Knowledge and Metadata Integration for Warehousing Complex Data

Jean-Christian Ralaivao [1,2] and Jérôme Darmont [2]

[1] École Nationale d'Informatique
BP 1487
Fianarantsoa (301)
Madagascar
ralaivao@mail.univ-fianar.mg

[2] Université de Lyon (ERIC Lyon 2)
5 avenue Pierre Mendès-France
69676 Bron Cedex
France
first_name.last_name@eric.univ-lyon2.fr

Abstract: With the ever-growing availability of so-called complex data, especially on the Web, decision-support systems such as data warehouses must store and process data that are not only numerical or symbolic. Warehousing and analyzing such data requires the joint exploitation of metadata and domain-related knowledge, which must thereby be integrated. In this paper, we survey the types of knowledge and metadata that are needed for managing complex data, discuss the issue of knowledge and metadata integration, and propose a CWM-compliant integration solution that we incorporate into an XML complex data warehousing framework we previously designed.

Keywords: XML data warehousing, complex data, metadata, knowledge, metadata and knowledge integration.

1 Introduction

Decision-support technologies, and more particularly data warehousing [Inm02, KR02, JLVV03], are nowadays technologically mature. Data warehouses are aimed at monitoring and analyzing activities that are materialized by numerical measures (facts), while symbolic data describe these facts and constitute analysis axes (dimensions). However, in real life, many decision-support fields (customer relationship management, marketing, competition monitoring, medicine...) need to exploit data that are not only numerical or symbolic. For example, computer-aided diagnosis systems might require the analysis of various and heterogeneous data, such as patient records, medical images, biological analysis results, and previous diagnoses stored as texts [Saa04]. We term such data *complex data* [DBRA05]. Their availability is now very common, especially since the broad development of the Web, and more recently the Web 2.0 (blogs, wikis, multimedia data sharing sites...).

Complex data might be structured or not, and are often located in several, heterogeneous data sources. Specific approaches are needed to collect, integrate, manage and analyze them. A data warehousing solution is interesting in this context, though adaptations are

obviously necessary to take into account data complexity (measures might not be numerical, for instance). Data volumetry and dating are also other arguments in favor of the warehousing approach.

In this context, metadata and domain-related knowledge are essential in the processing of complex data and play an important role when integrating, managing, and analyzing them. In this paper, we address the issue of jointly managing knowledge and metadata, in order to warehouse complex data and handle them, at three different levels: at the supplier level (data providers), to identify all input data sources and the role of source type drivers; at the user level (consumers), to identify all data sources for analysis and their source type drivers; at the manager level (administrators), to achieve good performance.

Since data warehouses traditionally handle knowledge under the form of metadata, we discuss the alternatives for integrating domain-related knowledge and metadata. Our position is that knowledge should be integrated as metadata in a complex data warehouse. On this basis, we also present an XML-based architecture framework for complex data warehouses that expands the one we proposed in [DBRA05].

The remainder of this paper is organized as follows. In Section 2, we survey the various kinds of knowledge and metadata that are required for managing complex data. In Section 3, we discuss the issue of knowledge and metadata integration, justify our choice, and present our revised architecture framework for complex data warehouses. In Section 4, we summarize the state of the art regarding knowledge and metadata integration. We finally conclude this paper and provide future research directions in Section 5.

2 Knowledge and Metadata Needs

2.1 Knowledge Types

Two types of knowledge must be taken in consideration: tacit and explicit knowledge [NSIH02]. Tacit knowledge includes beliefs, perspectives and mental models. Explicit knowledge is knowledge that can be expressed formally using a language, symbols, rules, objects or equations, and thus can be communicated to others. In data warehousing environments, we are particularly interested in explicit knowledge.

Then, different kinds of questions must be considered regarding the types of knowledge that are needed to manage complex data warehouses. These questions determine the description context (what), the organizational context (who, where and when), the processing context (how) and the motivation and business rules (why).

Responses to the “what”-type question describe business concepts. These elements guide the link between metadata and knowledge; while knowledge representation uses metadata contents and structure. The “how” and “why” questions relate to each process’ motivation and the way it operates, in comparison to an existing organization. Eventually, answering to the “who”, “where” and “when” questions helps in connecting the first two categories of questions to a particular organization.

Furthermore, one type of knowledge that is often forgotten is universal or background knowledge. For example, the number of days in a month, the work scheduler with wrought days, public holidays, constitute some background knowledge that is essential for decision or analytical queries.

We must also consider statistical knowledge, which may include descriptive statistics about the data warehouse contents, or hypotheses about attributes' characteristics, such as probabilistic laws or sampling methods. Statistical knowledge may be provided by data analysis or data mining, and results should be reinjected into the system.

Technical knowledge is also very important at different phases of the data warehouse lifecycle. At a high level of abstraction, it is closely related to metadata. Technical knowledge includes knowledge about data sources and targets, standard and specific data types, database management systems (DBMSs), software and hardware platforms, technologies, etc. Indexing techniques available in each DBMS belong to this type of knowledge too.

Closely related is knowledge about organizational and geographical deployment, which includes information about users, their needs, their attributions and their constraints in regard to their needs (e.g., in terms of response time, volume of processed data, result format, etc.).

The last kind of knowledge we must consider relates to data warehouse administration. It provides information about how the data warehouse is used (access statistics) and how the interface between the data warehouse and the operational systems articulates, i.e., what the transactional applications and their characteristics (frequencies, response times, users...) are; and what the major Extracting, Transforming and Loading (ETL) problems (planification to satisfy user requirements with respect to work schedule, identification of peak periods...) are. The refreshment policies of the data warehouse contents are also important here, since they dictate the rotation period of summary data, the purge period and dormant data determination.

2.2 Metadata Types

We identified five transversal and complementary classifications for metadata in the literature. In the first classification [HMT00], metadata are classified based on the data warehouse architecture layers, as follows:

- metadata associated with data loading and transformation, which describe the source data and any changes operated on data;
- metadata associated with data management, which define the data stored in the data warehouse;
- metadata used by the query manager to generate an appropriate query.

The second classification [HMT00] divides metadata into:

- technical metadata that support the technical staff and contain the terms and definition of metadata as they appear in operational databases;
- business metadata that support business end-users who do not have any technical background;
- information navigator metadata, which are tools that help users navigate through both the business metadata and the warehoused data.

In the third classification [HMT00], metadata may be:

- static metadata that are used to document or browse the system;
- dynamic metadata that can be generated and maintained at run time. A new kind of metadata is made of metadata that handle the mapping between systems.

In the fourth classification [Kim05], metadata may be:

- system catalog metadata or data descriptors;
- relationship metadata that store information about the relationships between data entities (primary key/foreign key relationships, generalization/specialization relationship, aggregation relationship, inheritance relationships and any other special semantic relationship implying update or delete dependency);
- content metadata formed by descriptions of the contents of stored data at an arbitrary granularity. Content metadata may be as simple as one keyword, or as complex as a business rules, formulae or links to whole documents;
- data lineage metadata, which are lifecycle data about stored data (information about the creation of data, subsequent updates, transformation, versioning, summarization, migration, and replication, transformation rules, and descriptions of migration and replication);
- technical metadata that store technical information about stored data: format, compression or encoding algorithm used, encryption and decryption algorithms, encryption and decryption keys, software used to create or update the data, Application Programming Interfaces (APIs) available to access the data, etc.;
- data usage metadata or business data that are descriptions of how and for what purposes the data are to be used by users and applications;
- system metadata that are descriptions about the overall system environment, including hardware, operating system and application software;
- process metadata that describe the processes in which the applications operate, and any relevant output of each step of these processes.

Eventually, the fifth classification we identified [SE06] is based upon functionality categories: infrastructure, data model, process, quality, interface and administration.

- Infrastructure metadata contain information on system components.
- Data model metadata (also called data dictionary) include definitions of data entities and the relationships among them.
- Process metadata capture information on data generation and transfer from sources to targets.
- Quality metadata contains information on the actual data stored and helps in assessing data quality (e.g., factual measurements).
- Interface metadata (also called reporting metadata) support data delivery to end-users.
- Finally, administration metadata include data that are necessary for administering the data warehouse and its associate applications (security, authentification, usage tracking...).

To conclude this section, we cite an important standardization initiative: the Common Warehouse Metamodel (CWM [Gro03]). CWM has been established by the Object Management Group (OMG) within its framework of Meta-Object Facilities (MOF). CWM purposes a metamodel that can be instantiated to obtain an operational data warehouse. Each of the metadata types we enumerated in the above classifications should be mapped into one or several CWM components.

3 Knowledge and Metadata Integration for Complex Data Warehousing

3.1 Integrating Knowledge and Metadata

Current data warehouse architectures are based on metadata. However, they are sometimes themselves a materialization of domain-related knowledge that facilitates the management of data warehouses and helps in achieving good performance. It is difficult for classical architectures to manage complex data without domain-related knowledge nor background knowledge. For example, a data warehouse administrator needs some background, domain-related knowledge in addition to metadata to select clustering or indexing techniques.

There are three possibilities to jointly manage knowledge and metadata: coding and representing knowledge as metadata; modelling metadata to match knowledge representation; managing metadata and knowledge separately. The advantages and drawbacks of each possibility are discussed below.

Coding and representing knowledge as metadata present an important advantage: we can keep on using and maintain current architectures and techniques. However, it is necessary to find a solution for knowledge representation, a kind of mapping between classical knowledge representation and metadata implementation.

Modelling metadata to match knowledge representation hedges on the domain of knowledge warehouses [NSIH02], which supposes important adaptations and new considerations about current architectures. Some metadata cannot be converted into knowledge and there is a risk to loose some information. Moreover, finding a knowledge representation that can accept actual metadata is not obvious.

As for the third possibility, i.e., managing metadata and knowledge separately, a great change of architecture would be essential, because a structure that allows to coordinate and to compile metadata and knowledge contents must be devised. Instead of reducing complexity, this solution would increase it with the consideration of a new element: managing the connection between knowledge and metadata.

In conclusion, in order to build upon the assets of current data warehouse architectures, in particular in terms of performance, we select the first solution and explore it in this paper.

3.2 Revised Architecture Framework for Complex Data Warehousing

3.2.1 Global Architecture

In [DBRA05], we have already proposed an architecture framework for complex data warehouses. The main components of this architecture (Figure 1) are: the data warehouse kernel, which may be either materialized as an XML warehouse, or virtual (where cubes are computed at run time); operational databases; source type drivers that notably include mapping specifications between the sources and XML; and finally a metadata and knowledge base layer that includes three submodules related to three management processes.

These three processes for managing a data warehouse are:

1. the ETL and integration process that feeds the warehouse with source data from the operational databases ($DB\ Op$) by using drivers that are specific to each source type (ST);
2. the administration and monitoring process ($MD\ \&\ KR$) that manages metadata and knowledge (the administrator interacts with the data warehouse through this process);
3. the analysis and usage process that runs user queries, produces reports, builds data cubes, supports On-Line Analytical Processing (OLAP), etc.

Each of these processes exploits and updates the metadata and the knowledge base through four types of flows:

1. the external flow, which includes the ETL and integration flow and the exploitation (analysis and usage) flow (the warehouse may thus be considered as a black box);
2. the internal flow, between the warehouse kernel and the metadata and knowledge base layer, and between the metadata and knowledge base layer and the source type drivers;

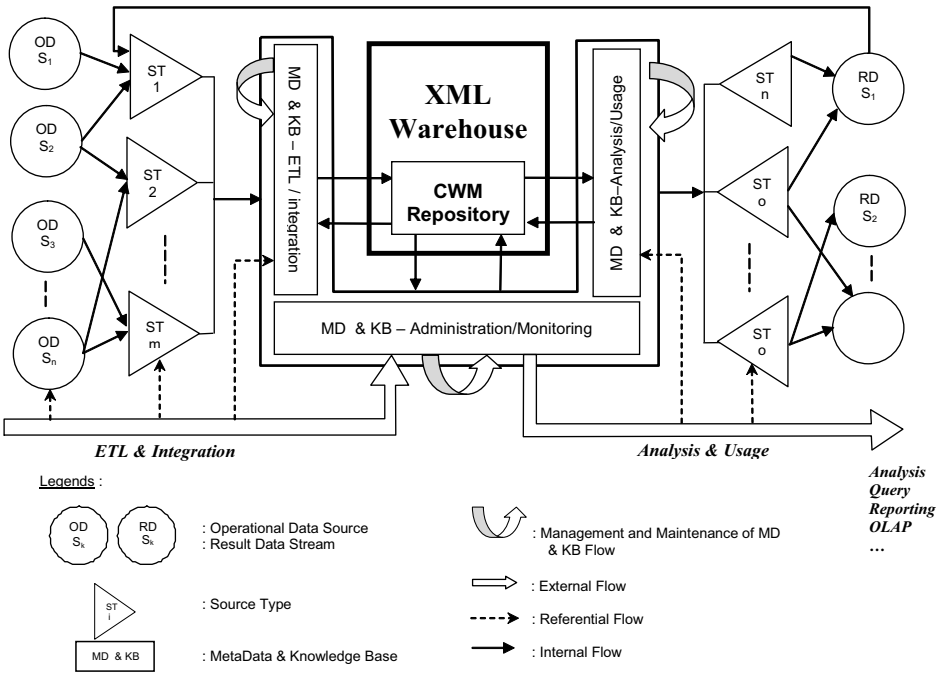

Figure 1: Complex Data Warehouse Architecture Framework

3. the metadata and knowledge management and maintenance flow, which acquires new knowledge and enriches existing knowledge;

4. the reference flow, which illustrates the fact that the external flow always refers to the metadata and knowledge base layer for integration, ETL, and analysis and usage in general.

The symmetric aspect between "sources" and "usages" around the data warehouse core allows us to eventually re-inject results as data sources. For instance, a data mining analysis may discover dependencies between variables and highlight causal relationships among them. We do use such techniques to determine the relevance of complex data with respect to given analysis goals. Then, knowledge obtained by mining can be integrated into the metadata repository and later re-used in the definition of complex data cubes.

3.2.2 Core Interface

In this section, we expand the architecture framework presented in Section 3.2.1 by integrating knowledge and metadata. Around the data warehouse core, with respect to the external components (operational data sources, result data stream and administration and monitoring), we define three metadata and knowledge base (MD & KB) repositories

corresponding to the three sides of the core (Figure 2). They constitute an interface functionality.

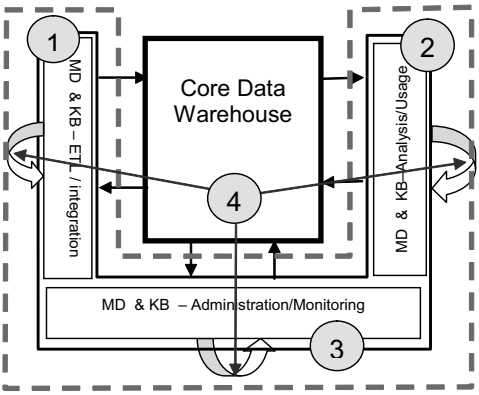

Figure 2: Interface around the core

The first $MD \ \& \ KB$ repository (labeled (1) in Figure 2) lies at the data integration and ETL process level, and includes:

- an ontology for modelling domain-related knowledge;
- information about data sources and source types;
- mappings for the extraction and transformation processes (the E and T in ETL);
- information about the loading (the L in ETL: frequency, mode...) and cleansing (purge) processes;
- a referential or metadata repository about data, materialized views, index, clusters, aliases, etc.

The $MD \ \& \ KB$ repository that is labeled (3) in Figure 2 lies at the administration and monitoring level, and references :

- an ontology for modelling domain-related knowledge;
- deployment, hardware and software constraints;
- an interface between the integration and ETL level and the usage level;
- information on users and data providers;
- data warehouse usage information (statistics, response time, availability, feedback, dormant data...);
- a referential or metadata repository about data, materialized views, index, clusters, aliases, etc.

Eventually, the MD & KB that is labeled (2) in Figure 2 lies at the usage level and completes our interface with:

- an ontology for modelling domain-related knowledge;
- information about aggregate operators (hierarchical lattice construction [Pei03] if necessary) and data lineage that would allow users to go up to the sources if necessary;
- query optimizer data (query reformulation and rewriting);
- a referential or metadata repository about data, materialized views, index, clusters, aliases, etc.

Note that some of the elements we have just enumerated (e.g., ontology and referential repository) are present in more than one interface. Hence, they must be factorized at a higher level (labeled (4) in Figure 2). Moreover, this level must include metaknowledge, i.e., knowledge for acquiring, expressing, using, storing, retrieving knowledge, and even creating new knowledge. The major part of this level resides within the CWM repository.

3.2.3 XML as a Pivot Language

The architecture we propose necessitates a universal formalism so that all its components (core, metadata, knowledge, drivers, interface, data and knowledge interchange...) can interoperate. With its vocation for semi-structured data exchange, the eXtensible Markup Language (XML) already offers a great flexibility to represent complex data, and great possibilities for structuring, modelling, and storing them [DBB+03]. XML indeed allows to store together data and their description, either implicitely or through a schema definition. This type of representation is particularly useful in a data warehousing environment where such metadata are casual. Furthermore, many XML and MOF-related facilities, such as the XML Metadata Interchange (XMI [Gro05]) or the Common Warehouse Metadata Interchange (CWMI), can help in managing metadata in an XML data warehouse and specify source-type drivers, while ensuring CWM compliance.

CWM compliance is ensured by the CWM repository that is integrated into the data warehouse kernel. All MD & KB modules use this repository to communicate with the data warehouse. CWM, through its five metamodels (object, foundation, resource, analysis and management), provides UML components (classes, associations and packages) for modelling all the data warehouse's elements [Gro03]. Table 1 illustrates the correspondences between the MD & KB modules in our architecture and the CWM metamodels.

Eventually, the advances in XML warehousing [Pok02, HBH03, RRT05, BMCA06] render this solution plausible in the near future, especially since XML-related metadata interchange facilities integrate very well in data warehouses [AvM02]. Storage possibilities are also numerous, either into relational, XML-compatible DBMSs such as Oracle, SQL Server or DB2; or into XML-native DBLSs such as Lore, eXist or X-Hive. Furthermore, XML query languages such as XQuery allow the formulation of analytical queries that

MD & KB modules	CWM metamodel			
	Foundation	Resource	Analysis	Management
ETL / Integration	X	X		
Administration / Monitoring	X			X
Analysis / Usage	X	X	X	

Table 1: MD & KB and CWM correspondences

are intricate to express in a relational system, e.g., moving window aggregations or rollup operations on ragged hierarchies [BCC+05]. Hence, our XML-based framework provides an architecture that is both extensible and "stable", and that can be compliant with future external elements (data sources, analytical techniques and usages...).

4 State of the Art

Though the litterature about metadata and knowledge is abundant, the issue of integrating metadata and knowledge is scarcely addressed. In this section, we provide a quick overview of the studies that are nonetheless related to this problem. Metadata are always present in data warehouse architectures [Inm02]. In our particular context, some interesting efforts aim at decentralizing the management of metadata into functional components of data warehouses [HMT00, Kim05, SE06]. They do not address the issue of domain-related knowledge, though.

Knowledge is indeed rarely exploited as such in data warehouse environments. However, issues related to knowledge management in the context of heterogeneous data warehouse environments have been addressed, by augmenting a federated warehouse with a knowledge repository [Ker01]. Discussions about using knowledge as a basic element for managing metadata are also regularly discussed in [Ste07]. However, this issue is mostly addressed by the knowledge management community, which works on knowledge warehouses [NSIH02, WAK05], and whose focus is obviously knowledge.

Finally, a study from the field of Geographical Information Systems (GISs, which are premium providers of complex data) is of particular interest to us. An extension of current metadata schemes has indeed been proposed to include context-based and tacit information about semantic attributes [SL06]. These ontology-based extended metadata improve data selection and interoperability decisions. Though we are more particularly interested in explicit knowledge in our context, we can exploit this solution in our framework.

5 Conclusion

In this paper, we have underlined the growing need for warehousing so-called complex data, a task that requires the management of knowledge and metadata related to these data.

We enumerated the various kinds of knowledge and metadata that must be taken into account. On this basis, we proposed to integrate knowledge as metadata in the warehouse. Finally, we expanded an XML-based, CWM-compliant architecture framework for complex data warehouses we had previously proposed in the light of the new insights discussed in this paper.

One immediate perspective of our work is to validate our present proposal by experimentation, and to evaluate the impact of metadata and knowledge integration in complex data warehouses in terms of performance. Performing performance evaluations and comparisons, basically with and without integrating knowledge and metadata, shall show the actual relevance of our solution.

A related, important follow-up of our work is to assess the consequences of metadata and knowledge integration on traditional performance optimization techniques such as view materialization, indexing, partitioning, query optimization, etc. These techniques will presumably need to be adapted to take into account domain-related knowledge and achieve the best performance.

Eventually, our position in this paper is to manage metadata and knowledge integration by representing knowledge as metadata. Though we discussed arguments in favor of this particular approach in Section 3.1, it would be interesting to explore and assess the efficacy of the other possible solutions, namely representing metadata as knowldege or managing knowledge and metadata separately.

References

[AvM02] G. Auth and E. von Maur. A Software Architecture for XML-Based Metadata Interchange in Data Warehouse Systems. In *XML-Based Data Management and Multimedia Engineering – EDBT 2002 Workshops XMLDM, MDDE, and YRWS, Prague, Czech Republic*, volume 2490 of *LNCS*, pages 1–14. Springer, March 2002.

[BCC+05] K.S. Beyer, D.D. Chamberlin, L.S. Colby, F. Özcan, H. Pirahesh, and Y. Xu. Extending XQuery for Analytics. In *2005 ACM SIGMOD International Conference on Management of Data (SIGMOD 05), Baltimore, USA*, pages 503–514, 2005.

[BMCA06] O. Boussaïd, R. Ben Messaoud, R. Choquet, and S. Anthoard. X-Warehousing: an XML-Based Approach for Warehousing Complex Data. In *10th East-European Conference on Advances in Databases and Information Systems (ADBIS 06), Thessaloniki, Greece*, volume 4152 of *LNCS*, pages 39–54. Springer, September 2006.

[DBB+03] J. Darmont, O. Boussaïd, F. Bentayeb, S. Rabaseda, and Y. Zellouf. *Web multiform data structuring for warehousing*, volume 22 of *Multimedia Systems and Applications*, pages 179–194. Kluwer Academic Publishers, 2003.

[DBRA05] J. Darmont, O. Boussaïd, J.C. Ralaivao, and K. Aouiche. An Architecture Framework for Complex Data Warehouses. In *7th International Conference on Enterprise Information Systems (ICEIS 05), Miami, USA*, pages 370–373, May 2005.

[Gro03] Object Management Group. *Common Warehouse Metamodel (CWM) Specification, version 1.1*, March 2003.

[Gro05] Object Management Group. *MOF 2.0/XMI Mapping Specification, v2.1*, September 2005.

[HBH03] W. Hümmer, A. Bauer, and G. Harde. XCube: XML for data warehouses. In *6th International Workshop on Data Warehousing and OLAP (DOLAP 03), New Orleans, USA*, pages 33–40, 2003.

[HMT00] T.N. Huynh, O. Mangisengi, and A.M. Tjoa. Metadata for Object-Relational Data Warehouse. In *2nd International Workshop on Design and Management of Data Warehouses (DMDW 00), Stockholm, Sweden*, volume 28 of *CEUR Workshop Proceedings*, page 3, June 2000.

[Inm02] W.H. Inmon. *Building the Data Warehouse*. John Wiley & Sons, third edition, 2002.

[JLVV03] M. Jarke, M. Lenzerini, Y. Vassiliou, and P. Vassiliadis. *Fundamentals of Data Warehouses*. Springer, second edition, 2003.

[Ker01] L. Kerschberg. Knowledge Management in Heterogeneous Data Warehouse Environments. In *3rd International Conference on Data Warehousing and Knowledge Discovery (DaWaK 01), Munich, Germany*, volume 2114 of *LNCS*, pages 1–10. Springer, September 2001.

[Kim05] W. Kim. On Metadata Management Technology: Status and Issues. *Journal of Object Technology*, 4(2):41–47, March-April 2005.

[KR02] R. Kimball and M. Ross. *The Data Warehouse Toolkit: The Complete Guide to Dimensional Modeling*. John Wiley & Sons, second edition, 2002.

[NSIH02] H.R. Nemati, D.M. Steiger, L.S. Iyer, and R.T. Herschel. Knowledge Warehouse: An Architectural Integration of Knowledge Management, Decision Support, Artifical Intelligence and Data Warehousing. *Decision Support Systems*, 33(2):143–161, June 2002.

[Pei03] J. Pei. A General Model for Online Analytical Processing of Complex Data. In *22nd International Conference on Conceptual Modeling (ER 03), Chicago, USA*, volume 2813 of *LNCS*, pages 321–334, October 2003.

[Pok02] J. Pokorný. XML Data Warehouse: Modelling and Querying. In *5th International Baltic Conference (BalticDB&IS 02), Tallin, Estonia*, pages 267–280, 2002.

[RRT05] L.I. Rusu, J.W. Rahayu, and D. Taniar. A Methodology for Building XML Data Warehouse. *International Journal of Data Warehousing and Mining*, 1(2):67–92, 2005.

[Saa04] K. Saad. Information-based Medicine: A New Era in Patient Care. In *ACM 7th International Workshop on Data Warehousing and OLAP (DOLAP 04), Washington, USA*, page 58, 2004.

[SE06] G. Shankaranarayanan and A. Even. *Managing Metadata in Decision Environments*, pages 153–184. Processing and Managing Complex Data for Decision Support. Idea Group Publishing, Hershey, PA, USA, April 2006.

[SL06] N. Schuurman and A. Leszczynski. Ontology-Based Metadata. *Transactions in GIS*, 10(5):709–726, November 2006.

[Ste07] R.T. Stephens. Knowledge: The Essence of Metadata. Monthly column, *DMReview.com*, 2003-2007.

[WAK05] K. Wecel, W. Abramowicz, and P.J. Kalczynski. *Enhanced Knowledge Warehouse*, pages 1057–1063. Encyclopedia of Information Science and Technology (II). Idea Group, 2005.