A taxonomy of spatial data integrity constraints

SOPHIE COCKCROFT

Department of Information Science,
University of Otago,
PO Box 56,
Dunedin, New Zealand.

Telephone: +64 3 479 8090
Fax: +64 3 479 8311
Email: Scockcroft@commerce.otago.ac.nz

Keywords: Database Constraints, Spatial data quality, System development, Rules

Abstract

Spatial data quality has become an issue of increasing concern to researchers and practitioners in the field of Spatial Information Systems (SIS)[1]. Clearly the results of any spatial analysis are only as good as the data on which it is based. There are a number of significant areas for data quality research in SIS. These include topological consistency; consistency between spatial and attribute data; and consistency between spatial objects' representation and their true representation on the ground. The last category may be subdivided into spatial accuracy and attribute accuracy. One approach to improving data quality is the imposition of constraints upon data entered into the database. This paper presents a taxonomy of integrity constraints as they apply to spatial database systems. Taking a cross disciplinary approach it aims to clarify some of the terms used in the database and SIS fields for data integrity management. An overview of spatial data quality concerns is given and each type of constraint is assessed regarding its approach to addressing these concerns. Some indication of an implementation method is also given for each.

[1] The term spatial information system (SIS) is used here in preference to geographic information system (GIS) because many of the principles described here apply to a wider spatial context than just the geographic.

1. Introduction

Most database applications have certain integrity constraints that must hold with respect to the data. The simplest example of this is specifying that a data item must be of a certain data type. There are more complex integrity constraints that, for example, govern the relationships between database records. A full discussion is given in section 4. The major challenges for data quality management in spatial databases fall into two categories: first, ensuring positional and attribute error are minimised; and second, ensuring the logical completeness of the data. Adopting a more rigorous approach to integrity constraint management can reduce these errors.

The users of most spatial data sets have no idea of the accuracy of the data contained within them. They base their subsequent analysis using the datasets on the assumption that the data is error free or that errors are kept to an 'acceptable' level (Marble, 1990). This raises the question of whether any level of error is acceptable and if so, on what criteria would such a level be based.

One approach to improving data quality is the imposition of constraints upon data entry. This paper presents a taxonomy of integrity constraints as they apply to spatial data management. The difference between a constraint and a rule is discussed and put in the context of computer science and spatial information systems (SIS) literature. The implementation of such rules/constraints is also discussed.

The aim of presenting such a taxonomy is to clarify the terms used in spatial data management and to identify areas of commonality between computer science and SIS disciplines. This will lead to the identification of appropriate implementation strategies. It will be seen that different integrity constraints lend themselves to different implementation approaches. It is also apparent that *violations* of constraints have different levels of severity depending on the constraint in question. This has two implications; the first is that differing types of warning may have to be issued on data entry; the second implication is that, when setting up the spatial information system, it will be possible to identify a system of authorisation for personnel using the system. That is, a level of security appropriate to the type of rule to be enforced.

The following section expands on the issue of spatial data quality and in particular the need for reporting, which is discussed in section 2.2. Section 3 discusses the distinction between constraints and rules and identifies the many manifestations of the two in computer science and spatial information systems literature. In section 3.2.1, *rules* as opposed to constraints, within the taxonomy are discussed in particular with respect to constraint and deductive databases. This sets the scene for a full discussion of integrity constraints as they apply to SIS and the presentation of the taxonomy. Section 4 presents the existing understanding of static integrity constraints in database systems. In section 5 the definition of what would traditionally be referred to as business rules are extended into topological, semantic and user rules to reflect the nature of spatial data. In section 6 the concepts of the preceding sections are combined to give a full taxonomy. The implications of this taxonomy for implementation and management of SIS are discussed.

The importance of reporting data quality, which was introduced in section 2, is given a practical treatment in this section where the application of the taxonomy with regard to reporting is examined. Knowledge representation and how it applies to classes of the taxonomy is discussed in section 7. In conclusion a suggestion is made for an alternative approach to the implementation and management of spatial data integrity constraints.

2. Spatial Data Quality

In this section a review of data quality issues is given. Improvement of data quality is one of the key objectives of establishing integrity constraints in spatial databases.

2.1. Correctness and Accuracy

Correctness concerns the consistency between, and completeness of, the data and the original source about which the data are collected. A thorough discussion of consistency with respect to spatial data was given in (Laurini & Milleret-Raffort, 1991). Accuracy has several components including accuracy of attribute values, spatial and temporal references. Also of relevance in data quality is how observations are taken, measurements made and input into the computer, how data are processed and how results are presented. These are represented in the bottom section of Figure 1.

Positional and attribute error result in either the coordinates associated with a feature, or the characteristics/qualities of the feature, being wrongly described. Hunter (Hunter, 1996) gave examples of two further forms of error (see Figure 1), which he referred to as secondary forms as follows; Logical inconsistency, for example the failure of road centrelines to mathematically meet at intersections, and completeness, for example the removal of soil polygons from a soils data set which have an area less than a certain minimum threshold. Hunter and Beard (Hunter & Beard, 1992) put forward the classification illustrated in Figure 1 for error in Geographical Information Systems. Whilst positional and attribute error are often discussed together, there are compelling reasons for dealing with them separately (Collins & Smith, 1994). The main reason is that positional accuracy can be quantified as some true value, and error models are emerging for this purpose. Attribute accuracy on the other hand is qualitative in nature. That is, the *wrongness* of an attribute's description cannot be quantified. The implementation of integrity constraints has the potential to improve attribute accuracy, however, but it is unlikely to affect positional accuracy because the source of this error is based more on measurement than knowledge. A more mundane problem, which has historically been faced in spatial databases, is the maintaining of the currency of linkages between spatial and non-spatial data. Modern SIS have largely addressed this problem.

2.2. Reporting of data quality

Regardless of the nature of error there are some other more general quality issues which were discussed in (Hunter, 1996) at some length. They include protecting the reputation of the data provider, minimising the exposure to risk of litigation and reducing the likelihood of product misuse through quality reporting. On the last point Hunter coined the phrase '*there is really no such thing as bad data just inappropriate data*' (Hunter, 1996: page 96). An example was given about the use of a data set with inaccurate road centre line data. This would be a severe error on the part of a utility manager who wanted to exactly pinpoint the location of water mains but insignificant for a marketing manager wanting to identify target addresses along the road in question. It is now becoming more common for data providers to furnish their clients with *metadata,* that is data about data, on quality, lineage and age. Age, appropriateness and cost of the data are relevant to the quality (Worboys, 1995). With regard to cost, it has been posited that when data are distributed free of charge there is usually a tacit understanding that 'you only get what you pay for' (Hunter, 1996: page 98). Appropriateness of data is of course the determining factor in the search for acceptable levels of error referred to in section 1.

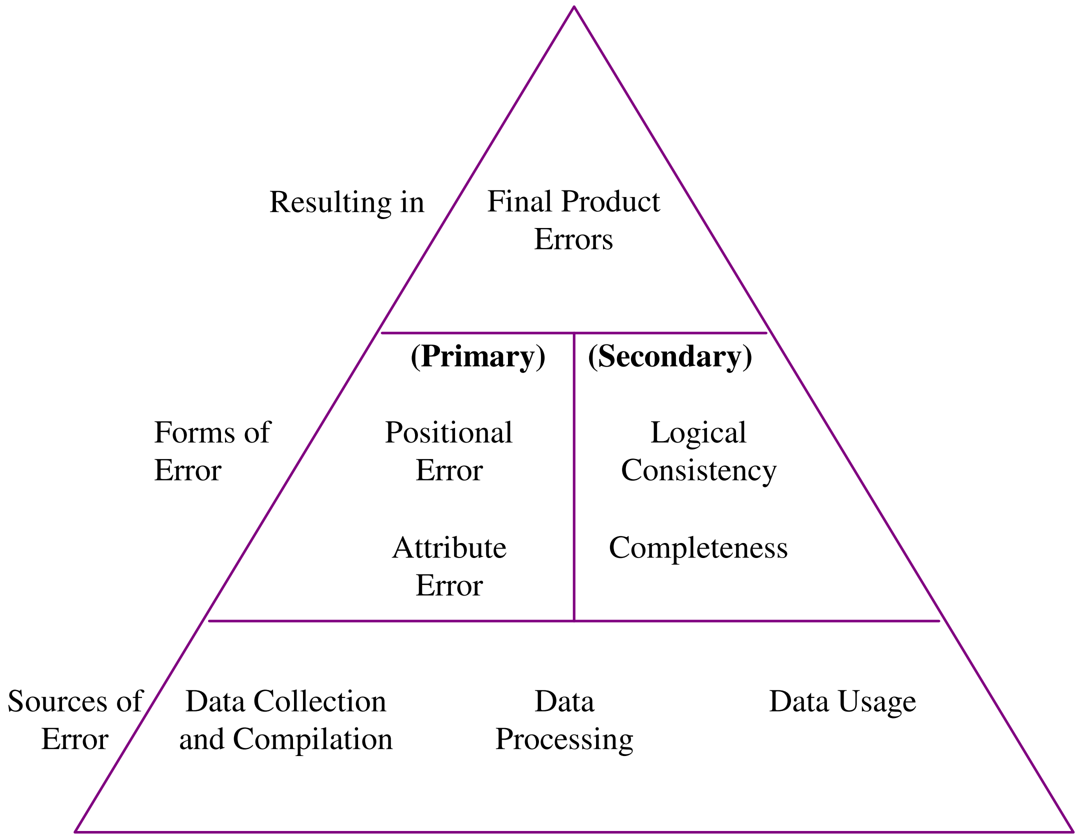

Figure 1 A classification of error in Geographic Information Systems (Hunter, et al., 1992)

2.3. Sources of error

The problem of errors in the final product illustrated in the top section of Figure 1 was expressed in the forward to a recent conference (Congalton, 1994: page 3)

"Funds for developing digital natural resource data bases are often meagre, and/or hard to justify. Corners are cut; little attention is paid to quality. It may be shocking to some, but themes are often digitised directly on unrectified aerial photographs, or ragged and creased paper map sheets that noone has any idea of how they were produced or where they came from [sic]. I wish I could say this situation was the exception, but it is closer to the rule"

Causes of these errors, hinted at in the above quote, were originally catalogued by Aronoff (Aronoff, 1989). Collins et al (Collins, et al., 1994) presented them in the form shown in Table 1. The use of unrectified or bad quality maps is of concern at the stage when data is being prepared for input. It is also related to data collection. The lack of supporting information, or metadata, for data sets has implications for the use of results in the final row of Table 1. There are also implications for data manipulation if topological integrity is not maintained. There has been some work on checking the consistency of spatial data already entered to a database as well as at data entry (Laurini, et al., 1991; Ubeda & Servigne, 1996). There has also been some work on improving the results of queries through the imposition of spatial integrity constraints (Egenhofer, 1994). The second row, concerning data input, is of most relevance to the work presented here because it is at this point that database constraints, to ensure the integrity of attribute data, can be imposed.

STAGE	*SOURCES OF ERROR*
Data collection	• Inaccuracies in field measurements • Inaccurate equipment • Incorrect recording procedures • Errors in analysis of remotely sensed data
Data input	• Digitising error • Nature of fuzzy natural boundaries • Other forms of data entry
Data storage	• Numerical precision • Spatial precision (in raster systems)
Data manipulation	• Wrong class intervals • Boundary errors • Spurious polygons and error propagation with overlay operations
Data output	• Scaling • Inaccurate output device
Use of results	• Incorrect understanding of information • Incorrect use of data

Table 1 Separation of error into time phases (Collins, et al., 1994)

3. Constraints and Rules

The term 'user defined spatial integrity constraint' is adopted here to describe the type of database constraint that is set up in response to a rule, defined by the user of the system, relating to the way they wish the database to respond to a given event. It may be specific to a given application. A non-spatial example would be 'A pay rise cannot have a negative value'. The term *business rule*, which is used for this type of constraints in mainstream database systems, will be abandoned for the purposes of this work. There are two reasons for this; first the word 'business' does not universally apply to spatial systems. In fact only a subset of the type of rules described will have anything to do with business or the rules by which business is run. This is contrary to the use of the term 'business rule' in mainstream database technology. Second, there is a common understanding of what is meant by business rule in data centred commercial applications and the term can be applied uniformly across such applications. This is not the case for the spatial equivalent. Thus the term user defined integrity constraints will be adopted. A classification of integrity constraints as they are understood in the non-spatial sense is given in section 4. The classification needs to be enhanced in certain areas in order to serve the purpose of spatial systems. This is because the nature of the relationships on which the constraints are defined is different. The enhancement of this classification is one of the aims of this paper.

3.1. User defined spatial integrity constraints

Central to the definition of user defined spatial integrity constraints put forward in this paper is the idea that spatial integrity constraints may be defined in terms of attribute data. This issue was alluded to in the review paper by Gunther et al (Günther & Lamberts, 1994: Page 17).

"..GIS usually do not offer any functionality to preserve semantic integrity. For example, it is not possible for a user to specify that a value must be included in a particular value range or that it is valid only in connection with certain other values. In DBMS, on the other hand, consistency according to user-defined semantic constraints can often be maintained as well."

An example of defining spatial integrity constraints in terms of attribute data was given in (Chadwick, 1995: page 1050) when referring to an SIS application for a pipe network:

" A butterfly valve can only be connected to a pipe > 14 inches in diameter"

Clearly, although this rule will have to be implemented in both spatial and non-spatial components of an SIS, it is based on attribute data from the non-spatial component. The word *connect* has implications for the spatial component of the system since it implies a topological relationship. Point set topological relationships have been formally defined in (Egenhofer & Franzosa, 1991). These are used as a basis for describing the spatial relationships upon which the user rules would be defined. However, the basis on which the rule, in the example above, would be checked is the value of the attribute “diameter”.

3.2. Other rules in spatial systems

Rules in the spatial information systems literature more frequently refer to expert system rules (Luo & Jones, 1995) (Jones & Luo, 1994). Recent advances in the area of logic programming, expert systems and database have resulted in the emergence of constraint databases (Kanellakis, 1995). For completeness a brief discussion of deductive and constraint databases is given here.

3.2.1. Expert systems, Deductive and Constraint databases

Deductive databases provide a means of defining the structure and semantics of complex objects in a declarative formalism based on predicate calculus. A deductive database is comprised of two parts the extensional database (EDB) which is similar to a classical relational database and the intensional database (IDB) which contains reasoning rules combined with deductive processing. Through the use of deduction it is possible to translate between a user’s specification of a query and the contents of the extensional database (Jones, et al., 1994) (Luo, et al., 1995). Static integrity constraints can be specified by the user in a first order logic language and translated into production rules to be stored in the IDB. This approach is exemplified by active database systems developed by Medeiros et al. (Medeiros & Andrade, 1994; Medeiros & Cilia, 1995; Medeiros & Magalhaes, 1993). Standard relational databases cannot encapsulate data and procedures that are specialised for particular types of geographical phenomena. Specialised data structures must be adapted to fit the tabular relational structure. The use of logic programming for implementing some object-oriented concepts in geographical data was illustrated by Egenhofer and Frank (Egenhofer & Frank, 1990). Smith (Smith, Ramakrishnan & Voisard, 1992) has emphasised the scope for deductive databases to assist in the integration of environmental modelling procedures within SIS.

Constraint Databases combine concepts from database theory and logic or constraint programming. The framework for such databases was first introduced in a paper by Kanellakis, Kuper and Revesz (Kanellakis, 1995) and has since become a very active area of database research. In constraint databases, the notion of a tuple in a relational database is replaced by a conjunction of constraints from an appropriate language - for example order constraints, or linear arithmetic constraints. Such a tuple can be seen as representing a large, possibly even infinite, set of points in a compact way. Thus, a benefit for spatial databases is immediately apparent.

4. Traditional Database Management Systems: The Incorporation of Integrity Constraints

The specification of integrity constraints is part of the database design process. Some constraints can be specified within the database schema and automatically enforced.

Others have to be checked by update programs or at data entry. These functions are expressed in a constraint language in the application. Figure 2 presents one view of integrity constraints in traditional DBMS. It shows how they relate to the *data model*, the *data definition language* and the *application* that uses the database. The constraints applicable at each of these levels of spatial information systems development are *inherent* constraints, *implicit* constraints and *explicit* constraints respectively. Definitions of these constraints according to Elmasri and Navathe (Elmasri & Navathe, 1994) follow. The distinction between conceptual and logical data models was not made.

Inherent constraints are inherent to the data model itself and do not need to be specified in the schema but are assumed to hold by the definition of the model constructs. For example an inherent constraint in the formal relational model is that an attribute value is not divisible ie it is atomic. An inherent constraint in the Entity Relationship (ER) model is that every instance of an n-ary relationship type R relates to exactly one entity from each entity type participating in R in a specific role. Implicit constraints are specified using the Data Definition Language (DDL). Explicit constraints are any constraint beyond the scope of the DDL. They are managed by the application programs associated with the database.

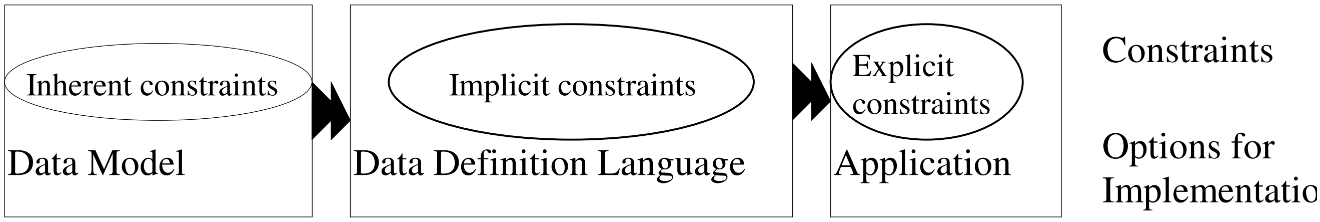

Figure 2 Classes of integrity constraint in terms of where they are applied

4.1. Database state and integrity

Fahrner et al (Fahrner, Marx & Philippi, 1995) identified a further type known as dynamic constraints as described below. Elmsari et al (Elmasri, et al., 1994) noted that integrity could also refer to transaction integrity, which governs such factors as concurrency control and recovery techniques. Other constraints can be subdivided into state or *static* constraints and *transition* constraints (Date, 1990; Elmasri, et al., 1994).

Static constraints - these must be satisfied at every single state of the database. They express which database states are correct and which are not. It could be the fact that a salary cannot be negative, or a manager's salary must always be greater than any employee's is within the same department.

Transition constraints - these restrict the possible transitions from one database state to another. A user may want to specify that on updating a salary database, salaries should not decrease.

Dynamic constraints (Fahrner, et al., 1995) - these restrict the possible sequence of state transitions of the database. Thus an employee who is made redundant could be restricted from *then* getting a pay rise. These constraints have not been defined elsewhere. It seems that they have some overlap with transaction integrity mentioned earlier in this section. Figure 3 gives a further breakdown of static constraints that are described in section 4.3.

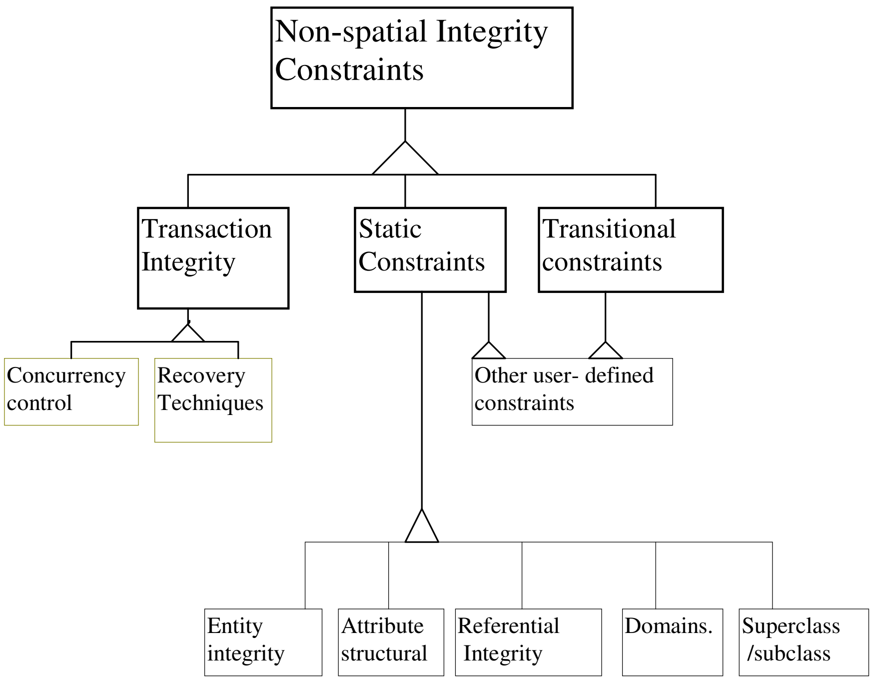

Figure 3 Classes of traditional integrity constraints

4.2. Database State and Implementation options

It has been seen, in Figure 2 and in section 4.1 concerning database state and integrity, that constraints can be classified both in terms of database state and in terms of where they are applied. These two classifications are not mutually exclusive but it is hard to draw direct comparisons between them. Inherent constraints are static by nature. Implicit constraints are usually static except certain transition constraints can be expressed by triggers in the DDL. Explicit constraints are usually transitional except complex static situations that cannot be described in the DDL e.g. functional dependencies.

4.3. Static, implicit integrity constraints in mainstream database development

The majority of traditional database constraints are types of static constraint. Since they can be specified and represented in database schemas it should follow that they are also implicit. The following examples are indeed implicit. However, rules involving multiple tables (or classes in object oriented terminology) cannot be specified in the DDL.

Domains - constraints on valid values for attributes. The attribute must be drawn from a specified domain.

Entity integrity rule - each instance of an entity type must have a unique identifier or primary key value that is not null. The implication here is that if you cannot uniquely identify a real world object, then it does not exist.

Attribute structural constraints - whether an attribute is single valued or multivalued and whether or not 'null' is allowed for the attribute.

Referential Integrity constraints - a database must not contain any unmatched foreign key values. Foreign key values represent entity references. So if a foreign key A references a primary key B then the entity that B uniquely identifies must exist.

Superclass / Subclass constraints - specify disjointedness or totality of specialisations or generalisations. These are specified using a predicate condition. For example, in an employees database all members of the secretary subclass must satisfy the defining predicate jobtype = secretary.

4.4. The need for more well defined/extended integrity constraints

The remaining classification of 'other user-defined constraints' in Figure 3 corresponds to the static aspects of user defined integrity constraints, which will be discussed in section 5.3. These constraints are classified as explicit and must be implemented either procedurally in the transactions of the database eg update, or using a constraint specification language or triggers. Detailed discussion of these implementation methods is beyond the scope of this paper. The reader is referred to Elmsari et al (Elmasri, et al., 1994). In the object oriented approach some spatial integrity constraints can be encapsulated as private methods and thus constraints deriving from computational geometry can be defined alongside others (Laurini, et al., 1991).

Assuming that the relational model or an extension thereof, is used to implement the spatial database, these constraints are all relevant. In 1990 Date (Date, 1990: page 186) noted that when relational products came to the market in the early 1980s *"..the emphasis was primarily on performance and other physical matters, not on logical issues such as integrity*". He went on to say that, although this was no longer the case, support for integrity in commercial products was still a long way from perfect particularly in the area of transition rules and support for domains. The need to extend traditional Database

Management Systems in order to incorporate semantic information was also identified by Stonebraker et al (Stonebraker & Kemnitz, 1991: page 85)

"It is clear to us that all DBMSs need a rules system. Current commercial systems are required to support referential integrity, which is merely a simple-minded collection of rules. However, there are a large number of more general rules which an application designer would want to support"

This applies to spatial data in particular, to the extent that the classification given in Figure 3 is considered incomplete. Necessary adaptations are discussed in sections 5 and 6.

5. User defined spatial integrity constraints which have their origin at the system design stage

In the remaining sections of the paper attention is turned to the spatial aspects of data integrity. The second classification is based on the distinction between topological, semantic and user rules. This classification is illustrated in Figure 4

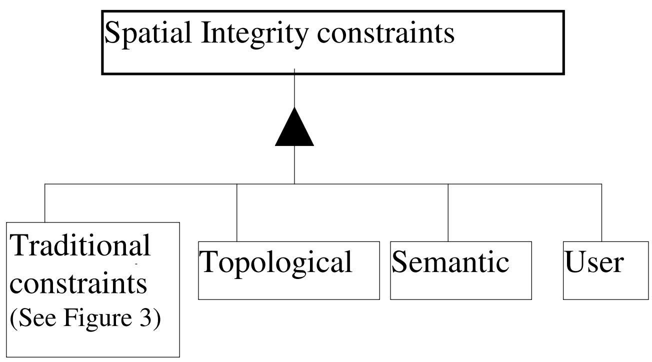

Figure 4 Classification hierarchy for spatial integrity constraints

5.1. Topological integrity constraints

Topology is the mathematical procedure for defining spatial relationships between points lines and polygons. There has been some theoretical research into the principles of formally defining these relationships (Egenhofer, et al., 1991). The issue of defining topological integrity constraints in databases according to these principles has also been investigated (Hadzilacos & Tryfona, 1992). These principles can be applied to application specific entities and relationships to give a basis for integrity control. In (Hadzilacos, et al., 1992) they were investigated in general terms as object class definitions, queries or integrity constraints. Techniques for the enforcement of integrity constraints were not elucidated. At the design level, it has been shown (Firns, 1994) that spatial integrity constraints could be implemented in a relational database. Hadzilacos et al (Hadzilacos, et al., 1992) stated that defining topological constraints in terms of absolute positions would be straightforward but cumbersome, and totally impractical due to the performance overheads incurred in implementing them in such a fashion. It is important to be able to formally express topological constraints directly. A contribution to research in this area was the GDM language to express topological integrity constraints put forward by Hadzilacos et al (Hadzilacos, et al., 1992). This was later enhanced in (Hadzilacos & Tryfona, 1996) to give the georelational data model GRDM.

5.2. Semantic integrity constraints

These differ from topological integrity constraints in that they are concerned with the *meaning* of geographical features. An often-quoted (and encountered) data quality problem is that of road centrelines not meeting. The concern is the *topological* consistency of the line object road centre line, which has implications for analysis. This could be addressed regardless of the semantic information that this is a road. Semantic integrity constraints apply to database states that are valid by virtue of properties of objects that need to be stored. In this category, an example would be a rule that stated that a user must not enter a road running through any body of water in a class of 'water' objects including rivers, lakes and streams. If the user attempts to enter, for example, a road running through a lake a semantic rule would be activated stating that this is a body of water and a road would not normally run through it.

5.3. User defined integrity constraints

These differ from semantic integrity constraints that are more esoteric in nature and not necessarily based on semantics. User defined integrity constraints allow database consistency to be maintained according to *user defined* constraints analogous to business rules in non-spatial DBMS. For example, for external or legal reasons it may desirable to locate a nuclear power station a given distance from residential areas. When attempting to enter a case where this does not occur, a user rule would be activated.

6. Spatial constraint taxonomy

In spatial systems there seems no reason to assume that all the integrity constraints defined in section 5 are static. In fact this is counter intuitive because spatial data is, by its very nature, continually changing and it is often these changes that we wish to document. However it is desirable to express them in static fashion in the DDL where possible. Thus a taxonomy is presented based on two dimensions. On one axis is the static/ transitional distinction made in section 4.1. On the other axis these are classified in terms of semantic/user/topological constraints described in section 5. This results in six combinations. These six combinations are discussed in section 7.5 and illustrated in Figure 5. In addition to the six combinations, inherent constraints, for example entity/referential constraints are included to give a full taxonomy.

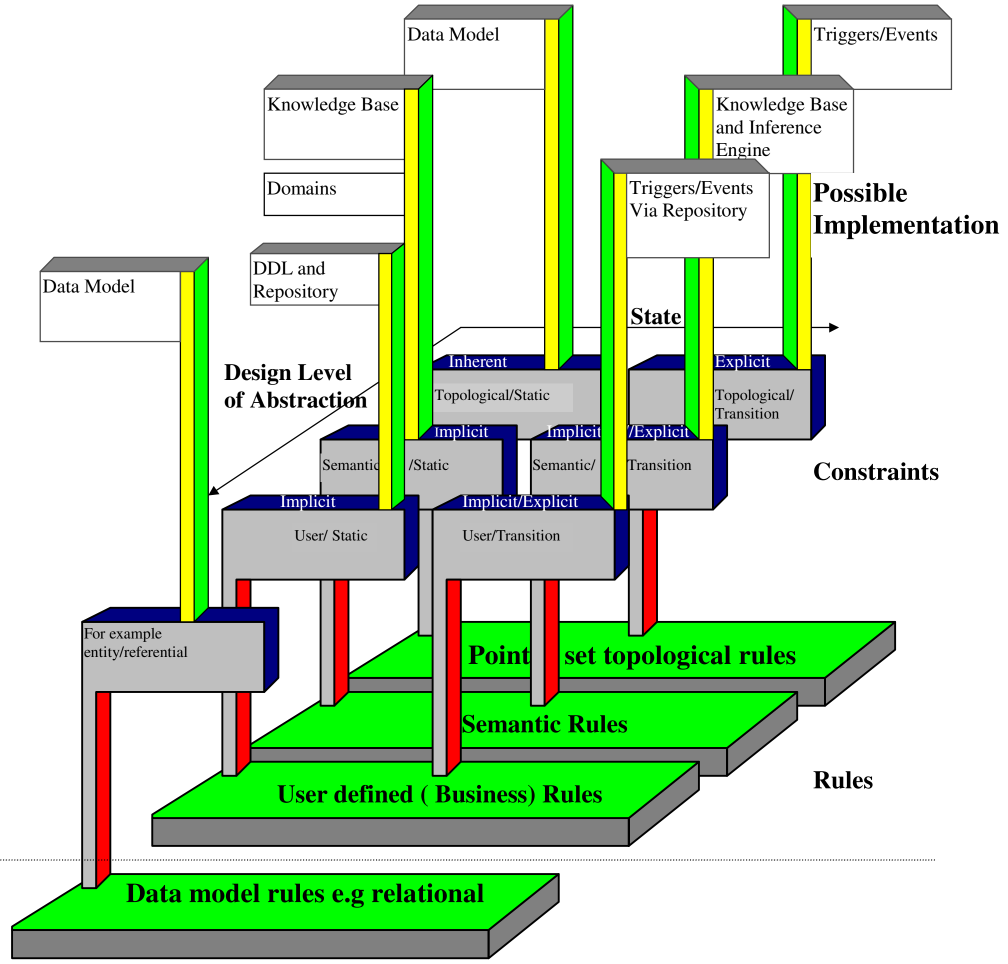

Figure 5 Taxonomy of spatial integrity constraints

In this section an example of each type of rule is given. It is not the intention of this paper to formally define the constraints of the taxonomy; the emphasis is more on classification. Part of the reason for this is because the diverse nature of the constraints. The examples 1,3 and 4 given may be readily implemented in the constraint language proposed by Date (Date, 1990) because they are based on attribute values. This would not be the case with a static semantic constraint such as ‘a road may not cross a building’. This example has its basis in topology and thus before it could be defined, a topological model, such as the Triangulated Irregular Network, Spaghetti Model of Topological model of Aronoff (Aronoff, 1989), would have to be assumed. This is also the case with examples 2,5 and 6. It should be noted however that the latter three examples, whilst based on topology are all at different levels of abstraction. This is what provides the basis for this classification.

1. **Static semantic:** the height of mountain may not be negative.
2. **Static topological:** all polygons must close
3. **Static user:** All streets wider than seven metres must be classified as highways.
4. **Transition semantic:** A spatial example of this would be that the height of a mountain may not decrease.
5. **Transition topological:** If a new line or lines are added making a new polygon the polygon and line tables must be updated to reflect this.
6. **Transition user:** Road of any type may not be extended into body of water of any type

7. Implementation

This paper is concerned with the spatial constraints one may want to impose as opposed to how to implement them. A brief description is, however, given here. Their implementation is to a large extent a matter of knowledge representation. The required knowledge could be represented as rules, predicate logic, semantic nets, a database with the appropriate facilities for data mining or a piece of procedural code. What makes expert systems "expert" is their behaviour rather than the means by which they accomplish this behaviour. Referring to Figure 5 above a discussion follows on how each type of rule could be implemented

7.1. User defined constraints

It is suggested that user defined rules may be stored and enforced by an active repository. This would obviously necessitate facilities for the generation of code to impose the constraints and some language for expressing them.

7.2. Semantic constraints

Semantic rules may be more appropriate to expert system approaches than the ad hoc user rules because the meaning behind topological relationships is based on reality. Since these are true, real world phenomena there may be some value in developing a knowledge base to 'learn' or reason about them. The object-oriented approach is particularly appealing as an option for implementing this type of constraint because of the ability of objects to inherit characteristics and constraints from members of their superclass in the form of class attributes.

7.3. Topological constraints

It has been shown that topological constraints may be enforced using an integration of logic programming and databases (Egenhofer, et al., 1990). Laurini and Milleret-Raffort (Laurini, et al., 1991) suggested that constraints derived from computational geometry could be enforced using private methods of the object oriented approach. Research is under way (Ubeda, et al., 1996) into a language for expressing and checking topological constraints.

7.4. Inherent constraints

Just as Entity /Referential constraints are inherent to database schemas of the relational model, topological integrity constraints are inherent in the Spatially Extended Entity Relationship (SEER) model (Firns, 1994). In the SEER model the value of an entity set's geometric type class attribute imposes constraints upon the genuses of spatial relationship types in which it can participate. With reference to Figure 6:

"*Any pair of entity sets that are both related to a sub-type of a node entity set are potentially related to one another spatially"*

This is because a node entity set is the means by which 'thematic layers' are modelled in SEER notation.

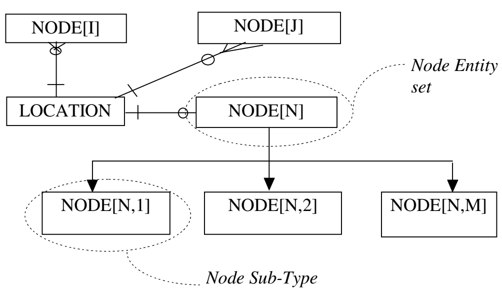

Figure 6 Node sub-types for a thematic layer representing *m* different types of spatially distributed phenomena (Firns, 1994)

7.5. Importance of taxonomy

The taxonomy in Figure 5 identifies six classes of spatial integrity constraints. There is an implicit assumption that constraints are set up in response to rules. The classes of rules on which the constraints are based are also shown. A discussion on the implementation options that are open for them was given in section 7. There are three important functions of this taxonomy.

7.5.1. Clarification of terms

Many discussions of integrity constraints are clouded by issues of implementation. Whilst implementation suggestions are made here, the main purpose is to identify where the similarities and differences exist between computer science and SIS terms.

7.5.2. Security

The second function of the taxonomy is related to security. One aspect of security is the ability to have control over which people within an organisation can make adjustments to which classes of constraints. If, as suggested earlier, all constraints are documented in a central repository a system of authorisations based on this taxonomy will be easier to implement. That is, who should have the right to alter the way constraints are set up in the first place, and which constraints should they have access to? The personnel in question could be the database administrator, user or other SIS managers.

7.5.3. The use of a repository for error warnings and data quality reporting

The final function of the taxonomy is to provide guidelines for the severity of error warnings. Depending on which type of constraint has been violated differing levels of severity warning will be issued by the repository. For example, fundamental topological errors would result in a total block on the user's progression until the problem was fixed whereas a more esoteric user rule could be designed to be overridden with a short warning. In either case the potential for generating a log of all constraint violations would be useful for reporting purposes. This issue was raised in section 2.

8. Conclusion

This paper has examined the issue of data integrity in spatial information systems. A taxonomy based on database state and degree of abstraction has been presented. It is suggested that user defined spatial integrity constraints are a superset of standard mainstream database constraints, the difference being the effect of topology on the possible relationships between database entities and the constraints thereon. A number of suggestions for the implementation of such constraints are given. 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.

9. References

1. Aronoff, S. [1989]. *Geographic Information Systems: A Management Perspective*. Ottawa, Canada: WDL Publications.
2. Chadwick, A. [1995]. "An Architecture to support business rules in custom GIS applications". In *Proceedings: 18th AM/FM International Conference,* 1047-1058.
3. Collins, F. C., & Smith, J. L. [1994]. "Taxonomy for error in GIS". In *Proceedings: International symposium on spatial accuracy in Natural Resource Data Bases "Unlocking the puzzle",* Williamsburg, Virginia: R. G. Congalton (Ed.), American Society for Photogrammetry and Remote Sensing. 1-7.
4. Congalton, R. G. [1994]. *International symposium on spatial accuracy in Natural Resource Data Bases "Unlocking the puzzle"*. Williamsburg, Virginia: American Society for Photogrammetry and Remote Sensing.
5. Date, C. [1990]. "A contribution to the study of database integrity". In *Relational Database Writings 1985-1989,* (pp. 185-215), Saratoga: Addison Wesley.

6. Egenhofer, M.J., [1994]. "Pre-processing Queries with spatial constraints", *Photogrammetric Engineering and Remote sensing*, **60**(6), 783-790.

7. Egenhofer, M. J., & Frank, A. U. [1990]. "LOBSTER: combining AI and database technologies for GIS", *Photogrammatic Engineering and Remote Sensing*, **56**(6), 919-26.

8. Egenhofer, M. J., & Franzosa, R. D. [1991]. "Point-set Topological Relations", *International journal of geographical Information Systems*, **5**(2), 161-174.

9. Elmasri, R., & Navathe, S. [1994]. *Fundamentals of Database Systems*. Benjamin/Cummings.

10. Fahrner, C., Marx, T., & Philippi, S. [1995]. *Integration of Integrity Constraints into Object Oriented database schema according to ODMG-93* (RR-9-95). University of Koblenz.

11. Firns, P. G. [1994]. "An Extended Entity Relationship Model Applicable to the design of Spatially Referenced Databases", *Ph.D. Thesis*, University of Otago, Dunedin, New Zealand.

12. Günther, O., & Lamberts, J. [1994]. "Object-oriented Techniques for the Management of Geographic and Environmental Data", *The Computer Journal*, **37**(1), 16-25.

13. Hadzilacos, T., & Tryfona, N. [1992]. " A model for expressing topological integrity constraints in geographic databases". In *Proceedings: Theories and Methods of Spatio-Temporal Reasoning in Geographic Space. International Conference GIS - From Space to Territory.,* Pisa , Italy: A. U. Frank, I. Campari, & U. Formentini (Ed.), 252-68.

14. Hadzilacos, T., & Tryfona, N. [1996]. "Logical data modelling for geographic applications", *International journal of Geographical Information Systems*, **10**(2), 179-200.

15. Hunter, G., & Beard, M. [1992]. "Understanding error in spatial databases", *The Australian Surveyor*, **37**(2), 108-119.

16. Hunter, G. J. [1996]. "Management issues in GIS: Accuracy and Data Quality". In *Proceedings: Conference on managing Geographic Information Systems for success,* Melbourne, Australia: G. J. Hunter (Ed.), Aurisa. 95-101.

17. Jones, C., & Luo, L. [1994]. "Hierarchies and objects in a deductive spatial database". In *Proceedings: Advances in GIS research: Proceedings of the 6th International Symposium on spatial data handling,* R. G. Healy (Ed.), 589-603.

18. Kanellakis, P. C., Kuper, G., Revesz, P.Z. [1995]. "Constraint Query Languages", *Journal of Computer and System Sciences*, **51**(1), 26-52.

19. Laurini, R., & Milleret-Raffort, F. [1991]. "Using integrity constraints for checking consistency of spatial databases". In *Proceedings: GIS/LIS 91,* Atlanta, Georgia: 634-642.

20. Luo, L., & Jones, C. [1995]. "A deductive database model for GIS". In *Innovations in GIS 2,* (pp. 33-42), P. Fisher (Ed.), Taylor and Francis.

21. Marble, D. F. [1990]. "The extended data dictionary: A critical element in building viable spatial databases". In *Proceedings: 11th annual ESRI user conference,*

22. Medeiros, C. B., & Andrade, M. J. [1994]. "Implementing Integrity Control in Active Data Bases", *Journal of Systems and Software*, **27**, 171-181.

23. Medeiros, C. B., & Cilia, M. [1995]. "Maintenance of binary topological constraints through active databases". In *Proceedings: 3rd ACM International Workshop on Advances in Geographic Information Systems,* Baltimore, Maryland USA: P. Bergougnoux (Ed.), 127-133.

24. Medeiros, C. B., & Magalhaes, G. C. [1993]. *Rule application in GIS - a case study* (DCC-18/93). UNICAMP.

25. Smith, T., Ramakrishnan, R., & Voisard, A. [1992]. "An Object-Based Data Model and a Deductive Language for Spatio-Temporal Database Applications". In *Proceedings: Proceedings of the Workshop of the BRA Esprit project ``GOODS",* M. Gambosi, Scholl, & P. Widmayer (Ed.), Springer-Verlag, Berlin,.

26. Stonebraker, M., & Kemnitz, G. [1991]. "The Postgres next-generation database management system", *Communications of the ACM*, **34**(10), 78-92.

27. Ubeda, T., & Servigne, S. [1996]. "Geometric and Topological Consistency of Spatial Data". In *Proceedings: 1st International Conference on Geocomputation,* Leeds UK. 830-842.

28. Worboys, M. F. [1995]. *GIS: A Computing Perspective*. London: Taylor and Francis.