Providing Multidimensional and Geographical Integration Based on a GDW and Metamodels

Robson do Nascimento Fidalgo, Valéria Cesário Times, Joel da Silva,
Fernando da Fonseca de Souza, Ana Carolina Salgado

Center for Informatics, Federal University of Pernambuco, P.O. Box 7851, Recife-PE, Brazil, 50.732-970
{rdnf,vct,js,fdfd,acs}@cin.ufpe.br

Abstract. Data Warehouse (DW), On-Line Analytical Processing (OLAP) and Geographical Information System (GIS) are tools for providing decision-making support. Much research is aimed at integrating these technologies, but there are still some open questions, particularly regarding to the design of spatial dimensional schemas of Geographical DW (GDW) and the models for providing a semantic integration among the metadata of such technologies. To perform this task we are based on the Geographical On-Line Analytical Processing Architecture (GOLAPA), to propose 1) the framework GeoDWFrame, which is based on the star schema and has been specified as guidance for designing geographical dimensional schemas and 2) a pair of MOF based metamodels, namely: Geographical Analytical Metamodel (GAM) and Geographical Multidimensional Metamodel (GeoMDM), which, respectively, provide the modelling of analytical and spatial applications and the integration of the CWM OLAP and GAM metamodels. We have also developed a case study, where the GeoDWFrame and these metamodels are applied.

Categories and Subject Descriptors: Information Systems [**Miscellaneous**]: Databases

Keywords: DW, OLAP, GIS

1. INTRODUCTION

Data Warehouse (DW) [Inmon 2002; Kimball 1996; Kimball et al. 1998; Chaudhuri and Dayal 1997], is a dimensional database that is organised over two types of tables: 1) dimensions - addresses descriptive data and 2) facts - addresses measuring data. Online Analytical Processing (OLAP) [Chaudhuri and Dayal 1997; OMG 2001] is a specific software category for providing strategic and multidimensional queries over the DW data, where its results are interpreted in multidimensional structures (hypercubes). These multidimensional structures support the following operations: Drill-up or Roll-up - generalization or aggregation of data, Drill-down - specialization or desegregation of data, Slice and dice - selection and/or projection of data and Pivoting - data reorientation. Geographical Information System (GIS) [Bernhardsen 1999; Demers 2002] is a system that helps in acquiring, manipulating, examining and presenting geographical features. Each one of these features is composed of its descriptive information (conventional data) and its geographical reference (geometrical data). These features are viewed as maps that can be combined, one above the other, in order to provide layers of geo-information. These layers are frequently known as themes, or coverages, and have basically two types of data format: vector or raster.

Concerning the integration among DW, OLAP and GIS, much research has already been done [Stefanovic et al. 2000; Kouba et al. 2000; Miksovský and Kouba 2001; Bedard et al. 2001; Rivest et al. 2001; Ferreira et al. 2001; Ferreira 2002; Shekhar et al. 2001; Papadias et al. 2001; Papadias et al. 2001; Rao et al. 2003]. However, some of these approaches do not deal with open and extensible solutions and there are still some open questions, particularly regarding to the design of the spatial dimensional schemas of Geographical DW (GDW) and to metamodels for providing a semantic integration among the metadata of such technologies. In order to address the first problem, the Geographical On-Line Analytical Processing Architecture (GOLAPA) [Fidalgo et al. 2001; Fidalgo 2003; GOLAPA 2004] was proposed. Regarding the second problem, we present the framework GeoDWFrame [Fidalgo et al. 2004], which aims to guide the design of dimensional schemas of GDW. In order to address the third problem, the Geographical Analytical Metamodel (GAM) and the Geographical Multidimensional Metamodel (GeoMDM) are discussed. GAM is used to describe logical models of analytical and geographical applications while GeoMDM provides the semantic integration between OLAP metadata of Common Warehouse Metamodel (CWM) [OMG 2001] and GAM metadata.

Aiming to be as open and extensible as possible GAM and GeoMDM are based on Meta Object Facility (MOF) [OMG 2002a] and eXtensible Mark-up Language (XML) [W3C 2004], more specifically follow XML Metadata Interchange (XMI) [OMG 2002b] and Java Metadata Interface (JMI) [SUN 2002]. In turn, GeoDWFrame defines that the geometries of the geographical Features, preferentially, following the Geography Mark-up Language (GML) [OGC 2002a]. MOF is the standard of Object Management Group (OMG) to provide interoperability of metadata among tools, systems and repositories. XMI and JMI are MOF standards for XML and Java, respectively. CWM OLAP is the OMG MOF standard for DW and OLAP tools. GML is the Open GIS Consortium (OGC) standard to provide interchange and storage of geographical Features in XML.

The remainder of this paper is organized as follows. Section 2 discusses some related work. Sections 3, 4 and 5 propose GeoDWFrame, GAM and GeoMDM, respectively. Then, a case study applying our proposals is given in section 6 and finally section 7 presents our conclusions and future work.

2. RELATED WORK

Some research [Stefanovic et al. 2000; Kouba et al. 2000; Miksovský and Kouba 2001; Bedard et al. 2001; Ferreira et al. 2001; Ferreira 2002; Shekhar et al. 2001; Papadias et al. 2001; Papadias et al. 2001; Rao et al. 2003] address the use of DW with geographical data, but some points still need to be further discussed, especially the design of its dimensional schema. However, these research tend to converge in one point: that the GDW schema is an extension of a star schema. Nevertheless, most of these research do not propose a true GDW, because in some cases [Kouba et al. 2000; Miksovský and Kouba 2001; Ferreira et al. 2001; Ferreira 2002] the descriptive data are handled in DW while the spatial data are handled in GIS, which stores its data in proprietary files (e.g. .SHP and .MIF).

From previous research, the work of Han et al. [Stefanovic et al. 2000; Bedard et al. 2001] is the more relevant, because this also proposes a framework to address a true GDW. The framework of Han et al. is based on a star schema and considers three types of dimensions and two types of measures. The dimensions are: non-geometrical (all levels contain only descriptive data), geometrical-to-non-geometrical (the finest granularity level contains only geometrical data and the others only descriptive data) and fully-geometrical (all levels contain only geometrical data). With regard to the measures, these are: numerical (only additive data) and spatial (a collection of pointers to spatial objects).

Concerning the research of Han et al., there are four issues that need to be discussed more. These motivated the proposal of the GeoDWFrame. The first debates the use of spatial measures. In any DW, a measure is a quantitative value and not a collection of pointers. For this, we understand that a spatial measure will be defined as a textual field, which needs first to be parsed for just afterwards to be processed, probably increasing the cost and complexity of computing. Another point about spatial measures is: if spatial dimensions already address the spatial data, why not process this data to achieve an equivalent result to spatial measure. This would be more natural, because dimension intrinsically deals with textual fields and these data would not need to be parsed. The second ponders the intrinsic redundancy of dimensional data [Inmon 2002; Kimball 1996; Kimball et al. 1998] and its impact over the high cost of storing geometrical data in common dimensions. The GDW model in discussion does not present an approach to minimize this issue, but a GDW can have more than one dimension with the same spatial levels (e.g. the Store and Customer dimensions have the same levels: Countries, Regions, States and Cities) or even if a GDW has just one dimension with spatial levels, there will still be the intrinsic dimensional redundancy of geometrical data to be addressed. The third argues that a geometrical-to-non-geometrical dimension just allows geometrical data in the finest granularity level. This fact may limit its use, because in practice, it is relevant that a GDW can permit geometrical data at any level. Otherwise, just the finest granularity level could be drawn on the map. And finally, the fourth issue deals with where the descriptive data of geographical objects are. These are useful for OLAP queries, but the dimensions in discussion just handle geometrical data.

Still about the work presented at the beginning of this section, just GOAL [Kouba et al. 2000; Miksovský and Kouba 2001] and SIGOLAP [Ferreira et al. 2001; Ferreira 2002] projects support the integration of multidimensional and geographical data by dealing with metadata. These work are briefly discussed below. In the GOAL architecture, the main component used is called Integration Module (IM). It is also built around metadata and aims to support the OLAP and GIS integration. However, the integration metamodel is not presented and the architecture is developed based on format dependent solutions (e.g Avenue scripts - the native language of ArcView and OLE DB for OLAP - the Microsoft OLAP API). In SIGOLAP, its architecture is based on a mediation approach, where metadata plays an important role in the OLAP and GIS integration. The integration metamodel is based on CWM OLAP and GeoFrame [Filho and Iochpe 1999], but just provides the integration at data level (e.g. codLocation of a OLAP Dimension to codLocation of a GIS Theme). The semantic integration among multidimensional and geographical metadata is not addressed. The SIGOLAP architecture is also developed based on format dependent solutions (e.g Visual Basic and Visual Basic Script).

3. THE FRAMEWORK GEODWFRAME

Kimball [Kimball 1996; Kimball et al. 1998] defined several techniques and guidelines to build a conventional DW. From Kimball's work and the issues of Han's work (pointed out in section 2), we present the framework GeoDWFrame. With regard to the four issues of Han's work, we decided that GeoDWFrame, respectively: 1) does not apply spatial measures, 2) normalizes the geometrical data, 3) provides geographical data in any dimensional level and 4) stores the descriptive data of geographical features. In order to provide support for these issues, GeoDWFrame proposes two types of dimensions, namely: geographical and hybrid. The first one is classified into primitive and composed, while the second one is classified into micro, macro and joint. The primitive and composed dimensions have at all levels (or fields) just geographical data (e.g. client addresses and its geo-references), while the others deal with geographical and conventional data (e.g. client addresses and its geo-references plus ages and genders). Besides these dimensions, the GeoDWFrame, as in any DW, also supports dimensions with just conventional data (e.g. a product dimension). In the sequence, the differences between primitive and composed dimensions are given.

A primitive geographical dimension represents the geo-coding data (geometries) used to handle (e.g. draw, query and index) spatial objects. Basically, it can be implemented using two approaches: 1) a relational database with long field (e.g. Text, Long or BLOB) or 2) a pos-relational database that supports a geometry abstract type (e.g. Oracle SDO_GEOMETRY or DB2 ST_GEOMETRY). With regard to the first approach, it allows the storage of geo-coding data in a textual format, but the spatial processing (e.g. query and indexing) is performed by geoprocessing software and not by the database system. Regarding the second approach, the database system performs the spatial processing, but the spatial extensions have not been standardized yet (e.g. divergence of spatial operators and spatial data types), which results in a proprietary solution.

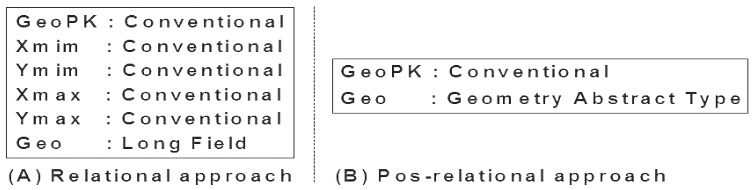

Fig. 1. Examples of how representing a primitive geographical dimension

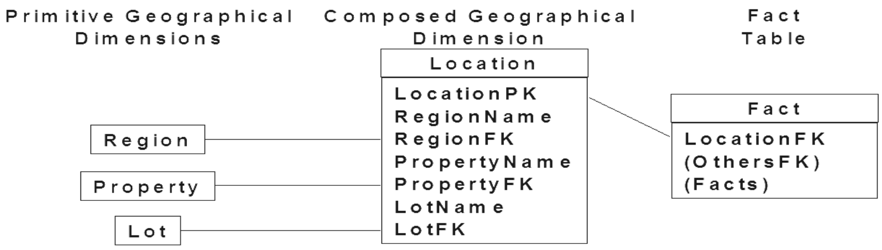

Fig. 2. An example of a composed geographical dimension

Figure 1 shows an example of how the last two approaches can model polygons. In the Figure 1(A), the fields GeoPK, Xmim, Ymim, Xmax and Ymax are conventional (e.g. Varchar2 or Number) while the Geo is a long field. In Figure 1(B), the GeoPK field is also conventional, but the Geo is a geometry abstract type. These fields represent the identification of an object (GeoPK), its Minimum Bounding Rectangle (MBR) (Xmim, Ymim, Xmax and Ymax) and its geo-coding (Geo). Realize that a primitive geographical dimension just deals with geometrical data. For this, it is not appropriate to OLAP queries. However, this is essential to: 1) support geo-operations, 2) draw the features and 3) keep the historic of the geo-objects applying slowly changing dimensions [Kimball et al. 1998; Chaudhuri and Dayal 1997].

The quantity of fields in a primitive geographical dimension may change according to: 1) the type of the spatial object (e.g. a point does not need a MBR), 2) the geo-graphical software that will be used (e.g. TerraLib[TerraLib 2004] needs more fields to handle multi-polygons) and 3) the chosen approach (relational or pos-relational). However, basically we focus on two fields: one to store the identification of the spatial object (GeoPK) and another to store its geographical representation (Geo). In GOLAPA, in order to be as open and extensible as possible, the primitive geographical dimension is implemented using the relational approach with the GML [OGC 2002a]. Note, in the future, when the spatial extensions of pos-relational database are completely standardized, these will be able to be used as well. Concerning the composed geographical dimension, it contains fields to represent its primary key, the description of the geo-objects and its foreign keys to primitive geographical dimensions.

Figure 2 shows an example of a composed geographical dimension. In this figure, the geographical hierarchy is known a priori (Region'Property'Lot), which allows it to aggregate the levels of this hierarchy using conventional methods [Harinarayan et al. 1996]. However, when the geographical hierarchy is unknown in advance (at design time), it is necessary, during the GDW building process, to apply spatial indexing methods [Guttman 1984] to define the order existing among the geographical objects. Realize that without the primitive geographical dimensions a dimension with spatial data would have to store their descriptions and geo-references together. Then due to the: 1) intrinsic redundancy of dimensional data and 2) high costs of storing the geo-references (compared to foreign keys to primitive dimension), this approach would result in a high cost of storage.

With regard to hybrid dimensions, that is, micro, macro and joint, these are described as follows.

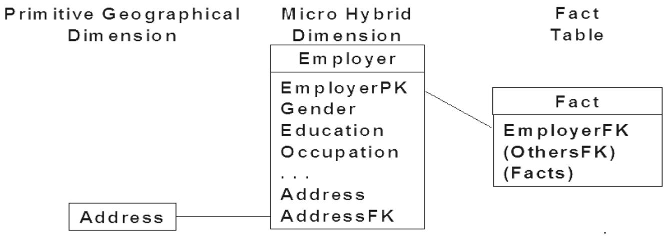

Fig. 3. An example of a micro hybrid dimension

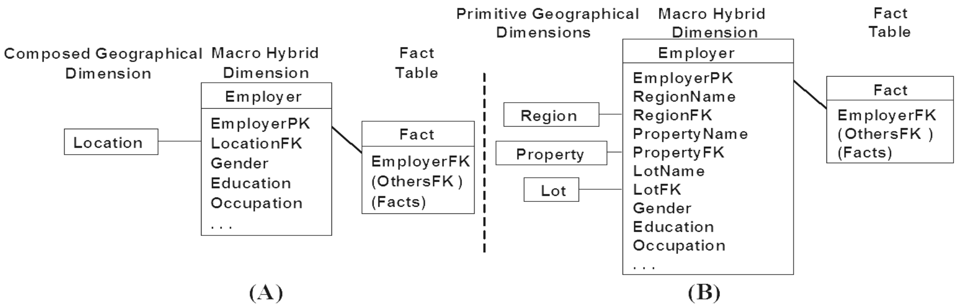

Fig. 4. Examples of a macro hybrid dimension

A micro hybrid dimension handles conventional and geographical data, but the geographical ones represent the finest geo-granularity of the dimension, which must represent points-objects (e.g. Addresses and Lots). Note that due to the small spatial granularity, other dimensions rarely share these geo-referenced data. However, if this is the case (e.g. in the dimensions Customer and Employee, we can have an employee who is also a customer), we have to analyse the context and decide whether duplicate or share the geographical data. The schema of a micro hybrid dimension has conventional fields and two geographical fields (the descriptive and the foreign key for its primitive geographical dimension). Figure 4 gives an example of a micro hybrid dimension.

A macro hybrid dimension, different to a micro hybrid dimension, handles geographical data that are usually shared (e.g. Countries, Regions, States and Cities). The schema of a macro hybrid dimension can be defined based on two approaches: 1) conventional fields plus one foreign key to a composed geographical dimension or 2) conventional fields plus one foreign key to each primitive geographical dimension that corresponds to each descriptive geographical field. Note that in the first one, the composed geographical dimension minimizes the number of foreign keys in the macro hybrid dimension and also can work as a mini-dimension or a role-playing dimension [Kimball 1996; Kimball et al. 1998]. Figures 4(A) and 4(B) respectively illustrated these approaches. In these figures, we assume Lot as the finest geo-granularity, for this, its geo-data must be defined as points-objects. However, in Figure 5, we assume Region, Property and Lot as polygons-objects and the Address as the finest geo-granularity, that is, points-objects. A joint hybrid dimension merges the micro and macro approaches in just one dimension. Figure 5 draws an example of this dimension. A case study applying GeoDWFrame is presented in section 6.

4. THE GEOGRAPHICAL ANALYTICAL METAMODEL (GAM)

CWM OLAP is the standard metamodel for multidimensional tools. However, geographical applications still need a similar metamodel, and as a result, the integration among these tools may be seen as a non-trivial task. In order to address this, we understand that a geographical system needs special ca-

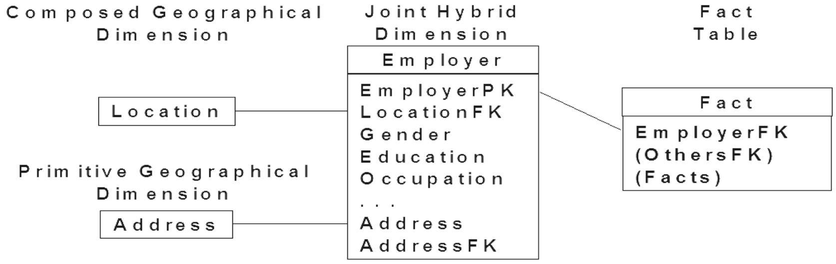

Fig. 5. An example of a joint hybrid dimension

pabilities, which we propose as belonging to an Analytical Geographical Information Service (AGIS). A detailed study about which capabilities an AGIS should own still needs to be accomplished. However, based on the classical DW definition [Inmon 2002], we define some basic capabilities: 1) integrated - its data can come from heterogeneous sources, but this cannot own neither syntactic heterogeneity nor semantic heterogeneity; 2) non-volatile - its operations must be limited to data load and query (an update can just be processed if a wrong data load occurs); 3) spatio-temporal - its operations must support spatial and temporal queries over GDW data; 4) geographical order- its data must be ranked aiming to support a geographical hierarchy. Realize that the last capability is important to address the OLAP drill operations in the AGIS environment. Concerning these capabilities, the first, second and third can be provided by a GDW, but the fourth requires a geographical metamodel that supports it. In order to address this issue, we propose the GAM, which follows the metadata proposals of the OGC [OGC 2001b; 2001c; 2003] and ISO [ISO 2002b; 2000; 2002a] and is based on the CWM OLAP metamodel [OMG 2001] to support the fourth capability and also achieve a symmetric metamodel that can be more naturally integrated with the CWM OLAP metamodel. Figure 6 depicts the GAM design view in UML [Larman 2001].

In figure 6, a Feature type (FeatureType) corresponds to the metadata in the smallest abstraction level, while the others incrementally represent higher abstraction levels. The Feature type corresponds to the same metadata defined by the OGC [OGC 2001b; 2001c; 2003] and the ISO [ISO 2002b; 2000; 2002a] specifications, which assures that applications based on GAM also follow ISO and OGC proposals. The geographical view (GeoView) is an abstract super-class that supports the geographical order capability of an AGIS. In GAM, a geographical view behaves similar to a relational view [Date 2000] and is based on the concepts of both OLAP hierarchies [OMG 2001] and OGC [OGC 2001b] - ISO [ISO 2002a] geometry basic classes.

A geographical view has two subclass, namely: primitive geographical view (PrimitiveGeoView) and aggregate geographical view (AggregateGeoView), which, respectively, merge the concepts of: 1) a hierarchy based on levels with an ordered set (ordered) of disjoint primitive Features subsets (GM_Primitive [1] and GM_Complex[2] [OGC 2001b; ISO 2002a]) and 2) a hierarchy based on value with a ordered set (ordered) of weak aggregations (decomposable) among Features (GM_Aggregate[3] [OGC 2001b; ISO 2002a]). Specifically, a primitive geographical view is formed by Feature types that are composed just by primitive Features (primitiveFT), which can take part in other primitive geographical views. Regarding an aggregate geographical view, it is formed by a single Feature type that is composed by aggregate Features (aggregateFT), which are indexed [Guttman 1984] aiming to achieve a linked node (parent-child) hierarchy [OMG 2001]. An aggregate geographical view, due to its peculiarities, can be composed of only one aggregate Feature type. Figure 7 compares a primitive geographical view and an aggregate geographical view. Realize that in a primitive geographical view

[1]A single (non-decomposed), connected, homogeneous geo-object in a coordinate system
[2]A set of disjoint primitive Features in a common coordinate system
[3]A Feature that gather, in a common coordinate system, other Features

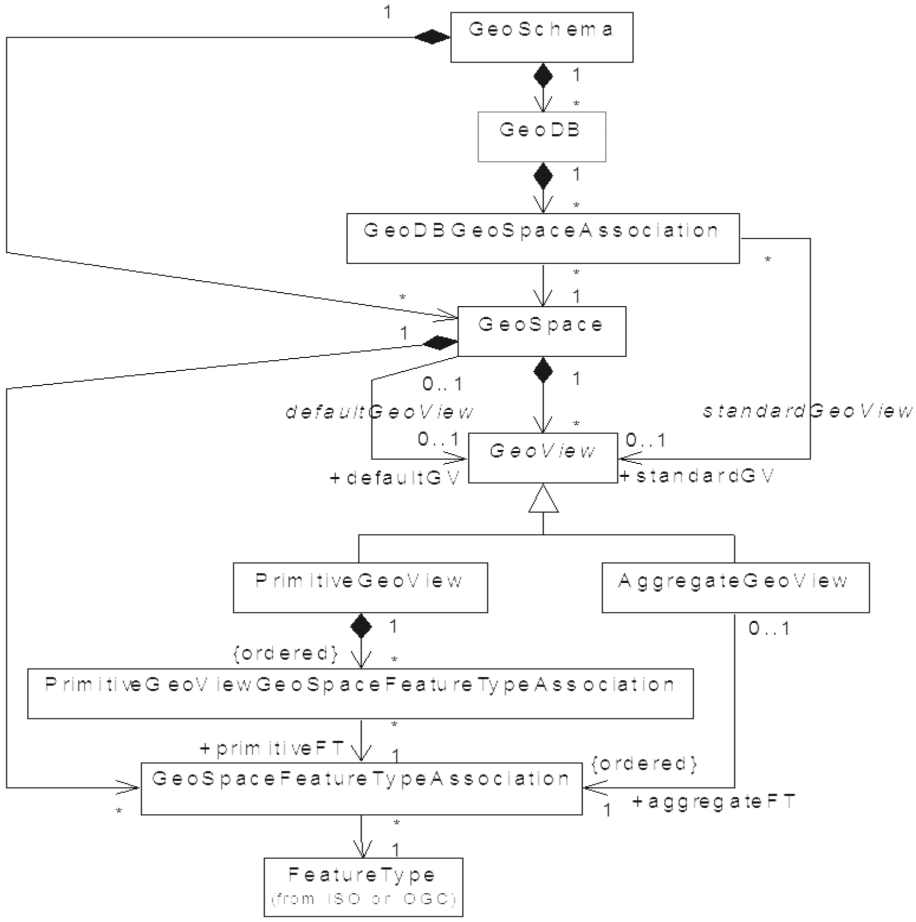

Fig. 6. GAM design view

the set of Feature types is well delimited and the geographical order is defined according to the natural position of the Feature types (e.g Continents ' Countries ' States ' Cities). However, in an aggregate geographical view, there is just one Feature type (it is a heterogeneous collection of geometric objects), where the geographical order is arranged according to a spatial index structure, which defines the order among the Features (not among the Feature Types).

A geographical space (GeoSpace) represents an analysis region that gathers a collection of Feature types, which may be shared among other geographical spaces. A geographical space can be examined from different geographical views, which compose a unique geographical space. A geographical database (GeoDB) corresponds to the Dataset level of the OGC [OGC 2001c] and ISO [ISO 2002b] metadata hierarchy and is formed by a set of geographical space, which may be shared among other geographical database. A geographical schema (GeoSchema) corresponds to the Dataset series level of the OGC and ISO metadata hierarchy and represents a logical container of a set of geographical databases and geographical spaces. Its main purposes are to impose high-level authorization policies (constraints) and mark the entry point for navigating GAM models.

The remainder metadata correspond to association metadata, namely: GeoSpaceFeatureTypeAssociation - it is a subset of geographical data (primitive or aggregate) that composes a geographical space and is used to define a geographical view. It corresponds to a reference to the Feature type of OGC [OGC 2001b; 2001c; 2003] and ISO [ISO 2002b; 2000; 2002a] specifications. Primitive-

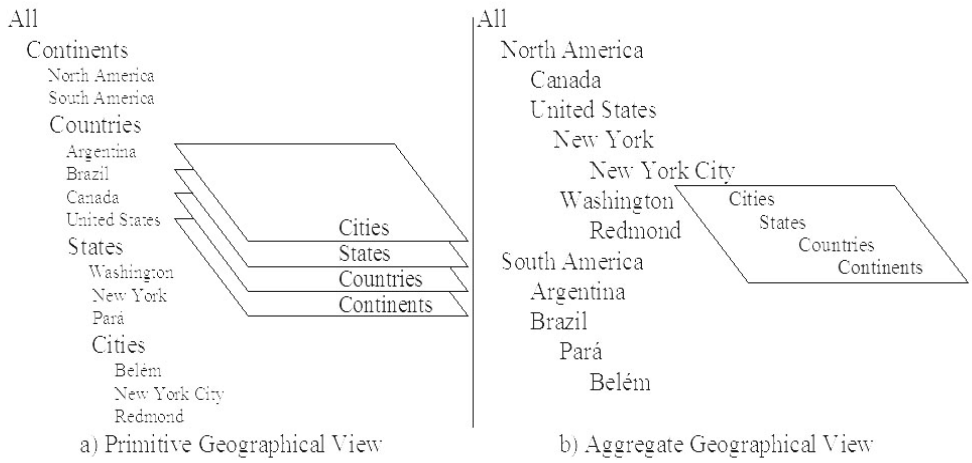

Fig. 7. Primitive geographical view vs. aggregate geographical view

GeoViewGeoSpaceFeatureTypeAssiciation - it represents a Feature type that composes a primitive geographical view. GeoDBGeoSpaceAssociation - it constitutes a geographical space that composes a geographical database. To achieve a symmetrical metamodel to CWM OLAP, GAM has a default geographical view (defaulGeoView) and a standard geographical view (standardGeoView) as optional associations, which, respectively, correspond to the displayDefault and calcHierarchy associations of CWM OLAP. That is, in first case (if defined), it represents a geographical view (defaultGV) of a GeoSpace metadata and in the second case (if defined), it represents a geographical view (standardGV) of a GeoDBGeoSpaceAssociation metadata. To not overload the GAM development with the storage of bi-directional references, its associations are unidirectional [Larman 2001].

In figure 6, similar to CWM OLAP metamodel, all metadata have their references attributes, but just the metadata that does not provide association (not finished with the word Association) owns the name and description attributes. However, in order not to impair the diagram legibility, we decided not to present them. In addition, a geographical space and a geographical view also have the attributes X_min, Y_min, X_max and Y_max that define its MBR [4] . Depending on the application, other attributes can be added. Preferentially, these must follow the OGC [OGC 2001c] or ISO [ISO 2002b] metadata specifications. A case study applying GAM is presented in section 6.

5. THE GEOGRAPHICAL AND MULTIDIMENSIONAL METAMODEL (GEOMDM)

Even with a GDW providing the conventional and geographical data integration, there is still not semantic integration among the metadata of OLAP tools and AGIS applications. That is, we may not know the semantic correspondences between CWM OLAP and GAM metadata, which implies that an integrated processing of these tools/services may not be processed yet. In order to address this issue, we propose GeoMDM. CWM OLAP and GAM are metamodels that have different focus and metadata nomenclature. However, there is intersection points that provide a semantic correspondence among theirs metadata. These semantic correspondences are graphically depicted in figure 8, which corresponds to the design view in UML of GeoMDM.

In figure 8, the metadata in light grey correspond to the GeoMDM metamodel and the metadata in dark grey correspond to source catalogues. Each source describes its identification and location in order to help a data consumer to search for the correct source. GeoMDM aims to support and

[4]These attributes are used to assure that the MBR of a Feature type must be minor or equal to the MBR of its geo-graphical view and the MBR of a geographical view must be minor or equal to the MBR of its geo-graphical space

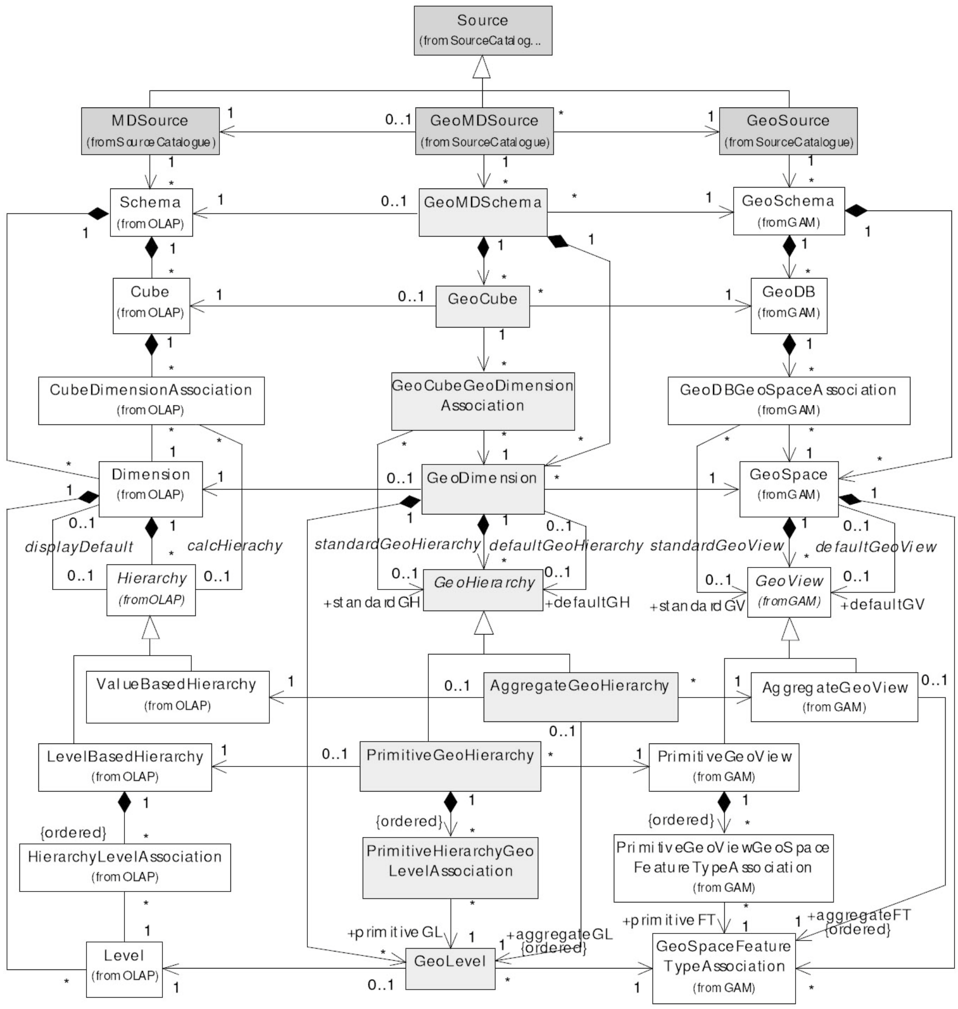

Fig. 8. GeoMDM Design View

document the semantic integration among CWM OLAP and GAM metadata in order to: 1) abstract the complexity found in the separated use of these metamodels, 2) publish location information about multidimensional and geographical metadata to help users or application programs to search for the metadata they need and 3) accomplish the semantic integration of these metamodels without altering their definitions (each GeoMDM metadata store the references of its CWM OLAP and GAM metadata). The semantic correspondences depicted in figure 8 (in light grey) are described as follows.

A GeoMDSchema represents a logical container of GeoCubes and GeoDimensions and provides the correspondence between a OLAP:Schema and GAM:GeoSchema. A GeoCube associates an OLAP:Cube with a GAM:GeoDB in order to represent a cube with geo-coding data. A GeoDimension links an OLAP:Dimension with a GAM:GeoSpace aiming to provide an analysis region that is mapped to an edge of a GeoCube. A GeoHierarchy is an abstract class that represents a view about an ordered subset of geo-coding data. A PrimitiveGeoHierarchy and a AggregatesGeoHierarchy are sub-classes of GeoHierarchy that support the integration between OLAP:LevelBasedHierarchy-GAM:PrimitiveGeoView and OLAP:ValueBasedHierarchy-GAM:AggregateGeoView, respectively. Note, there is no association between OLAP:Hierarchy and GAM:GeoView due to these are abstracts classes and their specializations already address it. We highlight that the OLAP:Hierarchy and

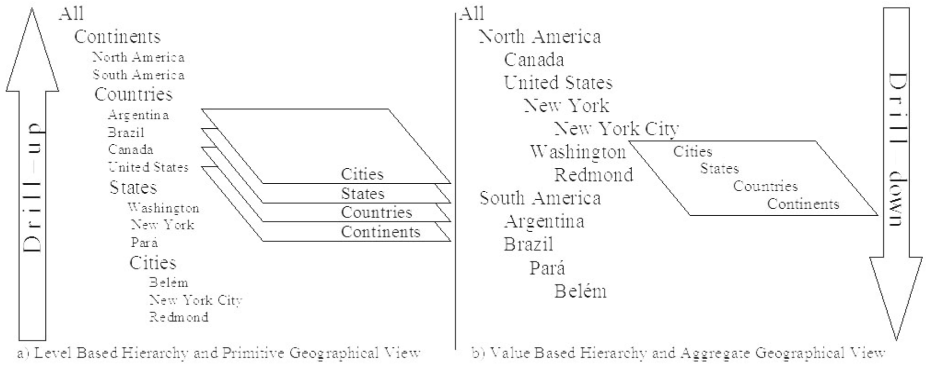

Fig. 9. Multidimensional and geographical data generalizations and specializations

GAM:GeoView concepts are essential for supporting data generalizations and specializations (drill operations) in CWM OLAP and GAM metamodels. Figure 9 summarizes this. A GeoLevel corresponds to a subset of geo-coding data, which correspond to the association between an OLAP:Level and a GAM:GeoSpaceFeatureTypeAssociation. Similar to CWM OLAP and GAM, the remainder metadata represent association metadata. In order not to impair the legibility of the GeoMDM diagram and also not to overload its development, we decided to follow the same reasoning applied for GAM attributes and association directions (see section 4). Regarding the GeoMDM association multiplicities, these specify that a CWM OLAP metadata may be associated at most one GAM metadata, which can be referenced to many CWM OLAP metadata. For example, a Status level of a Product dimension does not have a geographical reference. However, a Country level of a Customer dimension can just reference, at most, one Country Feature type, which can be referenced by many Country levels from other dimensions (e.g. Store or Vendor Country level). A case study applying GeoMDM is presented in section 6.

6. CASE STUDY

In order to be as open and extensible as possible and to demonstrate the application of our proposal we have developed 1) a GDW following GeoDWFrame and applying GML and 2) the metamodels CWM OLAP, GAM and GeoMDM according to MOF specification, more specifically based on XMI and JMI.

The case study GDW has extended a conventional DW by adding GML data. We have used the Sales_Fact_1997 Data Mart of Food-Mart DW, because it can be easily acquired from the SQL Server CD installation. To build the GDW we basically: 1) update the Sales_Fact_1997 schema according to GeoDWFrame proposal; 2) code the geo-data into GML; 3) load the GML data into primitives dimensions and 4) store the conventional data into respective dimensions. Regarding the metamodels implementation, it is based on a sequence of tasks found in Santos et al. [dos Santos et al. 2003a; 2003b]. These can be summarized as follows: 1) definition of the metamodels in UML; 2) generation of XMI metamodels from the UML metamodels; 3) importation of XMI metamodels into a Java compliant MOF repository; 4) exportation of the XMI metamodels to JMI metamodels interfaces and 5) Implementation of JMI metamodels classes to instantiate its metadata. Figure 10(A) presents the Metadata Repository (MDR) browser with the instanced metamodels. MDR is an open-source MOF repository based on Java platform. Figure 10(B) depicts the Sales_Fact_1997 updated schema according to GeoDWFrame. The dimensions: 1) Country, State and City are primitives, 2) Shared is composed, 3) Customer and Store are macro and 4) Time_by_day, Product and Promotion are conventional. Figure 11 shows a Web Service architecture [W3C 2002] that is an instance of layers I,

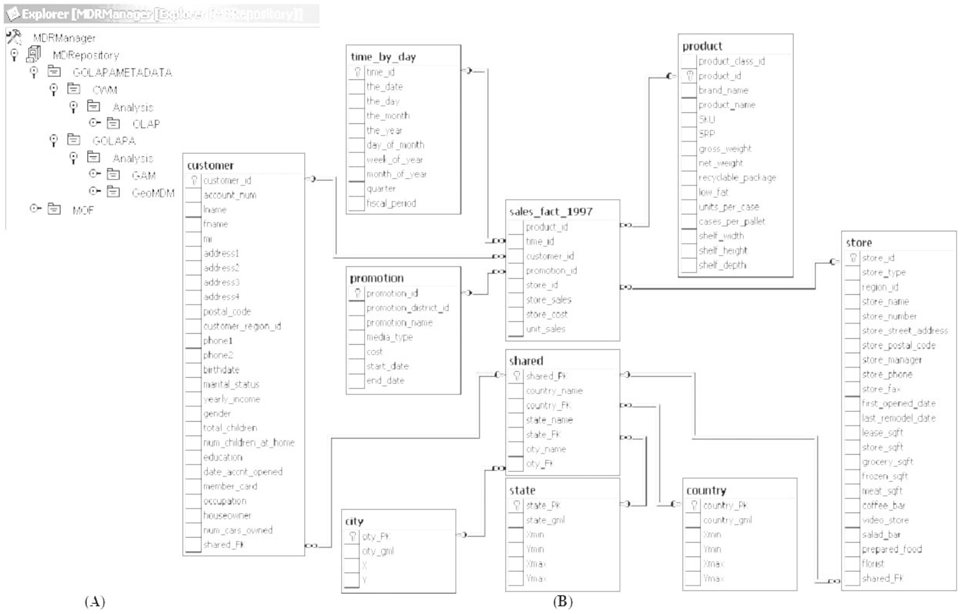

Fig. 10. MDR with CWM OLAP, GAM and GeoMDM plus the Geo_Sales_Fact_1997 schema

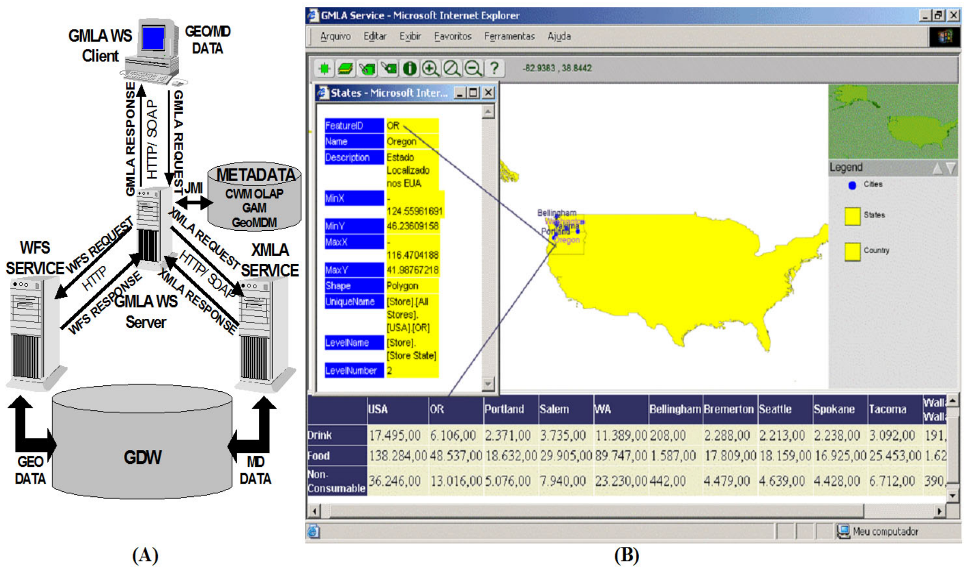

Fig. 11. An instance of GOLAPA and our geographical-multidimensional prototype

II and III of GOLAPA and the prototype of our geographical and multidimensional tool.

In figure 11(A), the GMLA WS Client is responsible for formulating the requests according to the GMLA request schema [da Silva et al. 2004], sending these requests to the GMLA WS Server and then, after obtaining the GMLA response document [Fidalgo et al. 2003], for graphically displaying the results. The METADATA repository stores the MOF metadata of CWM OLAP, GAM and GeoMDM. The GMLA WS Server is both consumer and provider. It is a consumer because it accesses the XMLA multidimensional service [XMLA 2002] and/or the WFS geographical service [OGC 2002b] in order to get the requested data. It is also a provider, because its Application Program Interface (API) provides operations that allow a GMLA WS Client to request multidimensional and/or geographical data. The communication between the GMLA WS Server and the XMLA SERVICE is based on Web Services, which send and receive information coded in a SOAP envelope [W3C 2002]. On the other hand, the communication between the GMLA WS Server and the WFS SERVICE is achieved by sending and receiving HTTP requests, instead of using the SOAP protocol due to the WFS SERVICE do not operate according to the Web Services standard.

Figure 11(B) illustrates a prototype of the GMLA WS Client prototype. It shows the total of the product sales, classifying by the product category and grouping the results by states and cities of USA, where the cities should be inside a BBOX [OGC 2002b; 2001a] that contains just the Oregon and Washington states. These data are stored in the GDW (Geo_Sales_Fact_1997) presented in figure 10(B).

In figure 11 (A and B), after having received the request parameters sent by the GMLA WS Client, the GMLA WS Server performs the multidimensional request (total of the product sales, classifying by the product category and grouping the results by states and cities of USA), which is processed by the XMLA service. Following this, the GMLA WS Server searches in GeoMDM, and consecutively in CWM OLAP and GAM, which returned multidimensional members have metadata with geographic correspondence (states and cities). In the sequence, GMLA WS Server elaborates the spatial request (cities inside a BBOX that contains just the Oregon and Washington states). The request is sent to the WFS geographic service to recover all the geographic features that are involved in spatial request. After this, the GMLA WS Server: 1) processes the multidimensional and geographical data integration (a join operation) and builds a GMLA response document containing the integrated data and its metadata, 2) creates a SOAP envelope and then, 3) returns it to the client application. The GMLA WS Client receives and processes the SOAP envelope to graphically show the result. For this graphical processing, the GMLA response document is transformed (based on eXtensible Stylesheet Language Transformations (XSLT) [W3C 2003]) to maps in the Scalable Vector Graphics (SVG) format [W3C 2008] and to tables in the HyperText Markup Language (HTML) format [W3C 1999], respectively. The final graphic result can be visualized in a Web browser with a SVG Plug-in [5].

7. CONCLUSION AND FUTURE WORK

The work presented in this paper is part of the GOLAPA project, which is concerned with the development of an open and extendible architecture for multidimensional and/or spatial processing.

Much research addresses the use of DW with GIS, but still lacks a consensus about the design of the spatial dimensional schemas of Geographical DW (GDW) and of metamodels for providing a semantic integration among the metadata of such technologies. Aiming to address this, we have analysed some related work in order to propose the framework GeoDWFrame together with the GAM and the GeoMDM metamodels. GeoDWFrame is used to guide the design of geographical dimensional schemas that must support analytical, geographical or analytical-geographical operations. To provide this capability, GeoDWFrame proposes two types of dimensions: geographical (primitive and composed) and hybrid (micro, macro and joint). The use of these dimensions aims to minimize the geo-data redundancy and moreover, provide a natural support to OLAP, AGIS and OLAP-AGIS services, because the GDW will handle descriptive and geometrical data and will not apply spatial measure. Note that according to GOLAPA, the GeoDWFrame uses GML to define geo-data, but the GeoDWFrame is not limited to GOLAPA and, for this reason, it can use geometry abstract types to handle geo-data.

Regarding the metamodels proposed in this paper, these are based on CWM OLAP and OGC/ISO specifications and follow the MOF standard to describe its syntax and semantic aiming to allow the

[5]http://www.adobe.com/svg

interchange of metadata in a standard format. In order to achieve a symmetrical metamodel to CWM OLAP, GAM add two abstraction levels in ISO/OGC metadata hierarchy, named GeoSpace and GeoView. These, respectively, correspond to 1) an analysis region that gather a collection of Feature types and 2) a relational view that represents an ordered subset of Features instances that composes a GeoSpace. While GAM aims to describe logical models of AGIS applications, GeoMDM aims to provide the semantic integration between CWM OLAP and GAM metadata. The GeoMDM purposes include 1) abstract the integration complexity, 2) publish location information about the integration metadata to help data consumer to search them and 3) accomplish the semantic integration of CWM OLAP and GAM metamodels without altering their definitions. GAM and GeoMDM are independent from GOLAPA and they may be applied to any other projects that aim to provide a multidimensional and geographical processing.

Applying the GeoDWFrame in large GDW (with vector or raster data), investigating the challenges to extract, transform and load geo-data into a GDW (based on the GeoDWFrame), enrich the capabilities of our geographical-multidimensional prototype and investigate optimization approaches based on XML compose the future work of GOLAPA project.

REFERENCES

Bedard, Y., Merrett, T., and Han, J. Fundamentals of spatial data warehousing for geographic knowledge discovery. In H. J. Miller and J. Han (Eds.), *Geographic Data Mining and Knowledge Discovery*. Taylor & Francis, New York, pp. 53–73, 2001.

Bernhardsen, T. *Geographical information systems : an introduction*. John Wiley & Sons, 1999.

Chaudhuri, S. and Dayal, U. An overview of data warehousing and olap technology. *SIGMOD Rec.* 26 (1): 65–74, 1997.

da Silva, J., Fidalgo, R. N., Times, V. C., et al. Towards a web service for geographic and multidimensional processing. In *Proceedings of the Brazilian Symposium on GeoInformatics*. Campos do Jordão, Brazil, pp. 2–17, 2004.

Date, C. J. *An Introduction to Database Systems, 7th ed. Addison-Wesley*, 2000.

Demers, M. N. *Fundamentals of Geographic Information Systems, 3rd Edition*. John Wiley, USA, 2002.

dos Santos, H. L., de Barros, R. S. M., and Fonseca, D. A proposal for management of rdf and rdf schema metadata in mof. In *Proceedings of the Confederated International Conferences DOA, CoopIS and ODBASE*. pp. 1014–1031, 2003a.

dos Santos, H. L., de Barros, R. S. M., and Fonseca, D. Uma proposta para gerenciamento de metadados nos padrões xml e dtd em repositórios mof. In *Proceedings of the Brazilian Symposium on Databases*. pp. 41–55, 2003b.

Ferreira, A. C. A model for supporting multidimensional and spatial analysis integration - master thesis - in portuguese, 2002.

Ferreira, A. C., Campos, M. L., and Tanaka, A. K. An architecture for spatial and dimensional analysis integration. In *Proceedings of World Multiconference on Systemics, Cibernetics and Informatics*. Orlando, USA, pp. 539–555, 2001.

Fidalgo, R. N. A basic infrastructure for integration of multidimensional and geographical metadata in a data warehouse environment - ph.d. thesis proposal and qualification, 2003.

Fidalgo, R. N., da Silva, J., Times, V. C., da Fonseca de Souza, F., and de Barros, R. S. M. Gmla: A xml schema for integration and exchange of multidimensional-geographical data. In *Proceedings of the Brazilian Symposium on Geoinformatics*, 2003.

Fidalgo, R. N., Times, V. C., da Silva, J., et al. Geodwframe: A framework for guiding the design of geographical dimensional schemas. In *Proceedings of the Data Warehousing and Knowledge Discovery*. Zaragoza, Spain, pp. 26–37, 2004.

Fidalgo, R. N., Times, V. C., and Souza, F. F. Golapa: An open and extensible architecture for integration between olap and gis. In *Proceedings of the Brazilian Workshop on Geoinformatics*. Brazil, 2001.

Filho, J. L. and Iochpe, C. Specifying analysis patterns for geographic databases on the basis of a conceptual framework. In *Proceedings of the 7th ACM international symposium on Advances in geographic information systems*. ACM, New York, NY, USA, pp. 7–13, 1999.

GOLAPA. Golapa project official home page - http://www.cin.ufpe.br/ golapa/architecture, 2004.

Guttman, A. R-trees: a dynamic index structure for spatial searching. In *Proceedings of the ACM SIGMOD International Conference on Management of Data*. ACM, New York, NY, USA, pp. 47–57, 1984.

Harinarayan, V., Rajaraman, A., and Ullman, J. D. Implementing data cubes efficiently. *SIGMOD Rec.* 25 (2): 205–216, 1996.

Inmon, W. H. *Building the Data Warehouse, 3rd Edition.* John Wiley, USA, 2002.

ISO. Iso/tc 211: Text of 19109 geographic information - rules for application schema, 2000.

ISO. Iso/tc 211: Text of 19107 geographic information - spatial schema, 2002a.

ISO. Iso/tc 211: Text of 19115 geographic information - metadata, 2002b.

Kimball, R. *The data warehouse toolkit: practical techniques for building dimensional data warehouses.* John Wiley & Sons, Inc., New York, NY, USA, 1996.

Kimball, R., Reeves, L., Thornthwaite, W., Ross, M., and Thornwaite, W. *The Data Warehouse Lifecycle Toolkit: Expert Methods for Designing, Developing and Deploying Data Warehouses with CD Rom.* John Wiley & Sons, Inc., New York, NY, USA, 1998.

Kouba, Z., Matousek, K., and Miksovský, P. On data warehouse and gis integration. In *Proceedings of the 11th International Conference on Database and Expert Systems Applications.* Springer-Verlag, London, UK, pp. 604–613, 2000.

Larman, C. *Applying UML and Patterns: An Introduction to Object-Oriented Analysis and Design and the Unified Process.* Prentice Hall PTR, Upper Saddle River, NJ, USA, 2001.

Miksovský, P. and Kouba, Z. Golap - geographical online analytical processing. In *Proceedings of the 12th International Conference on Database and Expert Systems Applications.* Springer-Verlag, London, UK, pp. 442–449, 2001.

OGC. Filter encoding implementation specification, 2001a.

OGC. The opengis abstract specification - topic 1: Feature geometry, 2001b.

OGC. The opengis abstract specification - topic 11: Opengis metadata, 2001c.

OGC. Opengis geography markup language (gml) implementation specification, 2002a.

OGC. Web feature service (wfs) implementation specification, 2002b.

OGC. Opengis reference model, 2003.

OMG. Common warehouse metamodel (cwm) specification 1.1, 2001.

OMG. Meta object facility (mof) specification. Tech. rep., OMG, 2002a.

OMG. Xml metadata interchange (xmi) specification, 2002b.

Papadias, D., Kalnis, P., Zhang, J., and Tao, Y. Efficient olap operations in spatial data warehouses. In *Proceedings of the 7th International Symposium on Advances in Spatial and Temporal Databases.* Springer-Verlag, London, UK, pp. 443–459, 2001.

Papadias, D., Tao, Y., Kalnis, P., and Zhang, J. Indexing spatio-temporal data warehouses. In *Proceedings of the International Conference on Data Engineering.* pp. 166–175, 2001.

Rao, F., Zhang, L., Lan, X. Y., Li, Y., and Chen, Y. Spatial hierarchy and olap-favored search in spatial data warehouse. In *Proceedings of the 6th ACM International Workshop on Data Warehousing and OLAP.* ACM, New York, NY, USA, pp. 48–55, 2003.

Rivest, S., Bédard, Y., and March, P. Towards better support for spatial decision-making: defining the characteristics. *International Journal of Geographical Information Science* 55 (4): 539–555, 2001.

Shekhar, S., Lu, C.-T., Tan, X., Chawla, S., Vatsavai, R. R., Bedard, Y., Merrett, T., and Han, J. In H. J. Miller and J. Han (Eds.), *Map Cube: A Visualization Tool for Spatial Data Warehouses.* Taylor & Francis, New York, 2001.

Stefanovic, N., Han, J., and Koperski, K. Object-based selective materialization for efficient implementation of spatial data cubes. *IEEE Transaction on Knowledge and Data Engineering* 12 (6): 938–958, 2000.

SUN. Java community process: Java metadata interface (jmi) specification, 2002.

TerraLib. Terralib official home page - http://www.terralib.org/, 2004.

W3C. Hypertext markup language (html) specification, 1999.

W3C. Web services architecture working draft, 2002.

W3C. Xsl transformations (xslt) specification, 2003.

W3C. Xml official home page - http://www.w3.org/xml, 2004.

W3C. Scalable vector graphics, http://www.w3.org/tr/svg11/, 2008.

XMLA. Xml for analysis (xmla) specification, 2002.