ESRI International User Conference July 25–29, 2005
San Diego Convention Center
San Diego, California

Multidimensional Modeling based on Spatial, Temporal and Spatio-Temporal Stereotypes

Gabriel Pestana, Miguel Mira da Silva1
INESC-ID/IST, Rua Alves Redol, 9
1000-029 Lisboa, Portugal

{gabriel.pestana, mms}@inesc-id.pt

Abstract

Spatial data provide valuable new insight that is invisible and sometimes not even possible to generate using traditional spreadsheets and data processing alone. Decision makers prefer – especially for queries on spatial, temporal, and spatio-temporal data – to drill around spatial functionalities and operations in order to speed up information flow and standardize map-reporting visualizations. However, spatial data management has also been plagued with data interoperability issues that make it difficult for companies to easily integrate and apply spatial information.

In this paper, we propose a multidimensional spatio-temporal data model to enable spatial analysis, in a context of evolving specifications. The proposed data model addresses the problem of spatial and temporal data integration by providing information to facilitate semantic interoperability and data analysis in a spatial DW that uniformly handles all types of data. Using a practical example in the field of land parcels, we evaluate the implementation of the model.

1. Introduction

A wide variety of applications need to capture spatial and time-varying characteristics of the entities they model. With the advent of mobile computing and location-based services, even more amount of spatial data is being collected and stored in business information systems. Therefore spatiotemporal data are becoming a corner stone for decision-makers to analyze business data in a spatial context [1]. However, the exploitation of spatial data, such as location-based data or customer profiles location-dependent contents, within the decision process is still done below of its full capabilities [2]. The traditional GIS applications are more focused in operational data requirements and usually working with spatial data separately from the business data. This loosely coupled approach has a data integrity drawback because the two types of data are managed apart [3].

On the other hand, decision-making users frequently rely on DW as an important platform for data analysis. DW architecture and OLAP tools provide the access to historical data with the aim of supporting the strategic decisions of organizations. OLAP tools allow users to dynamically manipulate the data contained in a DW. OLAP tools use hierarchies for allowing both a general and a detailed view of data using operations such as drill-down and roll-up [4].

The structure of a DW is usually represented using the star/snowflake schema, also called multidimensional schema, made up of a set of (fully denormalized) dimension tables, one for each dimension of analysis, and a fact table whose primary key is obtained by composing the foreign keys referencing the dimension tables. The most common class of queries used to extract information from a star schema are GPSJ queries [5]. A GPSJ (Generalized Projection-Selection-Join) query consists of a selection, over a generalized projection over a selection over a join between the fact table and the dimension tables involved. Within the scope of this paper we will assume that selections are always conjunctive range statements expressed on attributes in the dimension tables.

However, decision-makers often perceive the decision-making process to solve complex multi-dimensional spatial problems as unsatisfactory [6, 7]. Current multidimensional database technologies usually do not support the spatial data structures [4]. Moreover, the domain of conceptual design for multidimensional modeling does not always cope with the different kinds of hierarchies existing in real-world applications. Consequently spatiotemporal data do not always have a natural hierarchy that can be used at design time to efficiently compute pre-aggregation in the DW [8].

Therefore, we may conclude that spatiotemporal queries are complex because they are semi-structured or ill defined in the sense that the goals and objectives are not completely defined, which constrains traditional DW and OLAP tools to fully exploit spatial data. The problem is that the positions and the ranges of spatiotemporal query windows usually do not conform to pre-defined hierarchies, and are not known in advance. Without the hierarchy concept, DW can not pre-aggregate on spatial dimension [7]. In this domain, it is logical that grouping data heavily contributes to the global query cost and that such a cost can be reduced by pre-computing the aggregated data that are useful to answer a given workload.

The dynamic behavior of real objects further complicates the development of spatiotemporal queries. For instance: For every square kilometer in every county in a specific wine region, what has been the density of the vineyards for the last five year? In this spatiotemporal query the hierarchy criteria is ad-hoc and the spatial dimension may be volatile, i.e., the land parcels at the finest granularity may evolve over time. For instance, the cultivated area may change according to diverse conditions. Therefore, for the realisation of fast and flexible on-line analytical processing in a spatial DW environment, spatial indexing and accessing methods have to be implemented for efficient storage and access of spatial data. We will present some of the methods we used to improve such queries.

When analyzing historic data, spatial OLAP queries consider many parameters in the geoprocessing workflow, making it very difficult for decision-makers to keep track of all the facts, assumption, and datasets of existing values. In this paper, we study a multidimensional data model to store spatial data and analyze the requirements to implement efficiently on-line analytical processing of spatial data (i.e., spatial OLAP queries).

The proposed data model uses a visual modeling tool, named Perceptory [9], to support spatiotemporal data modulations. We are particularly interested in modeling geometric shape evolutions over time. Our approach preserves the traditional star schema [10] while bringing new spatial OLAP capabilities into the decision process.

The rest of the paper is organized as follows. Section 2 refers to related work on Spatial DW. Section 3, presents an overview of stereotypes and describes Perceptory graphical notation for spatial data modeling. Section 4, presents the problem we are trying to solve. Section 5, describes our proposal to solve the problem. Section 6 presents a practical example, and Section 7 concludes the paper, outlining how the model will be extended.

2. Related Work

A multidimensional spatial data model operates with facts and dimensions providing the descriptive characteristics of spatial features that bring spatial OLAP analysis to life. This means that spatial measures can be associated to a fact relationship, independently of whether the relationship is spatial or not. We define a spatial measure as a measure that either is represented by a geometry or numerical value that is calculated using spatial or topological operators.

If a multidimensional model contains spatial data, it should specify the spatial functions used for aggregation (roll-up and drill-down operations) along the hierarchies. These hierarchies may be spatial or non-spatial.

Current OLAP tools automatically aggregate numerical measures along hierarchies using deferent types of functions: distributive, algebraic, and holistic [11]. Likewise, aggregation functions for spatial data have also been defined [12]. For example, spatial distributive aggregates include *convex hull*, geometric *union*, and geometric *intersection*. Examples of spatial algebraic functions are *center of n geometric points* or *center of gravity*, while examples of spatial holistic functions are *equi-partition* or *nearest-neighbor index* [11].

As for spatial measure, a spatial dimensions (in addition to the usual descriptive attributes) may also store attributes of a spatial data type, enabling spatial operations such as *overlap*, *contain*, *intersect*, *merge*, or *split*.

Most of these issues are not new and some have already been individually addressed in other papers. In [13] the authors present some pioneer work in this area by proposing a spatial DW framework to select spatial objects for materialization (pre-aggregation). They also propose the idea of spatial measures defined as pointers to regions in space. The use of a spatial star-schema is also proposed by [14], with methods to process arbitrary aggregations. In [15], the building of spatiotemporal topological operator dimensions (STTOD) is proposed. The same research group has been working on extending spatial measures by the inclusion of measures represented as spatial objects or calculated using spatial metric or topological operators. We follow a resembling approach.

These cases, however, do not cover ad-hoc queries. An ad-hoc query not confined by the hierarchy would still need to access the fact table, even if the entire spatial DW was pre-aggregated.

In [16] the research was on methods to achieve the best data approximation within a pre-specified range of data. The focus was on finding the minimum partition of a region. In [17] a spatial DW modeling framework is presented. This work distinguishes different types of measures and dimensions. It highlights important aspects when implementing a spatial DW with commercial off-the-shelf (COTS) tools, namely how to implement a multidimensional spatial data model without paying significant syntax or user interface penalties. Several other proposals [4, 7, 13, 14] focus specialized indexes, and aggregation techniques to manage high volumes of spatial data as well as Spatial OLAP benefits [18]. This last one discusses the problems associated to the implementation of spatial DW and presents a scenario for geographic knowledge discovery using the kinds of dimensions specified in [13].

In this paper we follow the same line of reasoning in the discussion of the problems to implement a spatial

DW. We use Perceptory [9] as the conceptual modelling tool for multidimensional data modeling. Our work analyses the efficiency to exploit business data and the corresponding evolution of spatial object over time, taking into account three perspectives:

- Conceptual multidimensional modelling of spatial data to spatially support the decision-making process.
- Theories and methods of spatiotemporal reasoning, to develop ontology studies for geospatial data interoperability, aggregation and multirepresentation.
- Shortening spatiotemporal heterogeneities and enhance exploratory spatial data analysis through spatial and topological operations.

When focusing on the spatial DW dataset repository, a collection of synthetic data that would simulate a variety of real life scenarios is required. Although there exist several algorithms for generating static spatial data, by introducing motion and thus temporal evolution in spatial object definition, generators tend to be more complex settling new challenges with respect to materialization and indexing issues as well as on the associated query mechanisms.

Most of the indexing proposals in the literature cannot be embedded within existing database systems, therefore not solving the indexing problem from a practical perspective. But, since the development of new complex materialization algorithms or index structures is out of our research scope, we investigate how to map the most promising solutions in order to efficiently query spatio-temporal data using commercial off-the-shelf (COTS) applications. In this way, prototypes based on commercial applications can usually be utilized to solve real-world problems, such as the IVV case study in Section 6.

3. The Problem

To integrate and model spatiotemporal data, several inter-related problems appear. For instance, in a spatial DW the topological relationships can include more than two spatial dimensions [4]. This n-ary topological relationship is not a trivial task because it establishes explicit and multiple hierarchies in dimensions.

Explicit hierarchies are used for aggregating data to the right level of detail in exploratory analyses that use roll-up/drill-down operations [18, 19]. Conversely, support for multiple hierarchies means that multiple aggregation paths are possible. These concepts are important to enable better handling of the imprecision in queries caused by partial containment in spatial dimension structures (e.g., districts would, though approximately, roll up to cities).

Indeed, a huge amount of work has been done during the last few years to formally define the notion of imprecision of spatial data, in particular in the reasoning research area [20].

Furthermore, the spatial measure (e.g., shape and location) may change over time [21]. The spatial dimensions may also be volatile, i.e., the regions at the finest granularity may evolve over time. Considering vineyard evolution for each Wine Region as an example, it is accepted that vineyards cultivation area may change according to buy/sell operations or replanting rights transfer.

This dynamic behavior further complicates the development of spatial data models to support spatial OLAP queries because it requires some elaborate integration of spatial and temporal structures to model the existence and evolution of spatial objects over time.

The introduction of a spatial component in the decision process is undoubtedly most powerful in a DW environment, where high volumes of historical data are queried for decision making purposes. Nevertheless, the cost and response time of spatiotemporal queries is a limiting factor.

Although the experience gained in managing aggregated data in OLAP systems has already been extended for spatial data in spatial OLAP systems [5, 6, 19, 22], it is not clear whether existing index-based approach can efficiently handle aggregate queries that include predicates defined on explicit attributes. Most of the algorithms found in the literature for the evaluation of spatial aggregation queries seem to concentrate only on the problematic of range queries.

Another drawback we identified is that none can evaluate a spatiotemporal aggregate query resulting in a spatiotemporal relation. Once again because they all focus on the range aggregation problem, which after applying spatial and temporal predicates for selecting tuples, ignores the spatial and temporal properties of the qualifying tuples.

I our opinion, there is a need for algorithms that evaluate spatiotemporal aggregate queries that perform spatiotemporal group and partition composition. Such queries would identify the time-varying as well as the spatial nature of the aggregation over spatiotemporal relations. That is, the resulting aggregate value is also a spatial object that changes with time. Current approaches for evaluating spatiotemporal aggregation do not offer support for this kind of queries.

4. A Conceptual Multidaminsional Model

In this section we present the significance and convenience of representing spatial data in a multidimensional model, with a focus in spatial dimensions and spatial measures.

In the spatial data model design process we have to consider how to model spatial features, their geometric shape evolutions over time, and how to generalize /aggregate those features into higher levels of abstraction. Spatially-aware languages have become a recent research focus in academia, industry, and standardization bodies such as the OpenGIS Consortium [23] and IS0 TC211 [24], to extend UML for spatial database development. Figure 1 presents the Plug-ins for Visual Language (PVL), a graphical notation defined by Perceptory to facilitate spatial and temporal modeling. We describe the Perceptory graphical notations only briefly and refer to the literature [9] for further detail.

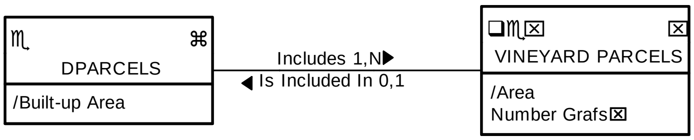

Fig. 1 - Perceptory pictograms formalisms to depict objects geometry and temporality.

In Figure 1, all types of parcels have their geometry and a derived attribute "area" (see UML "/" syntax) while only Vineyard Parcels have geometric evolution. Both types of parcels have an existence; this is represented through temporal pictograms.

The temporal pictogram records a date and the temporal pictogram a period (begin/end dates). The selection between the two depends on the temporal granularity defined into the repository for each class, attribute and geometry. To keep the values that spatial features (attributes) have during their lifespan, a temporal pictogram in placed at the right of a spatial pictogram (and after the multiplicity if it has one).

In addition the basic spatial pictograms for a 2D universe (point, line and polygon) may be mixed to represent more complex geographical notations, as presented in Table 1.

To keep trace of object existence a temporal pictogram is placed next to the name of the class. As explained in [9], the evolution of an object being the aggregate of its states, its default multiplicity is 1,N since those pictograms indicate the user interest to keep trace of the several states of the attribute or geometry it is applied to. Typically, most occurrences will have time to evolve during their life (the N multiplicity), but some won't due to their short life or stability (the 1 multiplicity).

Table 1 - Examples of the PVL Graphical Notation.

Pictogram	Description and Examples
	Complex shape (e.g., an hydrographical network composed of 1D river and 2D lakes)
	Alternative Shape (e.g. Parcel having a 0D shape if < 1 hectare or a 2D shape if > 1 hectare)
	Multiple shapes for an instance (e.g., polygonal municipalities having a non-derivable point located downtown
	Derived shape (example for a municipality centroid derived from other geometric information, i.e. the municipality polygon)

In section 5, we will see that temporal pictograms are most useful to differentiate spatial features with instantaneous existence like "Fire start-ups", or features with a durable life like "Replanting Rights".

5. Proposal

In this section we present the context in which a test bed project was developed jointly with the Wine and Vineyard Institute (IVV, in Portuguese) to conceive a spatially-enabled solution to support decision-makers in wine-policy enforcement and to exploit the sector activity by analyzing the business data.

We propose a multidimensional data model that uniformly integrates both spatial and non-spatial data and differentiates itself from traditional data modeling by including spatial and temporal operations.

By "integrates," we mean not only storing spatial and non-spatial data in the same database, but also to persistently store topologies concerning each geometric change over time. This approach benefits spatial OLAP queries such as: "show me the evolution on the geometry and total number of objects in the regions intersecting a query window qs during a time interval qt".

We consider that the query window (qs) defines a fence or spatial objects in the space to request information about features contained in, adjacent to, or overlapping a specified area [25]. This corresponds to a typical query window that searches for all features within a defined area or spatiotemporal context.

On the other hand, temporal queries (qt) may involve relationships such as adjacency, connectivity, or containment and distributive functions (e.g., count, sum, union) of objects for each timestamp.

We extend the traditional star schema to the spatial domain by operating with standard spatial data types and enabling spatial processing capabilities into the spatial DW. This was accomplished through the utilization of Oracle9i Spatial and the development of a data viewer/manager using the ArcGIS Engine for .Net platform.

Whenever possible we exploit spatial databases native aggregate functions to summarize geometry objects and reduce the complexity of the underlying queries. The main goal was to enhance the decision making process by the addition of spatial representation of data and its patterns and then to implement it using COTS tools. The problem of aggregation query cost at run time is a functional requirement which had also to be considered.

In the following sections we present a case study where decision-makers are mainly interested in spatial OLAP queries to evaluate, for instance, changes in land parcels derived from daily business activity and to efficiently respond to the problematic of estimating refund payments, for instance, caused by wildfires, droughts or any other extreme meteorological conditions.

5.1 Spatial Dimensions

We have adopted Han's definition of spatial dimension [7], whose primitive level and its entire high-level generalized are also spatial. For example, decision-makes may want to analyze refund payments concerning fire incidents, aggregated by county and by wine regions. This requires providing decision-makers the possibility for them to roll-up from vineyard parcels to a more general wine region level.

Notice that a spatial level, such as county, may have more than one way to be generalized to high-level concepts, and the generalized concepts can be spatial, such as map representing larger regions, or non-spatial, such as general description of the wine region [26]. As a consequence a spatial dimension may aggregate different spatial levels. We define a spatial level as a level for which we need to keep its spatial characteristics.

Spatial levels also relate to each other with topological relationships between spatial components, such as contain, equals, intersects, overlaps, etc. [4]. Therefore it is possible to overlap spatial dimensions rows based on the spatial level they describe.

Our spatial dimension modeling approach approximates Kimball's type II dimensions [10] definition because each update is stored as a new record with topological relationships (i.e. tuple versioning). Nevertheless, we need to be careful to avoid overcounting because we may have multiple rows in the spatial dimension for the same spatial feature. We use a most recent row indicator to do counts based on the most up-to-date descriptive values for a spatial object (e.g., vineyard parcel). We also employ effective and expiration dates to deal with spatial features counts at a given historical point in time and make use of two techniques to deal with rapidly changing dimensions. The first one is for browsing and tracking changes of key attributes in changing dimensions. This is accomplished by breaking off one or more minidimensions from the dimension table, each consisting of small clumps of attributes that have been administered to have a limited number of values. This is the case of DVINEYARDS dimension in Figure 2.

The second technique uses variable depth hierarchies. The representation of an arbitrary spatial hierarchy is an inherently difficult task [4, 11, 13] in a relational environment. For example, decision-makers may want to monitor changes to vineyard parcels derived from buy/sell operations. Thus, in Figure 2, the DPARCELS dimension rows can play the role of parent as well as child based on a downward or upward date operation.

To handle with unpredictable hierarchies we inserted a bridge table between the DPARCELS dimension and the fact table. The bridge table contains one row for each spatial object change. Each row contains the parcel key of the parent roll-up entity, the parcel key of the derived entity, the number of levels between the parent and the derived entity, a bottom-most flag that identifies a derived entity with no further nodes beneath it, and a top-most flag to indicate that there are no further nodes above the parent.

The diagram of Figure 2 was built using Perceptory, which is a tool for conceptual modeling and not for implementation modeling therefore no ID attribute is mention because the class stereotype contain already all required attributes for shape and time.

Figure 2 shows a simplified version of the IVV multidimensional spatial data model, in which the tuples of the DPARCELS and DGRAPEGRWERS spatial dimensions may change according to buy/sell operations, wildfires, droughts, or Grape-Grower personal data updates. The tuples of the DGRAPEGRWERS dimension are presented as points on the map.

The number of grafts or grapes determines the vineyard parcel area. The effective vineyard parcel is actually determined by this derived area; therefore it is possible for the same land parcel to be partially classified as a vineyard parcel, which brings into action the need for the multidimensional data model to offer built-in support for partial containment. In this paper we will not discuss the imprecision in dimension hierarchies introduced by the presence of partial containment. In [19] the subject is well discussed.

The data model Figure 2 requires a Time dimension to track for time aggregation query optimization. The temporality pictograms timestamp instances or periods of the entities, whereas the Time dimension represents periods required by decision-makers to analyze the business data.

The DVINEYARDS dimension describes the existing types of vineyards), the DVY_RIGHTS describes the vineyards rights, the DSTATISTICS stores third parties statistical information, the DCOUNTY correspond to administrative boundaries, and the DRegion stores administrative descriptive attributes. Although it could be modeled as a non-spatial dimension, we explicitly present it as a spatial dimension to have the possibility to spatially roll-up history based on current assignments.

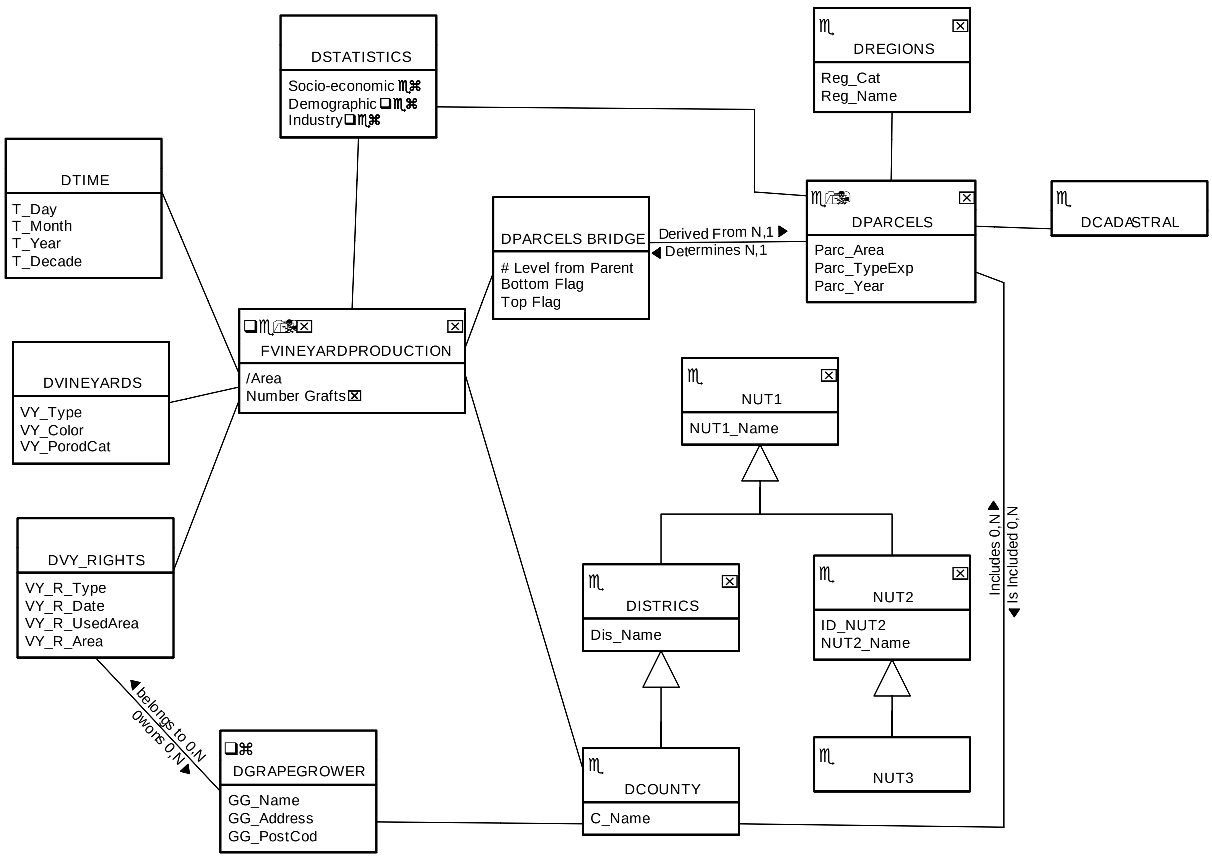

Fig. 2 - A Simplified Version of the IVV Star Schema.

5.2 Spatial Measures

In our approach we distinguish two types of measures:

Numerical measure is a measure containing numerical values. These numerical measures can be further classified into distributive, algebraic, and holistic [13]. A measure is distributive if it can be computed by partitioning the DW, such as count, sum, max; it is algebraic if it can be computed by algebraic manipulation of distributed measures, such as average, standard deviation; otherwise, it is holistic, such as median, most_frequent, rank.

Spatial measure is a measure that either (1) is represented by geometry with a spatial function used for aggregation along the hierarchies, or (2) represents a numerical value that is calculated using spatial or topological operators. For example, during a generalization (roll-up) operation, the regions of the same area might be grouped forming a spatial measure compose of a multi-polygon spatial data type.

Spatial measures can be associated to a fact relationship, independently if the relationship is spatial or not. We had to define how spatial measures should be complex geometries. With a direct impact in spatiotemporal queries and aggregation operations

5.3 Formal Definition of Spatiotemporal Aggregation

Different levels of spatiotemporal granularities can be used to define group, partition, and sliding window composition in a spatiotemporal relation. In spatiotemporal group composition tuples sharing the same spatial and temporal value at granularity G form a collection termed group. For each group, an aggregate function is applied and the group is annotated with the aggregate value. When computing spatiotemporal aggregation using group composition, the resulting relation is a spatiotemporal relation defined at granularity G = GS _ GT (i.e., the cross product of spatial granularity GS and temporal granularity GT).

Consider a land management application that keeps track of the vineyard parcels within a Wine Region. In this application, each stored object is a parcel (spatial extent) that can change with time (temporal extent). Vineyard parcels can change their shape due to natural phenomena such as wildfires or droughts. In addition, vineyard parcels composition may also vary over time because cultivation changes with seasons. For land management applications, we might also be interested in identifying correlations between wildfires and forest density. A useful query in this case will be the following.

5.4 Existing Approaches for Evaluating Spatiotemporal Aggregate Queries

A typical spatiotemporal query specifies spatial and temporal predicates to select tuples of interest. A spatial predicate is defined in terms of a point or an extent, while a temporal predicate can involve a time instant or a time interval [22].

Given a spatial range and a temporal interval, the Algorithms proposed in the literature for the evaluation of spatiotemporal aggregation queries returns summarized information of all the tuples valid during the time interval and that are contained or intersected by the query range. Unfortunately, the evaluation of this type of queries does not result in a spatiotemporal relation. The evaluation of spatiotemporal aggregation queries has only recently caught the attention of the research community.

Within the scope of our research, we adopted the aggregate 3-dimensional RB- tree (a3DRB-tree) [14] index-based approach which groups spatial objects into static regions and index those regions using an R-tree. This decision was taken based on the set of experiments the authors provided to demonstrated the applicability of the algorithm to realistic situations, namely:

- The a3DRB-trees maintains a large 3DR-tree for the whole history, each responsible for a relatively short interval. This fact has implications on the query performance.
- The a3DRB-tree is an off-line structure, meaning that the lifespans of its entries should be known before the structure is created; otherwise, we have to store unbounded boxes inside the 3DR-tree, which affects query performance severely.
- The a3DRB- trees permit temporally disjoint entries to share B-trees in order to save space without compromising query performance.

By keeping pre-aggregate information inside the index, aggregation queries can be answered by intermediate index nodes, thus saving accesses to detailed data. However, one drawback of a3DRB-tree is the distinct counting problem. This problem occurs if a data object remains in the query region for several timestamps during the query interval because such data object will be counted multiple times.

The a3DRB-tree minimises this problem, by combining B-trees with 3DRtrees. Every version of a region is modelled as a 3D box, so that the projection on the temporal axis corresponds to a time interval when the spatial extents of the region are fixed.

6. Case Study

The queries that we present in this section cover three scenarios of spatial OLAP queries: when there are no predefined hierarchies, when there is a need to support multiversioning to cover with geometric evolution parameters and when there is a need to improve spatial OLAP queries performance.

The first example concerns queries involving a fixed spatial dimension, i.e., there is a static set of vector features such as parcels polygons over which simple spatial operations are applied. On the second example, the spatial dimensions may be volatile, i.e., the regions at the finest granularity may evolve over time. Therefore users are mainly interested in understanding the reasons that caused those modifications and/or displacements. In the third example, changes to the geometric shape are seen as spatial measures. Therefore users are mainly interested in grouping the spatial measures to identify patterns or mining for hidden information. This implies the need to correlate the business data with third party data sources, which entails a closer understanding of the business model, for instance, to establish meaningful topological associations with statistical data (e.g., socio-economic, demographic, commercial/industry data) and cadastral information (e.g., land boundaries and subdivisions, parcels of land suitable for transfer of vineyard titles, value and ownership).

These queries bring out the difference between traditional and spatial DW. In the former, OLAP results are often shown as summary tables of text and numbers, whereas in spatial DW the data are organized as collection of maps or geometric shapes.

6.1 Data Model

Figure 3 presents the operational data source diagram used to feed the multidimensional schema of Figure 2. The entities in red model existing map features (spatial data) and the others represent extensions to the schema added in different periods. This detail has raised the complexity to model the spatial DW because of data integrity and integration issues (e.g., spatial data from heterogeneous sources and systems). However, in this paper, we are not going to address integration issues neither the ETL (Extract, Transform and Load) process.

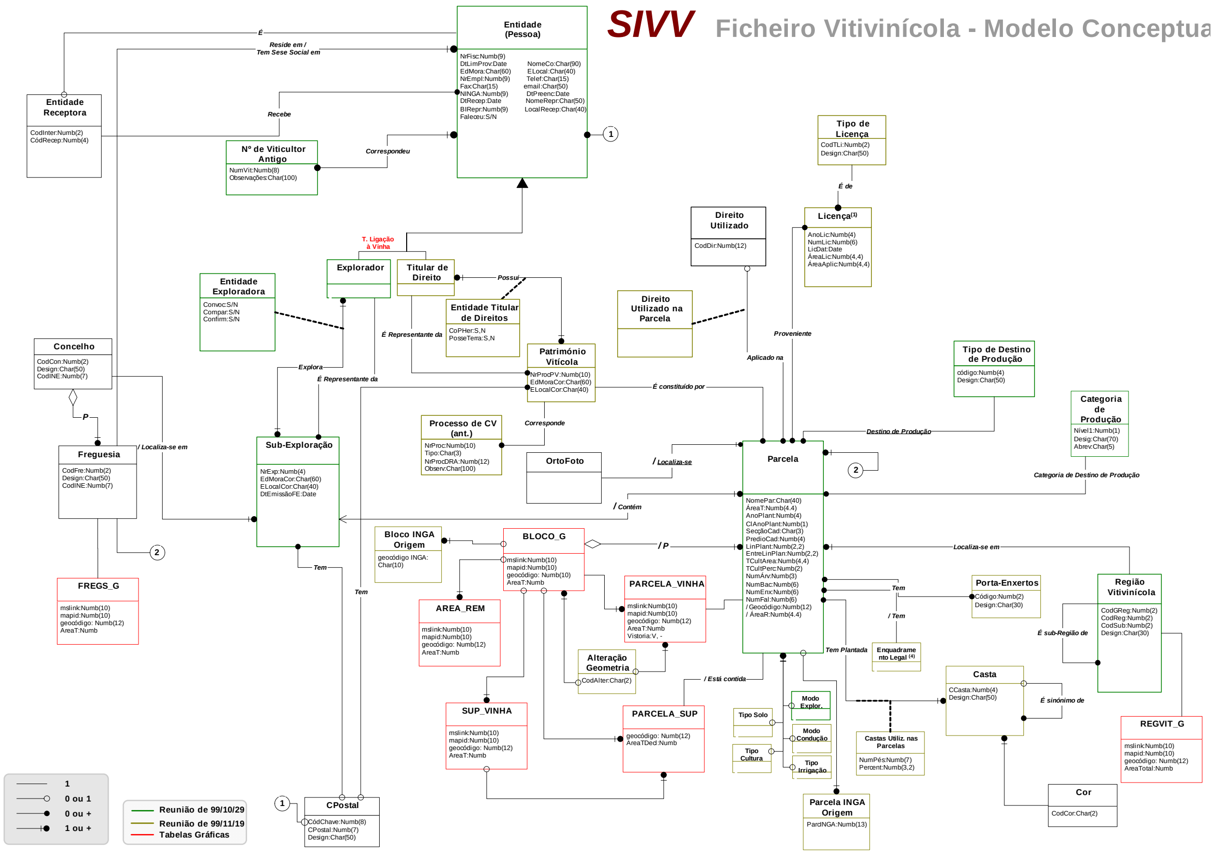

Fig. 3 – The IVV Operational Data Conceptual Model.

In our approach all geometric description of a spatial object was stored in a single row, in a single column of object type SDO_GEOMETRY [25], which correspond to a vector feature to be used to filter, sort, and group the query results. On the other hand, when the geometric shape is susceptible to be the subject of the spatial analyses it is computed as a spatial measure and stored in the fact table.

It is also expected that decision-makers may need to drill-around (zoom-in and zoom-out using a GIS nomenclature) dimensions hierarchy, requiring the overlay of multiple thematic maps and causing spatial data to be expressed at different granularities. Decision-makers may want to backtrack parcels changes, for instance, to monitor/validate parcels derived from split or merge operations. In this domain the Oracle9i Spatial object-relational conformity to spatial SQL standards [25] together with the Perceptory possibility to generate code to that target system helped in the implementation/validation phase. Unfortunately all the operations described into Perceptory weren't transferred by the code generator, because Perceptory allows only a way to conceptually describe them.

In our approach we include additional attributes to support GPSJ queries, typical of OLAP applications: essentially, queries consisting of a selection and an aggregation operated over a join. Table 2 outlines the groups of attributes that were considered: cartographic parameters, administrative attributes and descriptive business attributes.

The key factors which impact the optimization benefit for a GPSJ query are its aggregation level (defined by is grouping set) and its selectivity (defined by HAVING/WHERE clause). Thus the two existing techniques to diminished GPSJ execution cost are materialization or indexing.

Materialization offer great advantage for queries with coarse aggregation, which compute a few groups out of a huge number of tuples, since accessing a small view is much cheaper than accessing a huge table. Indexes are best in solving queries with high selectivity, which select only a few tuples, since accessing lots of useless tuples will be avoided. Therefore, GPSJ queries with coarse aggregation and low selectivity encourage materialization while queries with fine aggregation and high selectivity encourage indexing.

However, as sketched in Figure 4, it is difficult to predict which of the two optimization techniques will fit best for queries falling outside the identified regions (e.g., to accomplish with ad-hoc spatial OLAP queries).

Table 2 - Spatial Dimension Auxiliary Attributes.

Metadata		
Cartographic Parameters	**Administrative Attributes**	**Descriptive Business Attributes**
Scale Fraction, i.e., representative fraction within which the spatial object is valid.	Feature ID + Feature Name	These sets of descriptive attributes correspond to specific business rules, namely the spacing between grafts.
Map Units, Indicates the map measure unit	Feature Type (Polygon, Line, Point), helps to specifies the dimensionality of the spatial data, including multipoint, multiline, multipolygon or 3D objects	
Coordinate system (geographical or projected)	Feature Category, defines business data nomenclature to help in the ordering/classification of spatial data	
Projection type	Feature Status and Version, useful to simplify query performance and for multiversion proposes	
Projection Decription	Flag, a most recent row indicator very helpful to do counts based on the most up-to-date descriptive values	

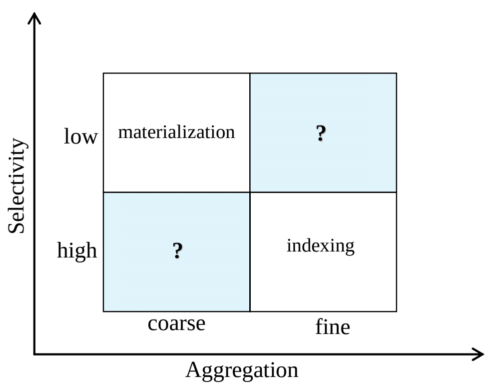

Fig. 4 – Optimization techniques based on the query selectivity and aggregation level.

Moreover, to operate within a spatial OLAP context we have to consider partial containment constrains. While dimension values in conventional multidimensional data models either are disjoint or exhibit total containment relationships, partial containment is prevalent in spatial data.

Partial containment, together with its transitivity property, introduces additional imprecision in aggregation paths. Thus algorithms for making dimension hierarchies onto, covering, and aggregation strict are required. However, current multidimensional database technology does not support the complex structures needed to handle spatial information.

Rather than proposing an entirely new multidimensional data model and query language, we choose to search for adequate proposals in the literature the Oracle9i Spatial configuration functionalities. This approach enables the use of practical (existing) preaggregation methods while still preserving the information about partial containment.

Furthermore, it is important that we can preaggregate values at any combination of dimension levels and reuse the preaggregated values to compute higher level aggregate results (summarizable results).

Thus, having evaluated the problematic associate to partial containment of spatial OLAP queries table 3 describes briefly the requirements for a multidimensional data model to support spatial data ad-hoc queries.

6.2 Evaluation

The IVV decision-makers usually need to monitor vineyard parcels' boundaries evolution. This type of information requires a multiversioning search, usually expressed on the following form:

- *What was the evolution in vineyards parcels derived from new or replanting rights being applied over the last decade?*

This query operates directly over DParcels, requiring only a multi-tree index structure to access the whole history (last decade) of geometry shapes which overlap the search region. This is particularly true for long periods covering a large geographic area.

On the other hand if decision-makers intend to monitor incidents caused by fire they may use a map-based layout to visually backtrack for patterns. A typical query would be:

- *What was the evolution of burned areas on vineyard plantation from 1990 to 2000 at Alentejo and Terras do Sado? Where did it start and through where did it spread?*

Although this query could be answered by using SQL-92 statements, it would be very expensive to check and enforce. Furthermore, topological relationship among spatial objects would be neglected making it difficult to cross-examine spatial and non-spatial data over maps and use spatial functionalities to seamlessly discover geographical patterns.

Table 3 – Requirements for a multidimensional data model with spatial data.

Data Model Requirements	Description
Explicit and multiple hierarchies in dimensions	Explicit hierarchies are useful in data analysis because they aggregate data to the right level of detail for roll-up/drill-down operations efficiency Support for multiple hierarchies means that multiple aggregation paths are possible. These are important because multiple hierarchies exist naturally in much data and to handle imprecision in queries.
Partial containment	A multidimensional data model should provide built-in support for dimensions with partial containment relationships (e.g., districts would, though approximately, roll up to cities).
Nonnormalized hierarchies	Situations occur where a hierarchy value has more than one parent, a value has no relationship to any value in the category immediately above it in the dimension hierarchy, or a value has no relationship to any value in any category below it (e.g., land parcel value may be related to several district parent values, or a city value may have no cell child values.
Different levels of granularity	Support for different levels of granularity enables a request to refer to other values than those in the category at the lowest level of a dimension hierarchy
Many-to-many relationships between facts and dimensions	This requirement implies that a fact may be related to more than one value in a dimension
Handling of imprecision	When facts are characterized by dimension values from different levels, imprecision in the data occurs because data for a query is missing or the transitive relationships between members of aggregation paths may become imprecise.

6.3 Implementation

The multidimensional spatial data model was implemented on Oracle9i Spatial. We use the spatial data types provided by the Oracle SDO table structure.

Each change to spatial data in vector format (e.g., geometry shape changes) is stored as an SDO_GEOMETRY [25] type in a single row. In this version raster data (e.g., satellite imagery, scanned topographical maps, or DTM grid data) are stored in the DBMS as descriptive attributes.

Once the database is implemented, the access to the stored data is not restricted to a certain GIS-package (like ArcGIS in our case) but is obtainable via any Oracle compatible system (e.g. GeoMedia, MapInfo, ArcView). We use ArcGIS 9 .Net as the development platform because it offers a wide and flexible range of capabilities for a systematic analysis of the available data.

We opted for COTS applications because they present some advantages over prototypes that are developed from scratch, namely reliability and features. In addition, COTS applications constitute an excellent starting point to expand software capabilities and improve system performance for the commercial marketplace. Consequently, prototypes based on commercial applications can usually be utilized to solve real-world problems, such as the IVV case study. At the time of publishing this paper, the first commercial Spatial OLAP running on top of COTS should be out and further save months of development (cf. 3rd author).

7. Conclusion

In this paper we presented a practical example in the field of land parcels, to evaluate the implementation of a spatial multidimensional model. The extension to the spatial domain provides a concise and organized representation of spatial phenomena [19] that facilitates the delivery of data for spatial OLAP systems, spatial data mining, or spatial statistical analysis.

A conceptual multidimensional model establishes a communication bridge between users and designers. It reduces the difficulties of spatial data handling as decision-makers do not usually possess the expertise required by software currently used for managing spatial data.

Concurrently, we are directing our research towards aspects such as object roles, constrains, data quality, SOLAP functionalities and spatial ETL (extract, transform, load) for multi-granularity spatial DW. When modeling spatial data these are aspects that have to be considered for the spatial properties, to avoid inconsistency and ambiguity. This concern has been neglected insofar.

Moreover, when considering interoperability of spatial data, we need to have a modeling notation (in our case Perceptory) compatible with today's' standards.

8. References

[1] D. Zhang and V. Tsotras, "Improving Min/Max aggregation over Spatial Objects", *In Proc. GIS*, 2001.

[2] F. Rao et al, "Spatial Hierarchy and OLAP-Favored Search in Spatial Data Warehouse". *In Proc. DOLAP*, 2003.

[3] M.L. Gonzales, "Seeking Spatial Intelligence," *Intelligent Enterprise Magazine*, 3(2), 2000.

[4] E. Malinowski and E. Zimányi, "Representing Spatiality in a Conceptual Multidimensional Model," *In Proc.GIS*, 2004.

[5] S. Rizzi, E. Saltarelli, "View materialization vs. indexing: balancing space constraints in data warehouse design", In Proc. CAISE, Velden, Austria, pp. 502-519, 2003.

[6] S. Shekhar and S. Chawla, *Spatial Databases: A tour*, Prentice Hall, 2003.

[7] J. Han and M. Kamber, *Data Mining: concepts and Techniques*, Morgan Kaufmann Publisher, Inc, 2001.

[8] Tao and D. Papadias, "Historical Spatiotemporal Aggregation," *In Proc. ICDE*, 2002.

[9] Y. Bedard et al, "Modeling Geospatial Databases with Plug-Ins for Visual Languages" ER2004, LNCS 3289, 17–30, 2004.

[10] R. Kimball, R. Merz and M. Ross, *The Data Warehouse Toolkit*, 2nd ed., 2002.

[11] P.A. Longley, M.F. Goodchild, D.J. Magurie, D.W. Rhind., *Geographical Information Systems, Principals and Technical Issues,* 2nd ed. Wiley & Sons, 2002.

[12] P. Rigaux, M. School and A.Voisard, *Spatial Databases with Application to GIS*, Morgan Kaufman 2002.

[13] N. Stefanovic et al, "Object-Based Selective Materialization for Efficient Implementation of Spatial Data Cubes," *IEEE Transactions on Knowleadge and Data Engineering, vol. 12*, nº. 6, p. 938-958, 2000.

[14] D. Papadias et al "Indexing Spatiotemporal Data Warehouse," 2002.

[15] P. Marchand et al, "Implementation and evaluation of a hypercube-based method for spatio-temporal exploration and analysis", Int. Soc. of Photogrammetry and Remote Sensing Journal, 59:1, 6-20, 2004.

[16] M. Indulska, M. Orlowska, "On Aggregation Issues in Spatial Data Management," *In Proc. 13th Australasian Database Conference*, p. 75-84, 2002.

[17] G. Pestana et al "A Prototype Implementation of a Spatial Data Warehouse for Integrating Business, His-torical and Spatial Data," *5th Int. Conf. of Intelligent Data Engineering and Automated Learning*, 2004.

[18] Y. Bedard et al, "Integrating GIS Components with Knowledge Discovery Technology for Environmental Health Decision Support", Int. J. of Medical Informatics, 70:1, 79-94, 2002,

[19] C.S. Jensen et al,"Multidimensional data modeling for location-based services" *VLDB Journal*, 13:1-21, 2004

[20] E. Camossi, M. Bertolotto, E. Bertino and G. Guerrini, "A Multigranular Spatiotemporal Data Model," *In Proc. ACM-GIS*, 2003.

[21] M. Body et al, "Handling Evolutions in Multidimensional Structures" IEEE 19th Int. Conf. on Data Engi-neering (ICDE), March 5-8, Bangalore, India, 2003.

[22] I. Lopez et al, "Spatiotemporal Aggregate Computation: A Surve, A TIMECENTER Technical Report, 2004.

[23] Open GIS Consortium, OpenGIS Specifications, 2003. http://www.opengis.org/ogcSpecs.htm.

[24] ISO TC211, 2003. http://www.isotc211.org

[25] C. Murray, "Oracle Spatial User's Guide and Reference," Release 10.1 (B10826-01), Oracle, 2003.

[26] J. Brodeur, et al, "Modelling Geospatial Application Database using UML-based Repositories Aligned with International Standards in Geomatics", ACMGIS 2000, Nov. 10-11, 2000.