relationships are described below.
- associations with domains classes: a generic class can be linked by a derivation relationship to one or more domain classes belonging to the same functional layer.

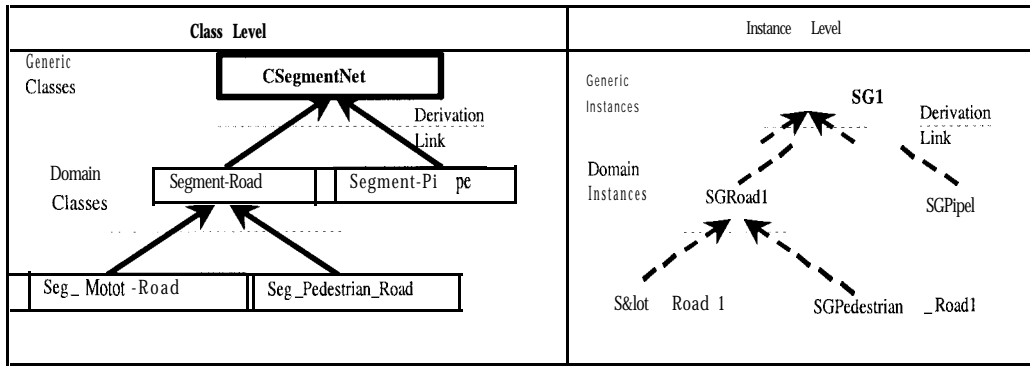


Figure 3: The functional layer.

An example of definition of the generic class SegmentNet is given in table 2.

Derivation classes: To every generic class C corresponds one derivation class C_Deriv that is essentially used to re-group all the domain classes that are derived from the generic class C and create transformation functions enabling links between different derived domain classes. A derivation class is defined by the tuple $\langle CLDerived, Fct \rangle$, where CLDerived is a set of derived domain classes, and Fct is a set of link functions between different derived domain classes. An instance from the derivation class establishes the following functions:

- Link function between domains layers: all the instances of the domain classes, which represent different interpretations of the same real object, are linked by one derivation class instance.
- Mediation function between different interpretations that a given object can have. This mediation can be reduced to a simple conversion of types (casting) or can play a more complex role.

2.3 Domain Layers

Domain layers are derived from generic layers. Each domain layer represents defining a specific application. They share the concepts and functionalities described by the generic layer. For example, one can define a generic layer and the corresponding domain layers to represent network based applications such as: highway networks, drinking water networks and cadastral management. The corresponding descriptions are derived from the generic layer elements definitions PointNet, SegmentNet or Networks discussed above. Each element of a domain layer represents one interpretation of an element of a generic layer. Therefore, one generic element can have multiple interpretations or representations. For example, the generic entity PointNet can be interpreted at the level of Road-Networks domain layer by specific the following entities Crossroad, Ttraffic circle, Toll which are different representation of a traffic node. Similarly, the generic entity SegmentNet can be interpreted by Street section, Bridge, Boulevard entities.

2.4 Instances and objects

In the context of a multiple representation databases, object creation can be realized through two levels: generic level

and domain level.

Generic level: a generic object (GO) is represented by a generic class instance. This instance is defined as an abstract or virtual structure. It can be:

- A common object which contains the common properties (in particular geometric properties) of the domain instances that are bound to this object (its different interpretations). These domain instances are grouped together in a hierarchy of which the generic object is the root.
- An object that enables a mediation function between the different interpretations that could have a generic object. This mediation can be reduced just to a type conversion (casting) or to more complex role [5].

Domain level: a domain object is an instance of a class domain. This instance is directly bound after its creation to the basic generic class instance. For example, Figure 3 presents a generic class instance CsegmentNet and an instance SG1 of CsegmentNet. This instance represents the root of a hierarchy which is composed from the instances SGRoad1 (Segment_Road class instance), SGPipe1 (Segment-Pipe class instance), SGMot_Road1 (Seg_Motor_Road class instance) and SGPedestrian_Road1 (Seg_Pedestrian_Road class instance).

To allow data exchange and sharing in interoperable information systems, objects are created as domain objects in each database, thus sharing higher functional characteristics. A mapping process is needed to create domain instance images. The creation and management of these domain instance images must be uniform even if the corresponding original objects are created and managed differently by local information systems. As in the case of multiple representations databases, the creation of the objects is achieved at the generic and domain levels. At the generic level, the generic classes are organized in the same way as the case of a multiple representations database. At the domain level, a mapping phase is achieved to create the domain instance images. The construction of domain instance hierarchies is based on these domains instance images. For example in Figure 3, the mapping phase creates for the instances SG1, SGRoad1 and SGPipe1 these instance images I_SG1, I_SGRoad1 and I_SGPipe1 through which they will be manipulated.

Table 1: Example of the generic network layer

Name: Networks	
op:	
Add-Point	$PointNet \rightarrow Networks$
Delete-Point	$PointNet \rightarrow Networks$
Add-Segment	$SegmentNet \rightarrow Networks$
Delete-Segment	$SegmentNet \rightarrow Networks$
Nb-Segment	$Networks \rightarrow Integer$
Nb-Point	$Networks \rightarrow Integer$
Successor	$PointNet \rightarrow PointNet$
Ancestor	$PointNet \rightarrow PointNet$
Distance	$PointNet \times PointNet \rightarrow float$
Path	$PointNet \times PointNet \rightarrow Network$
Optimised_Path	$PointNet \times PointNet \rightarrow Network$
RelatedPoint	$PointNet \rightarrow Setof PointNet$
Ax:	
Distance(P1, P1)	= 0
Successor(Ancestor(P1))	= P1
Ancestor(Successor(P1))	= P1
Nb_Segment(Delete_Segment(Net1))	<
Nb_Segment(Net1)	
Nb_Segment(ShorterPath(P1, P2))	< =
Nb_Segment(Way(P1, P2))	

3. INTER FUNCTIONAL RELATIONSHIPS

Two types of relationships can be defined among spatial objects: 1) a relationship between objects of different functional layers relations and 2) a relationship between generic objects and domain layers. Two types of relations are distinguished: equivalence relation and aggregation relation. The equivalence relation is used to link instances of domain classes that represent different interpretations of the same geographical object. It is defined by the tuple $r = \langle C_1, C_2, \dots, C_n \rangle$ where the C_i are generic classes or domain classes. The classes can belong to the same functional layer or to different functional layers. For example, PointNet and Coverage generic classes can be bound by an equivalence relation to express the fact that a point or a node in the traffic network domain can be represented by a Square (which is a surface or an area) in the Coverage domain layer. Similarly, a road section can be represented as a line segment in one application for road traffic and as a surface in another application for road repairs and works. This object will therefore be represented by several domain layer instances. The second type is the aggregation relation which is a particular case of the equivalence relation. It extends the 1-1 association of the equivalence relation to an 1-n association between instances. It represents the case where a real entity is globally described by one instance of a domain class and by a set of instances of another class (or several classes).

4. EXAMPLE APPLICATION

The complexity of the urban information systems resides in the number of different applications treating the same geographical data, representing and managing data differently. This section presents an urban management systems based on three functional layers: generic point, generic network and generic cover. Their definitions are as follows:

- **Functional Layer Point:** It is composed of one generic point layer and a set of domain point layers. Each domain point layer represents a thematic layer in which of which all elements are of type Point. An example of different themes is given by the distribution on the urban space, of display panels, public toilets, pollution detectors, etc. All these themes are defined from the same generic concepts who are described at generic point layer. For example, the point layer provide a generic operation that be used by the different types of points to compute the distance between two points. A generic point may have several interpretations. Each interpretation is represented by a derived instance of generic point and belongs to a point domain layer. A link is therefore maintained between the different interpretations through generic class. In the urban space for example, a point could represent at the same time, the position of a display panel, of a pollution detector and of a public toilet. All of the interpretations have the same geographical position.
- **Functional Layer Network:** It consists of generic types that describe network applications such as water mains, car-traffic network, electric or telephonic networks. It consists of the types PointNet and the SegmentNet. The abstract operations defined in this layer are classical graph manipulation operations that are used to determine a path or to compute the cost of a path between two nodes of a graph. Several domain layers are derived from the generic types. Each domain describes a precise network application. Several objects belonging to different domain layers can represent different interpretations of a same real object. For example, a car traffic node in the road network domain may correspond to a water pipe node in the water pipe networks domain.
- **Functional Layer Coverage:** It is used to describe urban space applications of type Surface such as parcels or land management, parks and green space management, or commercial and shopping centers. It defines coverage type objects. The methods attached to these objects correspond to various operations on surfaces such as the computation of perimeter or surface. Different domains are associated with the generic coverage layer.

5. CONCLUSION

In this paper, we have presented a spatial data model for allowing multiple representations spatial databases and sharing of data among heterogeneous systems. The contribution of this work is to define a data model based on multi-layered functional levels. Every functional level is defined in three layers: generic layer, geometric layer and domain layer. At the generic layer, the global spatial concepts are described in permitting the definition of a set of domain layers, which have common spatial functionalities. The specificities of every domain are described at the domain layers. In addition, we showed how links between different possible representations of the same object are maintained.

An important objective of our model is to guarantee a coherent and integral management of multiple representations in a spatial databases and information sharing in in-

Table 2: Example of the generic class SegmentNet

<p>Class SegmentNet: Segment</p> <p style="text-align: center;">Attributes:</p> <p><i>P1, P2</i> : Point /*attributes inherited from Segment <i>Weight</i> : integer /* the weight of this segment</p> <p style="text-align: center;">Methods:</p> <p><i>Extremities()</i> : (Float, Float); /*inherited methods <i>Length()</i> : Float; <i>JointSeg(SegmentNet)</i> : PointsNet; /*specific methods</p> <p><i>JointPoint(PointNet)</i> : Boolean; <i>ConnectedSegment()</i> : Setof SegmentNet ;</p> <p style="text-align: center;">Axioms:</p> <p><i>Extremities()</i> <> (0,0); <i>Length()</i> <> 0; <i>ConnectedPoint()</i> <> 0 <i>ConnectedSegment()</i> <> 0</p>
--

teroperable systems. This can be achieved on two point of views: the conceptual and the multiple representations data management. From a design point of view, the model allow a flexible design methodology that model applications in two levels of abstractions: the generic abstract definition of types and the domain level design that allows sharing and reuse of basic concepts by one or more applications. Moreover, the integration of the role concept provide a 'natural' design and description of different facets objects that can thus be dynamic and have multiple representations. From a data management point of view, there are two main advantages of the proposed model: global database consistency and integral data management. These two advantages are the immediate consequence of maintenance of a structural and semantic links allowing to regroup an object with its representations (or roles). This facilitates greatly the consistency control, the update propagation and the multiple themes query management. On-going work is on the definition of tools to support the proposed data model. This formalization will enable us to explain, with more details, the associations between different layers and then to propose a language for simple or multiple interrogation theme of the spatial databases.

6. ADDITIONAL AUTHORS

El Hassan Abdelwahed (Département Informatique, Faculté des sciences, Marrakech, Maroc, abdelwahed@ucam.ac.ma).

7. REFERENCES

- [1] D. Benslimane, E. Leclercq, M. Savonnet, M. Terrasse, and K. Yetongnon. On the definition of a generic multi-layer ontologies for describing gis based applications. *International journal of computers, environment and urban systems (CEUS)*, Elsevier Science Ltd, England, 24:191-214, 2000.
- [2] C. Claramunt and M. Mainguenaud. Spatial view: A dynamic and flexible vision of spatial database. In *Proceedings of the International Conference on* database and *expert systems applications (DEXA '95)*, pages 483-493, San Mateo, California, USA, 1995.
- [3] T. Devogele, S. Spaccapietra, and C. Parent. On spatial database integration. *International journal of geographic information systems*, 12(4):335-352, 1998.
- [4] F. Fonesca and M. Egenhofer. Ontology-driven geographic information systems. In M. C. B. Medeiros, editor, *ACM symposium on advances in geographic information systems*, pages 14-19, Kansas City, November 1999.
- [5] Y. Kambayashi and Z. Peng. Object deputy model and its applications. In *Proceedings of International Conference of Database Systems for Advanced Applications*, pages 1-15, Singapore, April 1995.
- [6] O. LI and F. Lochovsky. Adome: An dvanced object modeling environment. *IEEE Transactions on Knowledge and Data Engineering*, 10(2):255-276, March/April 1998.
- [7] M. Papazoglou and B. Kramer. A database model for dynamic objects. *VLDB Journal*, 6(2):73-96, 97.
- [8] D. Peuquet. A conceptual framework and comparison of spatial data models. *Cartographica*, 21(4):66-113, 84.
- [9] J. Richardson and P. Schwartz. Aspects: extending objectsto support multiple, independent roles. In *Proceedings of ACM SIGMOD International Conference on Management of Data*, pages 298-307, 1991.
- [10] H. Samet. *The design and analysis of spatial data structures*. Addison-Wesley, New York, 1990.
- [11] C. Vangenot, S. Spaccapietra, and C. Parent. Multi-representations and multiple resolutions in geographic databases. In *Proceedings of advanced database symposium (ADBS'99)*, Tokyo, Japan, December 1999.
- [12] R. Wieringa, W. Jonge, and P. Spruit. Roles and dynamic classes: a model logic approach. In *Proceedings of European Conference on Object-Oriented Programming (ECOOP)*, pages 32-59, 1994.