# MurMur: Database Management of Multiple Representations

**Christine Parent** and **Stefano Spaccapietra** and **Esteban Zimányi**

University of Lausanne
CH 1015 Lausanne, Switzerland
christine@lbd.epfl.ch

Database Laboratory
Swiss Federal Institute of Technology
1015 Lausanne, Switzerland
Stefano.Spaccapietra@epfl.ch

Department of Informatics
Université Libre de Bruxelles
1050 Bruxelles, Belgium
ezimanyi@ulb.ac.be

## Abstract

Successful information management implies the ability to design accurate representations of the real world of interest to targeted applications. Current systems do not provide representation schemes supporting the diversity of user needs. In the context of interoperability, as in Web access to heterogeneous data sources, they cannot properly integrate the diversity of stored representations. The objective of the European project described in this paper, MurMur[1], is to enhance GIS (or DBMS) functionality so that, relying on more flexible representation schemes, users may easily manage information using multiple representations. The added functionality will support multiple coexisting representations of the same real-word phenomena (semantic flexibility), including representations of geographic data at multiple resolutions (cartographic flexibility). This will in particular make possible a semantically meaningful management of multi-scale, integrated, and temporal geodatabases.

## Introduction

Databases are intended to keep an integrated and consistent set of data that provides the information needed to support application requirements from one or several user communities. That data represent real-word phenomena that are of interest to its users. While the real world is supposed to be unique, its representation depends on the intended purpose. Thus, different applications that have overlapping concerns about real-word phenomena normally require different representations of the same phenomena. Differences may arise in all facets that make up a representation: what information is kept, how it is described, how it is organized (in terms of data structures), how it is coded, what constraints, processes and rules apply, how it is presented, what are the associated spatial and temporal framework, etc.

Current data management technology relies on a centralized representation paradigm, where all application requirements are integrated into a single stored representation (at least at the logical level). A view mechanism allows deriving, on demand, from the stored representation any other representation that materializes the specific viewpoint of the requesting application. View mechanisms, however, are subject to strong limitations. In relational DBMSs it may not be possible to update the database using a view, because of the inherent ambiguity of updates on views that do not rely on a 1:1 mapping between tuples in the view and underlying tuples in the database. In object-oriented DBMSs view definition is further restricted to very simple views (e.g., defined by projection operations) because the rules governing data model constructs lead to inconsistencies in case of more complex view definitions.

What a view-based, centralized representation mechanism can definitely not support is the case where different application viewpoints are not derivable from each other (irreducible viewpoints). Assume a hospital information system, such that patients are identified by medical teams based on a patient number inscribed on a bracelet that the patient always carries, and the same patients are identified by the administrative staff based on a social security number. If the two viewpoints do not share other information (such as name and birth date) that could provide a common identification scheme, when the patient leaves the hospital two different update operations have to be made for the medical and the administrative realms (no update propagation from one realm to the other is possible). This has evident drawbacks in financial terms (double cost for updates) and in terms of consistency of the database, that cannot be guaranteed by the system and has to rely on manual procedures.

From a traditional, centralized database system perspective, the coexistence of irreducible viewpoints in a database may be considered as a design error. From a user perspective it is not. In current DBMSs it is up to application designers and users to cope with the situation of inter-relating different representations of the same phenomenon, relying on primitive system functionality, such as foreign keys in relational DBMSs or is-a links in object-oriented DBMSs. It is again up to users and application designers to define and enforce

Copyright © 2000, American Association for Artificial Intelligence (www.aaai.org). All rights reserved.

[1] The Murmur Group includes the following partners:

- Star Informatique, Liège, Belgium
- CEMAGREF, Grenoble, France
- Institut Géographique National, Paris, France
- Université Libre de Bruxelles, Belgium
- Université de Lausanne, Switzerland
- Ecole Polytechnique Fédérale de Lausanne, Switzerland

the appropriate consistency rules that may constrain the set of representations.

The centralized representation paradigm is even more uncomfortable when a database results from the integration of data from different pre-existing data sets, as is the case in federated or cooperative information systems. Such systems are more and more frequently required to support interoperation among different organizations, as well as for a single organization that needs to coalesce data from different sources, including the Web, to support its enterprise strategy. When data from various sources come together into a single data store, the situation where different representations of the same phenomena coexist is likely to happen and cannot be considered as a design error.

In summary, modern data management requires a new representation paradigm where multiple representations of the same phenomenon may coexist in a database, and this should be explicitly described and made known to the system so that it may manage the situation accordingly. In other words, existing data models need to be extended with new concepts such as a multi-representation link, with a well-defined semantics (which says “this representation is about the same real-word phenomenon that this other representation”), and associated constraints and operators. This is part of the research agenda of the MurMur project that started January 1st, 2000, with the support of the European Community. Expected benefits include better real-word modeling, enhanced understanding of schema diagrams and database content, improved consistency management, automatic update propagation, and data cleaning facilities (when two representations are used to check one against the other and determine if there has been some erroneous data acquisition).

## What is multiple resolution and why it is needed

Geographic data is a major provider in terms of systems and services for the citizen, as one of the most common concerns in everyday life is locating something you are looking for. Maps are the most natural way to provide location information, and also serve as an excellent mean of visualizing analytical data about phenomena that have a geographical correlation. Hence, map production and display is an economically significant activity, nowadays supported by GIS and DBMS databases. This includes geography-compliant maps, that show items of interest as faithfully as possible with respect to their real-word location and shape, as well as schematic maps (e.g., city transport systems, airline connections diagrams, train networks, facility management networks), where the focus is on correct connections and readability rather than on precisely locating lines and nodes.

A map is drawn according to a given scale. At different scales, the same information is usually drawn differently, a physical zoom-in or zoom-out is simply inadequate. The reason is that drawing standards may change from one scale to another one, items may (dis)appear or be (dis)aggregated because their size make them (in)visible depending on the scale, their shape may be modified (made simpler or more precise), or simply the information is not available at the requested scale. Unfortunately, there is no complete set of algorithms that automatically derive a map at some scale from a map at a more precise scale. Some algorithms exist and more are being investigated, e.g., within the European project AGENT [2], and the process they implement is called cartographic generalization. Given this situation, map production systems tend to keep a separate database per scale, leading to problems such as lack of consistency and uncertain update propagation.

The ideal setting would be of course to maintain a database where geometry information is kept at the most precise scale, and all geometries at less precise scales are automatically computed through cartographic generalization. Because this derivation cannot be fully automated, and also because cartographic generalization may be a long and costly process, the alternative is to perform cartographic generalization off-line and to store its result for direct reusability. Eventually, that means that a spatial object in a database may be associated to a variety of geometric representations that are scale-dependent. Databases with such a facility are called multi-scale databases. Actually, while the scale concept is perfectly understood and relevant when talking about maps, it is not relevant anymore when talking about representation of an object in a database (scale has to do with map drawing, not with object representation). When focusing on databases, it is more correct to use the term “resolution”, usually defined as the minimum dimension of an object to be represented. The resolution of information in the database is the resolution used at data acquisition. If different resolutions have been used for the same objects, we can talk about multi-resolution objects. Moving among resolutions has potential impact on the shape of objects (shape may be simplified in less precise resolutions), on their values (because of a corresponding change in semantic resolution according to user-defined hierarchical value domains), as well as on the existence of objects (because of a change in aggregation rules or creation of new aggregates). From the geodata application perspective, multiple resolution is certainly the most urgent multi-representation problem to solve.

## Multiple representation due to time-varying information

A wide range of applications need to manage time-varying information for analysis, planning, and forecast, in particular for decision support systems. This include classical database applications such as personnel management, medical records, financial applications, travel reservation systems, to name but a few. Examples of geographical applications needing temporal support include cadastral, risk management and environmental applications. A map is also characterized by a given time period of validity. Of particular importance for both classical and geographical applications is the manipulation of moving objects, such as cars, vessels, and pollution disasters, where the geographic

[2] http://agent.ign.fr/

characteristics of an object are time- varying. This is a currently unresolved problem for which several proposals have recently appeared in the research literature.

By definition, keeping temporal information means that each recorded fact has multiple representations that correspond to different points in time. When introducing a temporal dimension into data management it is necessary to determine which aspects of time are relevant for the application. There are two ways of adding time to information. The usual way consists in timestamping facts representing real-world phenomena. Thus, attribute values can have an associated period of validity, and objects and relationships can have an associated life cycle (i.e., when they are created, deactivated, reactivated, and deleted). The other association of time with information concerns the relative positioning of activities in time, that models aspects of the inter-object dynamics within the application (e.g., that object has created that other object; that object lived before that other one; this object is a snapshot of that other one). Although such relationships are rarely supported in existing systems, they are important in particular for applications related to the management, the analysis and the understanding of natural and human phenomena.

Another important aspect is the manipulation of data at different granularities (e.g., year, month, day, hour, minute, second). Temporal support requires functions to convert from one unit to another.

MurMur shall support the above basic components of temporal modeling as needed in the application framework of the test cases. We anticipate, for instance, that valid time will be the priority, rather than transaction time. The basic aim will be to obtain consistent different representations of a given set of objects and relationships at different points in time.

## Related Work

Multiple representation is the necessary consequence of both the subjectivity of perception and the diversity of interests during the database design phase. Traditional relational systems support multi-representation through the view mechanism. As little semantics is embedded in data structures, data items can be rearranged in any desirable way using relational algebra operators and the new structure stored as a view definition. Object-oriented database systems follow the same idea, although they can not support the same flexibility in data restructuring as relational systems. The semantics embedded in composition structures and in generalization links leads to unavoidable restrictions in the extent of the view definition mechanism. Beyond the easiest solutions (i.e., not allowing whatever cannot be easily managed by current object- oriented systems), there has been an increasing number of proposals to support multiple representation through multi-instantiation, i.e., allowing an object to be simultaneously represented in several classes. Some proposals aim at supporting multi-instantiation at the object level. Different instance level mechanisms have been designed to implement the possibility for objects to be dynamically grouped to form new classes, while keeping their membership in their original classes (Papazoglou, Kramer, & Bouguettaya 1994; Rieu *et al.* 1991). The other approach deals with the issue at the schema level. An analysis of the object life cycle determines the set of relevant possible representations for an object. These representations, called “roles”, allow to formulate that an object may play several roles during its life span (Albano, Ghelli, & Orsini 1995; Gentile 1996; Pernici 1990).

Even if only a few of these database mechanisms have been investigated for spatial databases, they all can be used for spatial multiple representations. For example, view definition can still be used to support multiple points of view (Claramunt 1998), but multi-representation needs cannot be fully satisfied using such a purely deductive approach. Spatial databases have to maintain multiple representations that are not deducible one from the other. This is namely the case in most cartographic applications, where maps of the same land at different scales are managed and spatial generalization (i.e., derivation of a less precise spatial representation from a more precise representation, which reduces the accuracy and the resolution of the dataset) is used to determine visual representations at different scales. The majority of spatial generalization processes are complex and interactive (i.e., need human intervention). So it is obvious that the view mechanism could not be used in this case. The result of an interactive generalization process has to be stored in order to be reused. But spatial generalization processes are usually very long and memorizing the result prevents from re-calculating it for each request.

(Kidner & Jones 1994; Francalanci & Pernici 1994; Timpf & Franck 1995) proposed to keep the different-scale representations obtained from an interactive cartographic generalization process. Objects are stored in a hierarchical data structure, where levels correspond to increasing detail and the various representations of the same object are linked together. Objects can then be used to compose a map at a particular scale. (Bauzer Medeiros & Jomier 1994; Bauzer Medeiros, Bellosta, & Jomier 1996) extended the version mechanism with some view operators. The version mechanism allows modeling of reality according to different points of view. Besides, the proposed model stores for an object several attributes referencing its representations at different scales.

The multi-faceted problem of multi-representation is presented in detail in (Scholl *et al.* 1996; Rigaux & Scholl 1995) as well as a discussion about the problems entailed and the potential solutions. They also study the impact of spatial and semantic resolution on data representation from both the modeling and querying points of view. They propose a model allowing representing particular kind of data, zones that fit into each other. The associated language allows database querying of attributes with hierarchical domains without exact knowledge of the data abstraction level.

(Stell & Worboys 1998) propose a system helping to process and reason with spatial data sets heterogeneous with regard to semantic and spatial resolution. The granularity of a representation specifies the levels of detail with respect to which the data are registered. They distinguish between spatial and semantic granularity. Their multi-representation system, called “stratified map space”, consists of a granularity lattice and for each granularity a "map space". The schemas corresponding to the semantic granularity and the datasets, which have a representation at this spatial resolution, are associated to the map space.

These approaches do not meet all our requirements for multi-resolution. We have then extended and integrated the choices and the concepts proposed in the previous works to propose a global and coherent set of concepts for multi-resolution. To be able to keep various values/geometries according to the resolution, different approaches were proposed: 1) to define different classes for the same real world entity, one for each resolution, and to link them (Kilpelaïnen 1998), 2) to describe a real-world entity as the association of one class defining its semantic characteristics and one or several classes defining its spatial representation, one for each resolution (Tryfona, Pfoser, & Hadzilacos 1997), and 3) to define the geometry as a spatial attribute with n values (Claramunt 1998; Timpf & Franck 1995). The main drawback of the first and second solution is that they multiply the number of classes in the schema and the number of objects. Moreover, we consider that the geometry is a property of the real-world entity and as such it has to be represented as an attribute of the object. (Claramunt 1998) does not offer a complete conceptual solution. In our approach, stamping is used to define the geometry as a spatial attribute with n values and to automatically access its values according to the resolution. The "Directed Acyclic Graph" of (Timpf & Franck 1995) could be an implementation of our conceptual model.

As in (Stell & Worboys 1998), we define two hierarchies of resolution, one for spatial resolution and one for semantic resolution but we extend their application domain to associations. From (Rigaux & Scholl 1995), we keep the possibility to describe and query attributes at various levels of semantic resolution and we extend it to object classification hierarchies. Moreover, we propose to use aggregation relationships to handle more complex cases where a real entity is described by several different sets of objects, as well as to derive the value of attributes and to define spatial integrity constraints.

## References

Albano, A.; Ghelli, G.; and Orsini, R. 1995. Fibonacci: A programming language for object databases. *Very Large Data Bases Journal* 4(3):403–444.

Bauzer Medeiros, C., and Jomier, G. 1994. Using versions in gis. In Karagiannis, D., ed., *Proc. of the 5th Int. Conf. on Database and Expert Systems Applications, DEXA'94*, LNCS 856, 465–474. Athens, Greece: Springer-Verlag.

Bauzer Medeiros, C.; Bellosta, M.-J.; and Jomier, G. 1996. Managing multiple representations of georeferenced elements. In Wagner, R., and Thoma, H., eds., *Proc. of the 7th Int. Conf. on Database and Expert Systems Applications, DEXA'96*, LNCS 1134, 364–370. Zurich, Switzerland: Springer-Verlag.

Claramunt, C. 1998. *Un modèle de vue spatiale pour une représentation flexible de données géographiques*. Ph.D. Dissertation, Département d'Informatique, Université de Bourgogne, France.

Francalanci, C., and Pernici, B. 1994. Abstraction levels for entity-relationship schemas. In Loucopoulos (1994), 456–473.

Gentile, M. 1996. *An Object-Oriented approach to manage the multiple representations of real entities*. Ph.D. Dissertation, Ecole Polytechnique Fédérale de Lausanne, Switzerland.

Kidner, D., and Jones, C. 1994. A deductive object-oriented gis for handling multiple representations. In Waugh, T., and Healey, R., eds., *Advances in GIS: Proceedings of SDH'94*, 882–900. Edinburg, Scotland: Taylor & Francis.

Kilpelaïnen, T. 1998. Maintenance of topographic data by multiple representations. In *Proc. of the Annual Conference and Exposition of GIS/LIS*.

Loucopoulos, P., ed. 1994. *Business Modeling and Re-Engineering, Proc. of the 13th Int. Conf. on the Entity-Relationship Approach, ER'94*, LNCS 881. Manchester, UK: Springer-Verlag.

Papazoglou, M.; Kramer, B.; and Bouguettaya, A. 1994. On the representation of objects with polymorphic shape and behaviour. In Loucopoulos (1994), 223–240.

Pernici, B. 1990. Objects with roles. In *Proc. of the Conf. on Office Information Systems*, 205–215.

Rieu, D.; Nguyen, G.; Culet, A.; Escamilla, J.; and Djeraba, C. 1991. Instanciation multiple et classification d'objets. In *Proc. of the VIIèmes Journées Bases de Données Avancées*.

Rigaux, P., and Scholl, M. 1995. Multi-scale partitions: Applications to spatial and statistical databases. In *Proc. of the 4th Int. Symp. on Advances in Spatial Databases*, LNCS 951, 170–183. Springer-Verlag.

Scholl, M.; Voisard, A.; Peloux, J.; Raynal, L.; and Rigaux, P. 1996. *SGBD Géographiques, Spécificités*. International Thomson Publishing.

Stell, J., and Worboys, M. 1998. Stratified map spaces: A formal basis for multi-resolution spatial databases. In *Proc. of the Int. Symp. on Spatial Data Handling, SDH'98*, 180–189.

Timpf, S., and Franck, A. 1995. A multi-scale dag for cartographic objects. In *Proc. of Auto Carto 12*, 157–163.

Tryfona, N.; Pfoser, D.; and Hadzilacos, T. 1997. Modeling behavior of geographic objects: An experience with the object modeling technique. In Olivé, A., and Pastor, J., eds., *Proc. of the 8th Int. Conf. on Advanced Information Systems Engineering, CAiSE'97*, LNCS 1250, 347–359. Barcelona, Spain: Springer-Verlag.