Spatial Data Integrity Constraints in Object Oriented Geographic Data Modeling

Karla A. V. Borges[1] Alberto H. F. Laender[2] Clodoveu A. Davis Jr.[1,2]

[1]Prodabel - Empresa de Informática e Informação do Município de Belo Horizonte
Av. Presidente Carlos Luz, 1275
31230-000 - Belo Horizonte - MG – BRAZIL
[karla, clodoveu]@pbh.gov.br

[2]Departamento de Ciência da Computação
Universidade Federal de Minas Gerais
Av. Presidente Antônio Carlos, 6627
31270-010 - Belo Horizonte - MG - BRAZIL
laender@dcc.ufmg.br

ABSTRACT

An important activity in the design of a particular database application consists in identifying the integrity constraints that must hold on the database, and that are used to detect and evaluate inconsistencies. It is possible to improve data quality by imposing constraints upon data entered into the database. These constraints must be identified and recorded at the database design level. However, it is clear that modeling geographic data requires models which are more specific and capable of capturing the semantics of geographic data. Within a geographic context, topological relations and other spatial relationships are fundamentally important in the definition of spatial integrity rules. This paper discusses the relationship that exists between the nature of spatial information, spatial relationships, and spatial integrity constraints, and proposes the use of OMT-G, an extension of the OMT model for geographic applications, at an early stage in the specification of integrity constraints in spatial databases. OMT-G provides adequate primitives for representing spatial data, supports spatial relationships, and allows topological, semantic and user integrity rules to be specified in the database schema.

Keywords

Geographic data modeling, conceptual modeling.

1. INTRODUCTION

A number of integrity constraints must be observed when updating a database, in order to preserve the semantics and the quality of stored data [6]. Achieving and preserving integrity of data is an established field in the database area. However, within the scope of geographic applications, special problems come up due to the locational aspects of data [14]. Most geographical information systems (GIS) use data that depend on topological relationships, and sometimes these data must be explicitly represented in the database, requiring special attention for the maintenance of the semantic integrity. Enforcing the integrity constraints must be considered one of the main implementation goals. Thus, it is convenient to explicitly specify on the geographic application schema the situations where the constraints cannot be disregarded. Many mistakes in the data entry process could be avoided, if digitizing processes based on these constraints were implemented.

Even though there is a very active research area interested in the design of robust and efficient spatial databases, it is still evident the inability of current GIS regarding the implementation and management of spatial integrity constraints [14, 17]. A modification in a spatial database may cause simultaneous updates in a large number of records in multiple files, making it hard to manage all the environment. A very sophisticated control is required to avoid redundancy and loss of integrity.

In the traditional database systems approach there is a relationship between conceptual, logical, and physical design, in which, through mapping operations, constraints that are identified in the conceptual schema are inherited and transformed into implicit constraints expressed by the data definition language (DDL) or into explicit constraints coded in the application programs [6]. This relationship must also exist in spatial information systems, so that spatial constraints can be likewise identified and implemented.

Improvement of quality is one of the key objectives of establishing integrity constraints in spatial databases [4]. It is possible to improve data quality by enforcing constraints upon data entered into database. These constraints must be identified and recorded at the database design level. However, it can be perceived that modeling geographic data requires models which are more specific and capable of capturing the semantics of geographic data, offering higher abstraction mechanisms and implementation independence [1, 2]. There are particular characteristics of geographic data that make modeling more complex than in the case of conventional applications. Within the geographic context, topologic relations and other spatial relationships are fundamentally important to the definition of spatial integrity rules. In geographic applications, topological and other spatial relationships are translated into topological integrity constraints among database entities, taking a relevant role in the data entry/updating process. "The imposition of such constraints on data entry/update is considered to have potential for the reduction of errors in data input and hence improvement in data quality [4]."

This paper addresses the relationship that exists between the nature of spatial information, spatial relationships, and spatial integrity constraints, and proposes the use of OMT-G [1], an extension of the OMT model [15] for geographic applications, at an early stage in the specification of integrity constraints in spatial databases. OMT-G provides appropriate primitives for representing spatial data, supports spatial relationships and allows the specification of spatial integrity rules (topological, semantic and user

ACM GIS '99 11/99 Kansas City, MO USA

integrity rules) through its spatial primitives and spatial relationship constructs. Being an object-oriented data model, it also allows some spatial constraints to be encapsulated as methods associated to specific georeferenced classes. Once constraints are explicitly documented in the conceptual modeling phase, and methods to enforce the spatial integrity constraints are defined, the spatial database management system and the application must implement such constraints.

This paper does not cover constraints associated to the representation of the objects, such as constraints implicit to the geometric description of a polygon. Geometric constraints are related to the implementation, and are covered here in a higher level view, considering only the shape of geographic objects. Consistency rules associated to the representation of spatial objects are discussed in [9].

This paper is organized as follows. Section 2 presents a classification of the spatial integrity constraints. Section 3 describes the OMT-G data model and its associated spatial integrity constraints. Section 4 discusses an example of use of OMT-G. Finally, Section 5 presents our conclusions.

2. SPATIAL INTEGRITY CONSTRAINTS

One important activity in the design of a schema for a particular database application consists in identifying the integrity constraints that must hold on the database. The main types of integrity constraints that occur frequently in database modeling are: domain constraints, key and relationship structural constraints, and general semantic integrity constraints [6]. Cockcroft [4] extends that classification in order to encompass the peculiarities of spatial data. This classification is based on the distinction between topological, semantic, and user rules, as follows.

Topological integrity constraints. Topology is the study of geometrical properties and spatial relations. There has been some theoretical research into the principles of formally defining spatial relationships [5]. These principles can be applied to application-specific entities and relationships to provide a basis for integrity control. Area subdivision is an example of this constraint. One city's administrative regions must be contained within the city limits, and there must not have any spot in the municipal territory that belongs to more than one administrative region or to none.

Semantic integrity constraints. These constraints are concerned with the meaning of geographic features. Semantic integrity constraints apply to database states that are valid by virtue of the properties of the objects that need to be stored. An example of this constraint is the rule that does not allow a building to be intercepted by a street segment.

User defined integrity constraints. User defined integrity constraints allow database consistency to be maintained as defined by the equivalent of "business rules" in non-spatial DBMS. This type of constraint acts, for instance, on the location of a gas station, which, for legal reasons, must lie farther than 200 meters from any existing school. The municipal permitting process must consider this limitation in its analysis. User-defined rules may be stored and enforced by an active repository.

3. THE OMT-G DATA MODEL AND SPATIAL INTEGRITY CONSTRAINTS

According to Elmasri and Navathe [6], every data model has a set of built-in constraints associated with its constructs. The OMT-G model allows several spatial integrity rules to be derived from its primitives. These rules constitute a set of constraints that must be observed in the operations that update a geographic database.

Topological integrity constraints are achieved through spatial dependence, spatial association, connectivity, and geo-fields rules. Likewise, semantic integrity constraints are achieved through spatial association and disjunction rules. User-defined integrity constraints are in turn obtained from methods that can be associated to the classes. These rules are described in the next section. The combination of the constraints supplied by the OMT model, added to the ones provided by OMT-G, grants more semantics to the application's schema.

3.1 Model overview

Starting from the primitives of the OMT object model, geographic primitives were introduced with the objective of increasing the model's semantic capabilities, thereby reducing the distance between the mental model of the space to be modeled and the usual representation model. Therefore, OMT-G provides primitives to model the geometry and topology of geographic data, providing support for "whole-part" topologic structures, network structures, multiple views of objects, and spatial relationships. OMT-G also translates topological and spatial relationships into spatial integrity constraints. Besides, the model allows for the differentiation between graphic and alphanumeric attributes, and the specification of associated methods for each class. The main strong points of the model are its graphic expressiveness and its representation capabilities, since textual annotations are replaced by the drawing of explicit relationships, representing the dynamics of the interaction between the various spatial or non-spatial objects. The OMT-G model is based on three main concepts: classes, relationships, and high-level abstractions (specialization, generalization, and aggregation), all of which incorporate spatial integrity constraints.

3.2 Classes

For geographic applications, three abstraction levels were considered: *real world level*, *conceptual/representation level* and *implementation level* [1]. The real world level contains geographic phenomena to be represented, the conceptual/representation level provides a set of formal concepts with which geographic entities can be modeled as perceived by user (at a high abstraction level), and the implementation level defines standards, storage mechanisms, and data structures to implement each representation, as defined at the conceptual/representation level. The OMT-G model works on the conceptual/representation level. Its basic classes represent the three main groups of data (continuous, discrete, and non-spatial) that can be found in geographic applications, thereby allowing for an integrated view of the modeled space. The classes can be *georeferenced* or *conventional*.

The distinction between conventional and georeferenced classes allows different applications to share non-spatial data, therefore making it easier to develop integrated applications and to reuse data [11]. A georeferenced class describes a set of objects that have spatial representation and are associated to features on Earth [2], assuming the fields and objects view as proposed by Goodchild [7]. A *conventional class* describes a set of objects with similar properties, behavior, relationships, and semantics, and which can have some sort of relationship with spatial objects, but which do not have geometric or geographic properties.

Georeferenced classes are specialized into *geo-field* and *geo-*

object classes. Geo-field classes represent objects and phenomena that are continuously distributed over the space, corresponding to variables such as soil type [2]. Geo-object classes represent individual, particular geographic objects, which can be traced back to real world elements, such as buildings. A georeferenced class is symbolized by a rectangle, subdivided in four parts (Figure 3.1). The top left-hand rectangle is used to indicate the geometry of the representation. The notation used for conventional classes corresponds to the notation used in the OMT model [16]. A simplified symbolization can be used in both cases.

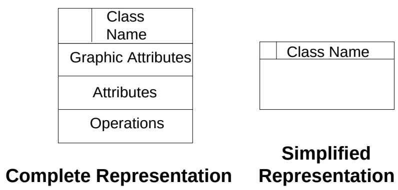

Figure 3.1 - Graphic notation for the georeferenced classes

Objects may or may not have non-spatial attributes, and can be associated to more than one geometric representation. OMT-G presents a fixed set of geometric types, using a symbolic representation that distinguishes geo-object and geo-field classes within a georeferenced class (Figures 2 and 3). Adding pictograms to the primitive element used to portray geographic classes (instead of using relationships to describe the geometry of the object) significantly simplifies the final schema.

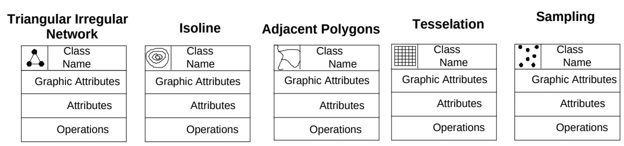

Figure 2 - Geo-fields

OMT-G has five geo-field descendant classes, *isoline*, *adjacent polygons*, *tesselation*, *sampling*, and *triangular irregular network* (Figure 2), and two geo-object descendant classes: *geo-object with geometry* and *geo-object with geometry and topology* (Figure 3). From these specializations, and from the creation of spatial aggregation primitives ("whole-part" primitives), as well as from the standardized spatial relationships, some spatial integrity rules can be deduced.

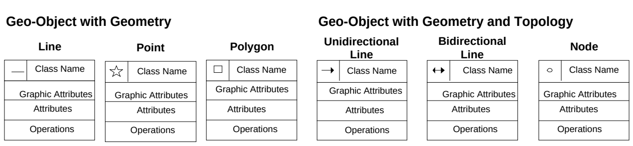

Figure 3 - Geo-objects

A *geo-object with geometry* class represents objects which have only geometric and is specialized in classes named *Point*, *Line*, and *Polygon*. A *geo-object with geometry and topology* represents objects which have, in addition to geometric properties, topological connectivity properties, and are specifically suited to the representation of spatial network structures. These properties are present in objects that are either nodes or arcs, in a graph-theoretic approach. Unidirectional lines indicate that the network has a definite flow direction, while bidirectional lines are used in the cases where the direction of the flow is deemed irrelevant. The focus here is not on the implementation of the relationship, but rather on the semantics of the connection among network elements, which is a relevant element for spatial integrity enforcing procedures. This class specializes into subclasses *Node*, *Unidirectional Line*, and *Bidirectional Line*.

From the usage of geo-field primitives, the spatial integrity rules listed in Table 1 can be derived.

Table 1 - Geo-field rules

Isoline	1. An isoline cannot intercept another isoline. 2. An isoline must be continuous.
Tesselation	3. Any point in the geographic space must belong to one and only one cell of each tesselation class.
Adjacent Polygons	4. Any point in the space can belong to one and only one instance of an adjacent polygon class. 5. Instances of this class must be completely adjacent, and there must not be any empty space between them
Triangular Irregular Network	6. Any point of the geographic space must belong to a triangle in the network. 7. There can be no superimposition among instances of this class.
Sampling	8. There cannot be superimposition among instances of the same sampling class.

3.3 Relationships

An existing problem in most data models is that the possibility of modeling the relationships between real world phenomena is often neglected [11]. Considering the importance of spatial and non-spatial relations in the understanding of the modeled space, OMT-G represents the three types of relationship that can occur between its classes: simple associations, topological network relations, and spatial relations. The discrimination of such relations has the objective of defining explicitly the type of interaction that occurs between classes. There are some applications that do not make use of spatial relations, but nevertheless there are applications on which spatial relations have a very relevant meaning, and therefore should be explicitly included in the application's schema.

3.3.1 Simple Associations, Spatial Relations, and Network Relations

Simple associations represent structural relationships between objects of different classes, conventional as well as georeferenced. *Spatial relations* represent topologic, metric, ordinal, and fuzzy relationships. Some relations can be derived automatically, from the geometry of each object, during the execution of data entry or spatial analysis operations. Topologic relations are an example of this. Others need to be specified by the user, in order to allow the system to store and maintain that information. The latter are called *explicit relations* [13].

In OMT-G, simple relations are graphically represented by continuous lines, whereas spatial relations are represented by dashed lines (Figure 4). Therefore, it is simple to distinguish between simple associations and spatial relations. Based on previous works [2, 3, 5, 10, 12], OMT-G considers a minimum set of spatial relations between georeferenced classes: *disjoint, contains, within* (contained in), *touches* (meets), *covers, covered by, superimposed to, adjacent to, near, above* (higher than), *below* (lower than), *over, under, between, coincides, crosses, traverses, in front of, to the left of,* and *to the right of*. *Contains/within* relations are treated as a special type of spatial aggregation (see Section 3.4.2). Some relationships are only allowed between specific classes, because they depend on the geometric shape. For instance, the existence of a *within* relationship assumes that one of the classes involved is a polygon. The set of concepts the user has about each real world object strongly suggests a particular representation, because there is an interdependence between the representation, the type of interpretation, and the usage given to each object class. In OMT-G this is considered in order to allow the placement of relations involving georeferenced classes.

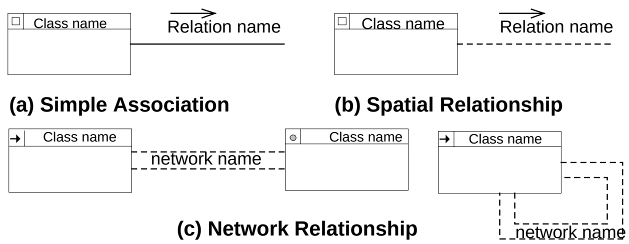

Figure 4 - Relationships

Predefining some spatial relationship names, some spatial integrity rules can be established. The OMT-G model defines the *disjunction rule*, a constraint which is applied to classes that cannot have any form of spatial relationship between themselves (Table 2). This rule is very important to maintain the integrity of the data stored in the database, and it must be used in order to check input data (see Figure 6).

Table 2 - Disjunction rule

Disjunction	9. The intersection between the geometry of objects belonging to disjoint classes must be the empty set.

OMT-G also defines *spatial association rules*, constraints that are imposed on the existence of proximity and containment spatial relations. These rules are listed in Table 3.

Table 3 - Spatial association rules

Proximity	10. The proximity relation is considered to be a *fuzzy* relationship, and therefore must receive parameters to determine its result (these parameters should vary according to the application).
Within	11. The instance that "contains" another one must always be an area object, such as a polygon or a cell.

Table 4 - Connectivity rules

Arc-node structure	12. Every node must be connected to at least one directed segment. 13. Every directed segment must be connected to two nodes. 14. Initial and final directed segments begin and end in a node.
Arc-arc structure	15. Every intermediate directed segment must be connected to two other directed segments of the same class as itself, one precedent and one successor. 16. Initial and final directed segments must be connected respectively to a successor and a precedent segment, all from the same class.

In OMT-G, *network relations* are relationships among objects that are connected with each other. Network relations are indicated by two parallel dashed lines, linking a node class to a directional line class. Network structures can be built without nodes, requiring a recursive relationship on the class which represents graph segments. The name given to the network is annotated between the two dashed lines (Figure 4c). The *connectivity rules*, which apply to network relationship primitives, are listed in Table 4.

The system is required to ensure the connection between all types of nodes and segments. Network relations can be maintained by the GIS using special data structures, and are represented by connecting arcs and nodes. Connectivity rules are usually enforced by the GIS itself.

3.3.2 Cardinality

Structural constraints on relationships are used to specify restrictions that limit the possible combinations of objects that may participate in a relationship [6]. The notation for cardinality ratio and participation constraints adopted by OMT-G is the same used by the Unified Modeling Language (UML) [15], and has been chosen because it has a greater expressiveness than the notation proposed by the original OMT object model.

3.4 High-level abstractions

In the OMT-G model, the generalization and specialization abstractions apply both to georeferenced classes and conventional classes, following the definitions and notation proposed in the OMT object model.

Aggregation is a special form of association between objects, where one of them is considered to be assembled from others. The graphic notation used in OMT-G follows the one used by the OMT model. An aggregation can occur between conventional classes, between georeferenced classes, and also between georeferenced and conventional classes. When the aggregation is between georeferenced classes, spatial aggregation must be used.

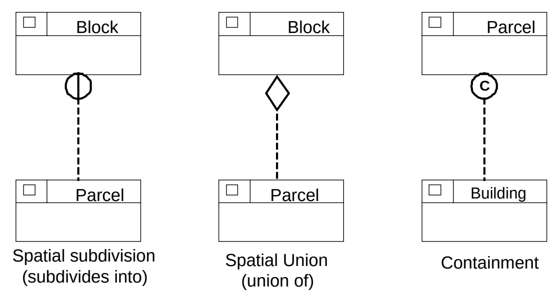

Figure 5 - Spatial aggregation - primitives and examples

Spatial aggregation is a special case of aggregation in which topological "whole-part" relationships are made explicit [8]. The usage of this kind of aggregation imposes spatial integrity constraints as to the existence of the aggregated object and the corresponding sub-objects. Beyond providing more clarity and expressiveness to the model, the observation of these rules contributes to the maintenance of the semantic integrity of the geographic database. Spatial aggregation, also called topological "whole-part", has been subdivided into *spatial subdivision*, *spatial union*, and *containment*. These structures have, as a common property, the not-null intersection of the geometry of each part with the geometry of the whole. The notation of the three structures is presented in Figure 5. In the spatial subdivision structure, the whole is divided into parts of the same geometric nature, and the geometry of the whole is fully covered by the geometry of the parts. The spatial union structure is the opposite of the spatial subdivision. The whole is formed from the union of the parts. The difference between them is in the origin of the geometry of the whole. In the containment structure, the geometry of the whole contains the geometry of the parts. Objects with different geometric nature can be contained in the whole. Examples are presented in Figure 5.

Table 5 - Spatial dependence rules

Spatial subdivision	17. The primitive object must originate at least two derived objects 18. Any portion of space contained within the primitive object must contain one and only one derived object. Area superimposition of empty spaces are not allowed. 19. In the case of polygons, the geographic boundaries of the derived objects must be completely contained in the primitive object's geographic boundaries. Partial coincidence with the boundaries is allowed, but exceeding them is not. 20. Modifications in the geographic boundaries of the primitive object force the modification of the boundaries of one or more derived objects. 21. Modifications in the boundary of one of the derived objects implies the modification of the boundary of some other derived object, so as not to allow any empty spaces within the primitive object. 22. Elimination of a primitive object forces the elimination of all derived objects.
Spatial union	23. The origin of an aggregate object depends on the union of at least two disjoint objects that belong to the same primitive class. 24. The boundaries of the aggregate object must coincide with the boundaries of the geometric union of the boundaries of the primitive objects, and must never exceed them. 25. Modifying the boundaries of the aggregate object can only be achieved through modifications in the boundaries of the primitive objects. 26. Eliminating one of the primitive objects implies on modifications on the boundaries of the aggregate object. 27. Eliminating all of the primitive objects that have originated the derived object will determine the elimination of the aggregate object.
Containment	28. The geometry of the containing object must encompass the geometry of the contained objects. 29. The boundaries of no contained object can exceed the boundaries of the containing object.

The concepts of primitive and derived classes, and of primitive and derived objects, will be used to describe the spatial integrity rules associated to the spatial aggregation primitives. A primitive class is a class which will originate other classes. A primitive object is an instance of a primitive class. A derived object is an instance of a derived class, and it originates from a primitive object. The corresponding *spatial dependence* and *containment* rules can be specified next. Spatial dependence rules are constraints that are imposed by the existence of aggregate objects, in which the nature of the aggregate object depends on the graphical existence of sub-objects, and vice-versa. These rules correspond to the spatial primitives *spatial subdivision*, *spatial union*, and *containment* (Table 5). Containment rules impose constraints on the existence of objects contained within another object's geometry.

There is an important correspondence between the spatial aggregation and the consistency of thematic attributes of the respective sub-entities. Attributes such as area and population need integrity control at the geometric-topological level to avoid distortion [14].

OMT-G also includes a *cartographic generalization* primitive, used to record user views. Since there is no spatial integrity constraint associated to this primitive, it will not be described here.

4. DISCUSSION OF AN EXAMPLE

In order to illustrate the spatial integrity constraints derived from the primitives and spatial relationships included in OMT-G, a sample model is presented in this section, corresponding to part of an urban cadastral database. The geographic space corresponds to a municipality. This space contain blocks, which are in turn subdivided into parcels. Each parcel is represented by its polygonal boundary. Parcels can be unoccupied or built, depending on whether one or more buildings have been erected on it. Building addresses are formed by concatenating the thoroughfare code to the street number. Each address is defined as a symbol, and is to be located inside the building's surface. Only built parcels can have addresses. A thoroughfare is represented by its segments, thereby composing the arcs in a street network. The nodes are thoroughfare intersections, at the crossings. Street segments are classified according to their dimension and to the intensity of traffic in four categories: local, collector, arterial, and regional link. Relief is represented by contour lines, that cover the whole municipal territory.

Figure 6 shows part of the simplified schema for the urban cadastral database. Usage of geo-fields rules can be identified in the `Contour Line` class, represented as a set of isolines. In the creation and maintenance of each `Contour Line` instance, it must be ensured that each one cannot intercept another `Contour Line` (rule 1) and each `Contour Line` must be continuous, never allowing, for example, breaks in the line for the insertion of text (rule 2).

The `Municipal Boundaries`, `Block`, `Parcel`, and `Building` classes are represented by polygonal objects. Spatial aggregation primitives have been used in the relationship between `Municipal Boundaries` and `Block`, between `Block` and `Parcel`, and between `Built Parcel` and `Building` classes. As an example of the use of spatial dependence rules, observe the `Block` class, which is subdivided into the `Parcel` class. The `Block` instance must exist first. In the creation and maintenance of each `Parcel` instance, it must be ensured that each one is contained in only one `Block` instance. Subdividing a block generates at least two parcels (rule 17), or else the block *coincides with* a parcel. This means that the block and the parcel instances have the same size and exist in the same location. In the case of a subdivision, each parcel must be adjacent to the other, ensuring no overlapping and no empty spaces (rule 18). The parcels' boundaries must be entirely contained in the block's boundary, possibly coinciding in part with it but not exceeding it (rule 19). If the block's boundary is changed, increasing or decreasing its area, parcels contained within it will be affected, and it must be decided which ones will be modified (rule 20). If a parcel's limits are modified, some or all of the adjacent parcels must be modified as well (rule 21). Eliminating a block implies the elimination of all parcels within it (rule 22).

As an example of containment rules, consider the `Parcel` class (`Built Parcel` subclass), which contains the `Building` class. When the boundaries of a `Built Parcel` object are created, no `Building` instances must be crossed, since any building's boundary must be contained within a single parcel (rule 28). No buildings outside of parcels are allowed, and no buildings can belong to more than one parcel (rule 29).

The disjunction rule is applied to the spatial relationship named *Disjoint,* between `Building` and `Street Segment` classes.

This means that there can never be any street segment overlapping a building (rule 9). If it becomes necessary to draw a street segment over a building, the building must first be deleted. The street segment and building creation routines should enforce this rule.

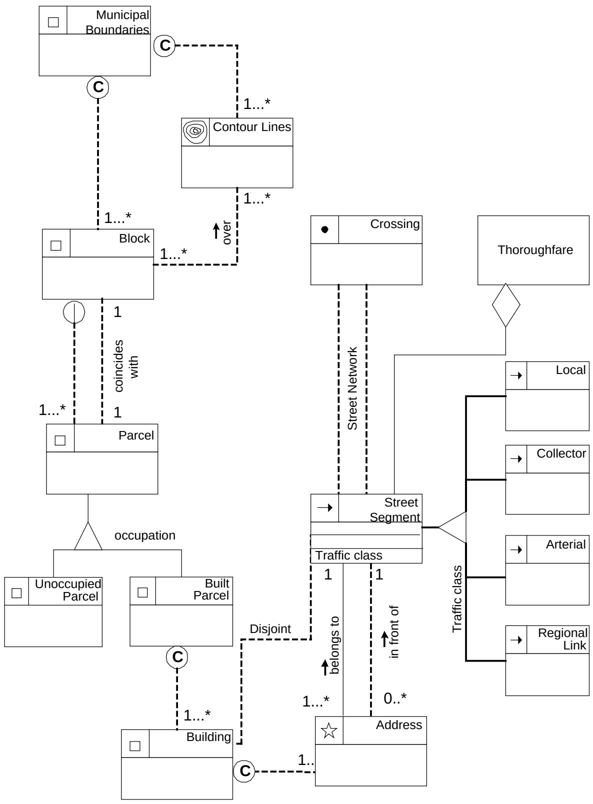

Figure 6 – Partial Schema of the Urban Cadastral Database

Street segments are classified according to their dimension and intensity of traffic into four categories: local, collector, arterial, and regional link (user-defined constraints). The execution of this rule (`Traffic Class` method) can produce a different viewing color for each object, according to the classification. The `Crossing` class (node geo-object) represents nodes in the street network. The street network is represented by the topologic network relationship primitive, and therefore the cardinality and the spatial constraint *connect to* are implicit (rules 12, 13, and 14). In the network creation process, the nodes must exist before the arcs are created. Structural constraints are used in each relationship.

Finally, it should be noticed that all rules discussed above are implicitly derived from the semantics of the OMT-G constructs and need not be described elsewhere.

5. CONCLUSION

This paper has examined the issue of data integrity in geographic information systems, focusing on the relationship that exists among the nature spatial information, spatial relationships, and spatial integrity constraints. The importance of identifying such integrity constraints at the conceptual level has been shown, and the use of OMT-G, an extension of the OMT model for geographic applications, proposed for the specification of integrity constraints in spatial databases. OMT-G provides adequate primitives for representing spatial data, supports spatial relationships, and allows topological, semantic and user integrity rules to be specified in the database schema. Being an object-oriented data model, it also allows some spatial constraints to be encapsulated as methods associated to specific georeferenced classes. A full implementation of the integrity constraints presented here ensures consistency through data entry and update. OMT-G has been compared with other models [1] and has proved in practice to be capable of representing the particular aspects of geographic data, making the modeling of geographic applications easier. Notice that the schema in Figure 6 could hardly be kept as compact and expressive as presented if other geographic data models were used instead of OMT-G.

6. REFERENCES

[1] Borges, K. A. V. *Geographic data modeling – an extension of the OMT model for geographic applications.* Master's thesis, João Pinheiro Foundation, 1997 (in Portuguese).

[2] Câmara, G. *Models, languages, and architectures for geographic databases*. Ph.D. Thesis, INPE, 1995 (in Portuguese).

[3] Clementini, E., Felice P., Oosterom, P. A small set of formal topological relationships suitable for end-user interaction. In *Proc. 3rd Symp.on Spatial Database Systems* 277-295, 1993.

[4] Cockcroft, S. A taxonomy of spatial data integrity constraints. *GeoInformatica* 1(4): 327-343, 1997.

[5] Egenhofer, M. J., Franzosa, R. D. Point-set topological spatial relations. *Intl. J. GIS* 5(2):161-174, 1991.

[6] Elmasri, R., Navathe, S. *Fundamentals of database systems*. 2nd Edition. Addison-Wesley, 1994.

[7] Goodchild, M. F. Geographical data modeling. *Computers & Geosciences* 18(4):401-408, 1992.

[8] Kösters, G., Pagel, B., Six, H. GIS-application development with GeoOOA. *Intl. J. GIS* 11(4):307-335, 1997.

[9] Laurini, R., Thompson, D. *Fundamentals of spatial information systems*. Academic Press, 1992.

[10] Mark, D. M., Egenhofer M. J., Shariff, A. R. M. Towards a standard for spatial relations in SDTS and geographic information systems. In *Proc. GIS/LIS'95* 686-695, 1995.

[11] Oliveira, J. L., Pires, F., Medeiros, C. M. B. An environment for modeling and design of geographic applications. *GeoInformatica* 1(1):29-58, 1997.

[12] Papadias, D., Theodoridis, Y. Spatial relations, minimum bounding rectangles, and spatial data structures. *Intl. J. GIS* 11(2):111-138, 1997.

[13] Peuquet, D. J. A conceptual framework and comparison of spatial data models. Cartographica 21:66-113, 1984.

[14] Plumber, L., Groger, G. Achieving integrity in geographic information systems: maps and nested maps, *GeoInformatica* 1(4): 346-367, 1997.

[15] Rational Software Corporation. The Unified Language: notation guide, v. 1.1 July 1997. (http://www.rational.com).

[16] Rumbaugh, J., Blaha, M., Premerlani, W., Eddy, F., Lorensen, W. O*bject-oriented modeling and design.* Prentice-Hall, 1991.

[17] Worboys, M. F. A unified model for spatial and temporal information. *The Computer Journal* 37(1):26-34, 1994.