©2005 IEEE. Personal use of this material is permitted. However, permission to reprint/republish this material for advertising or promotional purposes or for creating new collective works for resale or redistribution to servers or lists, or to reuse any copyrighted component of this work in other works must be obtained from the IEEE.

Conceptual Design of an XML-View Driven, Global XML FACT Repository for XML Document Warehouses

Rajugan, R.[1], MIEEE, Elizabeth Chang[2], MIEEE and Tharam S. Dillon[1], FIEEE
[1] *eXel Lab, Faculty of IT, University of Technology, Sydney, Australia*
E-mail: {rajugan, tharam}@it.uts.edu.au
[2] *School of Information Systems, Curtin University of Technology, Australia*
E-mail: Elizabeth.Chang@cbs.curtin.edu.au

Abstract

In this paper, we propose a view-driven, architectural construct for modeling and designing of a Global XML FACT (𝒢xFACT) repository using logically grouped, geographically dispersed, XML document warehouses and Document Marts. To deal with organizations' evolving decision-making needs, we also provide three design strategies for building and managing of such 𝒢xFACT in the context of modeling further hierarchical dimensions and/or global document warehouses.

***Keywords**: Data warehouse, conceptual models, Conceptual views, Logical views, XML document warehouse and Object-Oriented conceptual models*

1. Introduction

Enterprise Content Management (ECM) is the integration and utilization of one or more technologies, tools, and methods to capture, manage, store, preserve, and deliver content across an enterprise [1]. Since the introduction of eXtensible Markup Language (XML) [2], XML repositories have gained foothold in many global (and government) organizations, where, e-Commerce and e-Business models have maturated in handling daily transactional data among heterogeneous information systems in multi-data formats. Due to this, the amount of data available for enterprise decision-making process is increasing exponentially and are being stored and/or communicated in XML.

With enterprise content moving rapidly towards we-based e-Commerce/e-Business and web information systems, organizations' non-traditional data such as e-mails, text documents and spreadsheets etc. are increasingly being encoded and communicated in XML among business entities and stakeholders for compatibility and re-use.

This increasing trend towards XML contents creates the need to investigate XML Document Warehouses and Document Marts for business intelligence. The global reach of such archived XML repositories over Intra/Internet present their own benefits and challenges, thus deserve detailed study of their own. Conversely, with respect to globalisation and rapidly changing business needs (e.g. mergers, de-mergers and acquisitions), the rate at which the business requirements of such enterprise repositories adopt and reflect changes pose further challenges.

To address such XML data warehouse issues for large scale enterprises, authors of the work [3] presented an XML-view [4] driven, XML Document Warehouse (XDW) conceptual model with a meaningful XML FACT (xFACT) repository. This XDW model was extended further in [5], where they use Object-Oriented Conceptual Modeling (OOCM) and (warehouse) requirement engineering techniques [6] to build a conceptual framework for XML document warehouses. This XDW framework was also utilized in Web Document Warehouse (WDW) [7] design, where enterprise web content are archived for BI. In this paper, based on the XDW model, we propose an XML-view driven, *architectural construct* for conceptually modeling and designing a Global XML FACT repository (𝒢xFACT) based on logically grouped, geographically dispersed, XML FACT repositories (xFACT) [3, 5] of XDWs, WDWs and XML Document Marts.

The 𝒢xFACT repository will; (1) provide an integrated data source for BI tools for a given context (e.g. global/regional earnings), (2) provide data and data semantics to built further (global) dimensions and dimensional hierarchies, (3) provide seamless integration of existing data warehouse sources with preserved data semantics, (4) preserve conceptual semantics of initial data warehouse environments and

Proceedings of the 16th International Workshop on Database and Expert Systems Applications (DEXA'05)
1529-4188/05 $20.00 © 2005 **IEEE**

(5) support and reflect changing warehouse requirements at a higher level of abstraction.

The rest of this paper is organized as follows. In section 2, we review some early work done in data warehouse models, followed by a brief introduction of our three-layered XML view model in section 3. Section 4 provides a brief outline of an illustrative case study used in this paper. In section 5, we present our proposed global XML FACT ($\mathcal{G}$xFACT) model with its properties and implementation options. We conclude the paper in section 6 with a brief outline on future research directions.

2. Related Work

Data warehousing concept that has gained in importance in recent years [8-10] as part of ECM initiatives. At the most basic level, data warehousing has been an approach adopted for management of large volumes of historical data for detailed analysis to provide crucial business intelligent (BI) for organisations in; (1) Decision Support Systems (DSS) [10-12], Management Information Systems (MIS)[8] and (3) Executive Information Systems [10, 11]. A data warehouse integrates large amounts of enterprise data from multiple and independent data sources consisting of operational databases into a common repository [9] for querying and analysis (using BI tools). In addition, data warehouses are designed for online analytical processing (OLAP) [8, 9, 13, 14], where the queries aggregate large volumes of data in order to detect trends and anomalies. To reduce the cost of executing aggregate queries in such an environment, warehousing systems usually pre-compute frequently used aggregates and store each materialized aggregate view [9, 15, 16] in a multidimensional data cube [9, 16-18]. These data cubes group the base data along various dimensions, corresponding to different sets of operational attributes, and compute different aggregate functions (e.g. sum, avg, min, max) on measures. To address such requirements many models have been proposed for designing data warehouses.

Since the introduction of dimensional modeling, several design techniques have been proposed to capture multidimensional data (MD) at the conceptual level. Conceptual data models for such MD, which revolves around FACTs, dimensions and hierarchies, have been extensively discussed in research and industrial literature [9, 18]. These discussions normally includes support for data warehouses and OLAP data (ROLAP, MOLAP), where MD is the feasible data model for such applications.

The early work on MD and data warehousing concepts date back to works done by W.H. Inmon et al. [11, 12, 19, 20]. Later work by Ralph Kimball's popular Star Schema [13, 21] provided the base for other well-known conceptual models such as SnowFlake and StarFlake to be derived. More recent comprehensive data warehouse design models are built using Object-Oriented concepts on the foundations of the Star Schema. In [18, 22-26], and [27], two different OO modelling approaches are demonstrated where a data cube is transformed into an OO model integrating class hierarchies. The Object-Relational Star schema (O-R Star) model [28] aims to utilise Object-Relational (O-R) data model and its features for warehouse MD data and hierarchies.

For XML data, one of the early XML data warehouse implementations for web data includes the Xyleme Project [29]. The Xyleme project [30] was successful and it was made into a commercial product in 2002. It has well defined implementation architecture and proven techniques (such as materialised views) to collect and archive web XML documents into an XML warehouse for further analysis. Another approach by Fankhauser et al. [31] explores some of the changes and challenges of a document centric XML warehouse. Other works that use XML in data warehouse context includes [32, 33].

In another related work, authors focused on building a requirement driven, meaningful FACT repository in , the work on XML-view driven XML document warehouses [3, 5] and Web document warehouses [7], authors argue that, coupling these approaches with a well defined requirement-oriented [34] conceptual design methodology will help future error-free, maintainable design of such XML warehouse for large-scale XML systems.

In DW domain, views are mainly used to provide aggregate data and queries, performance (as materialized views), meta-data and OLAP queries [8, 10-13, 17, 18, 22, 35-37]. Only few work has been done in the direction of using views for providing DW architectural constructs and frameworks [3, 5, 15, 16].

3. Preliminaries: The 3-Layered view Model for XML

The 3-layered view model used to construct the $\mathcal{G}$xFACT model was proposed by the authors in [4, 38]. The XML-view model has three-layers of abstraction, namely; (1) conceptual, (2) logical or schematic and (3) document or instance level.

The conceptual layer describes the structure and semantics of XML views in a way which is more comprehensible to human users. It hides the details of view implementation and concentrates on describing objects, relationships among the objects, as well as the associated constraints upon the objects and relationships. Due to its abstract nature, conceptual views can be defined using any high level modeling languages such as Dillon & Tan notation [39], UML [40], XSemantic Nets [41] or Enhanced-ER (Enhanced or Extended Entity-Relationship Model (EER)) [8].

The middle schema (or logical) layer describes the schema of XML views for the view implementation, using the XML Schema language. Views at the conceptual level are mapped into the views at the schema level via the transformation mechanism developed in work [42-44]. The output of this level will be in either textual (such as XML Schema language) or some visual notations that comply from the schema language (such as graph).

The document (or instance) level implies a fragment of instantiated XML data, which conforms to the corresponding view schema defined at the upper level. A detailed discussion and the formal model of this three-layered view model can be found in [38].

4. An Illustrative Case Study

To illustrate our concepts, we use an example case study of a fictitious global logistics company called LWC & e-Solutions Inc. (e-Sol) that provides global logistics, warehouse space and cold storage facilities to their global (and regional) customers and collaborative partners. The e-Sol solution includes a standalone and distributed Warehouse Management System (WMS/e-WMS), and a Logistics Management System (LMS/e-LMS) on an integrated e-Business framework called e-Hub [45] for all inter-connected services for customers, business customers, collaborative partner companies, and LWC staff (for e-commerce B2B and B2C). Some real-world applications of such company, its operations and IT infrastructure can be found in [45-47].

For e-Sol to support DSS, EIS and MIS, it is essential to provide a data model to support dimensional data in the context of data warehouse. Due to e-Sol's dynamic and heterogeneous nature (both system and data), the data warehouse model should support rapidly evolving new data formats (from relational, XML to propriety data scripts), at a higher-level of abstraction. For a local stakeholder/partners' perspective, the XDW model solve some of the problems faced by e-Sol. But in a global perspective, where multiple stakeholders/partner system are involved and need to support e-Sol global information demand, the role and scope of XDW has to be challenged and a new global warehouse model is inevitable and an unfortunate reality; thus the $\mathcal{G}$xFACT model.

5. The GxFACT Model

In this section, we present the $\mathcal{G}$xFACT model, its properties and implementation options.

$\mathcal{G}$xFACT is an *architectural construct* (than an implementation or storage structure) to build and integrate multiple XML FACT repositories and XML data marts into one global XML FACT repository that provide perspectives to an organization at the global level (Fig. 2). It is an aggregated xFACT (i.e. xFACT of one or more xFACTs) to support DSS, MIS and EIS solutions in global business environments, in the context of a global XML document warehouse.

The $\mathcal{G}$xFACT model utilizes the XML-view formalism to provide three levels of abstraction, namely conceptual, logical/schema and document/instant levels, which in turn uses OOCM approach to data warehousing and the industry standard UML as the modeling language. The design methodology starts from initial OO operational data level (Fig. 2) to the global repository level. Since it is based on high-level modeling (conceptual and logical) semantics, it is independent of the implementation platform/storage architecture or the operational data model. Also, it does not matter which implementation model (or platform) is chosen for the $\mathcal{G}$xFACT, as far as the storage model supports native XML data. Thus we say that the model and design of $\mathcal{G}$xFACT is platform independent (or produce Platform-Independent-Model (PIM)).

Since the basic constructs are views, building hierarchical dimensions (VDim) and virtual view support for OLAP queries are built-in to the model, which can be designed using top-down approaches, based on user requirements and/or performance issues.

5.1. $\mathcal{G}$xFACT Properties

Let $t_1, t_2, ...t_i.....$ (where $1 \le i \le n$) be a finite set of operational document databases (ODB, Fig. 2) from which an XDW is built for an organization (here for LWC e-Sol). Here, we consider t_i (as one logical ODB, as one ODB can be physically stored, accessed and/or operational in multiple servers, departments and states/countries) in our e-Sol example. Let $\mathcal{T}$ be the set that holds all ODBs (one per XDW). $t_i \in \mathcal{T}$, where $1 \le i \le n$. Therefore the domain of $\mathcal{T}$ (thus domain of the resulting XDW) can be shown as, $dom(\mathcal{T}) = \bigcup_{i=1}^{n} dom(t_i)$.

1529-4188/05 $20.00 © 2005 IEEE

Let $(x\mathcal{F}_1, x\mathcal{F}_2, ...x\mathcal{F}_k ...) \in x\mathcal{FACT}$, (where $1 \leq k \leq m$) a finite set of xFACT repositories of logically connected, standalone XDWs (XDW of business stokeholds/partners in e-Sol). Based on the conceptual view definition 1(b) above, let $\mathcal{V}^c$ denote a conceptual views that can be defined over a context ζ. From conceptual view properties, it can be shown that $\mathcal{V}^c$ can be either; $\mathcal{V}^c = \zeta$ or $\mathcal{V}^c \subseteq \zeta$.

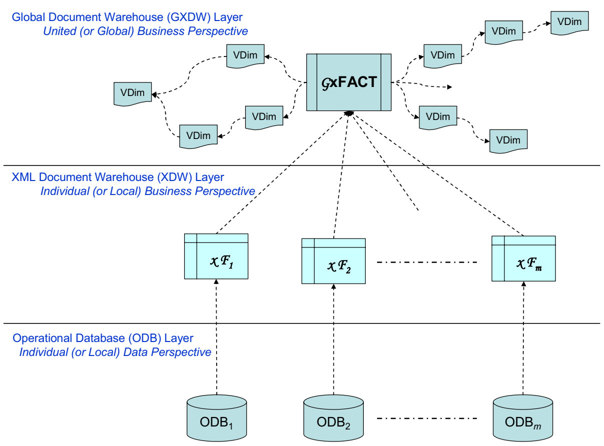

Figure 2: $\mathcal{G}$xFACT, Context Diagram

Let consider an xFACT, $x\mathcal{F}_k$ (Fig. 2-3), as a specialized context (whole xFACT/(s) or partial, as shown in Fig. 3) that is of interest to the organization at the global level. Let v_k, ($v_k \in \mathcal{V}^c$) be a conceptual view defined over the xFACT, $x\mathcal{F}_k$, where v_k represent the items in $x\mathcal{F}_k$ that is of some interest to the organization at the global level. This can be shown as; $x\mathcal{F}_k \cap \mathcal{G}x\mathcal{FACT} = v_k$.

It can be shown that, v_k can either be $v_k = x\mathcal{F}_k$ or $v_k \subseteq x\mathcal{F}_k$. This v_k is shown in Fig. 3 as the area of intersection between an xFACTs (from $x\mathcal{F}_1$ to $x\mathcal{F}_m$) and $\mathcal{G}$xFACT (highlighted). Therefore it can be shown that the resulting $\mathcal{G}$xFACT is union of all such conceptual views v_k. This is shown below;

$$\mathcal{G}x\mathcal{FACT} = \bigcup_{k=1}^{m} v_k$$

Also, in this case, since the domain of v_k is either $dom(v_k) = dom(x\mathcal{F}_k)$ or $dom(v_k) \subseteq dom(x\mathcal{F}_k)$. Also, it can be shown that the, domain of the resulting $\mathcal{G}$xFACT, $dom(\mathcal{G}x\mathcal{FACT}) = \bigcup_{k=1}^{m} v_k$

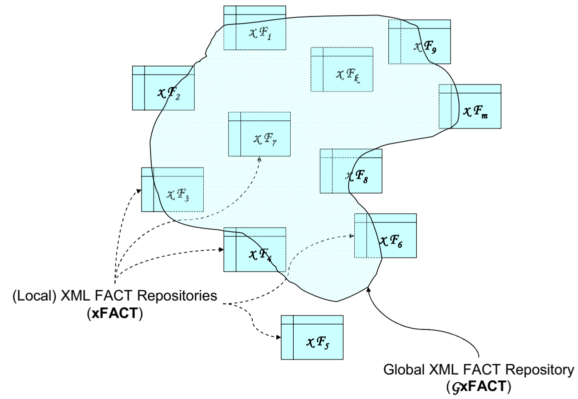

Figure 3: $\mathcal{G}$xFACT, xFACTs and the Contexts

In the original XDW conceptual model [3] consists of an xFACT repository and multiple hierarchical dimensions. This model, at the first glance is comparable to the Star/Snowflake schema [16] of the relational model, except the dimensions are XML-view based (in contrasted to dimensional tables); thus the name *virtual* dimensions (VDim).

Also, since xFACT is more complex than a relational FACT table, it can be considered as one context (e.g. earnings) that is of interest to the organization, with multiple sub-contexts (such as regional-earnings, earnings-by-city, earnings-by-warehouse etc.). Therefore it can be intuitively shown that, there exists a many-to-many (m:n) relationship between one xFACT and a VDim (or a VDim hierarchy).

Let $vd_j^{\ell} \in \mathcal{VD}^{\ell}$ ($\ell := 1 \rightarrow m$ and $j \geq 1$) where, $\mathcal{VD}^{\ell}$ is a finite set of *virtual dimensions* (VDim) [3], defined over an xFACT repository $x\mathcal{F}_{\ell} \in x\mathcal{FACT}$. Similarly, based on the works [3, 5, 7], it can be shown that, there can be many VDim (or VDim hierarchy/(ies)) defined over the $\mathcal{G}$xFACT to satisfy the global information need of an organization.

In our e-Sol example, for a $\mathcal{G}$xFACT "`Warehouse-Capacity`", many VDim's such as "`Regional-Warehouse-Capacity-by-Season`", "`Warehouse-Capacity-by-Country`", "`Warehouse-Availability-for-Logistics`", "`Profit-by-Region`" etc. can be defined, providing regional and/or global perspectives.

Some of the VDim definitions in the above example are not possible, if an XDW is built and maintained in a specific region/business unit, as integration of all available are not possible. In addition, some of these VDim requirements are redundant and an expensive task for regional/business unit (e.g. China or Australia), as they are only focused on their own local markets. On the other hand, for some warehouse

IEEE COMPUTER SOCIETY

operators (e.g. Hong Kong, USA), such global perspectives are important as they are focused mainly on global logistics and warehouse services. Therefore, though all belongs to one business organization, data perception and needs vary from one business unit another, within one organization.

In our e-Sol example, in addition to XDWs for local warehouse owners, regional $\mathcal{G}$xFACT (e.g. Europe, USA, China) can be built to support growing business demand. Also, if a requirement exists, where warehouse/logistics turn-over is very high. E.g. "`US-Logistics-Orders`", additional $\mathcal{G}$xFACT context/groups can be formed to support the demand.

5.2. $\mathcal{G}$xFACT Implementation Options

In the $\mathcal{G}$xFACT model; (1) the data source for the instance data comes from one or more xFACT of either XDW or XML document marts, (2) the Meta-Data (due to the nature of XML/XML Schema) is embedded within the source data and (3) the $\mathcal{G}$xFACT is designed using (conceptual and logical) views. Due these factors, we provide three implementation models for the global XDW (namely the $\mathcal{G}$xFACT and the associated VDims).

Option 1: Fully *persistent* (or materialized) $\mathcal{G}$xFACT repository with dimensions. In this option, the $\mathcal{G}$xFACT (the collections of views that form the $\mathcal{G}$xFACT) are fully materialized. New VDims can be defined over this repository as in simple XDW model. Also, if needed, depending on user and performance requirements, some of the VDims can be materialized (e.g. to support OLAP queries).

This option is preferred in a situation, where dimensional definitions are of dynamic nature (user query, to support third-part analytical tool etc.) and high-performance computing power (and network resources) are in abundance. This also suits well where the $\mathcal{G}$xFACT should remain reasonably static (e.g. due to the underlying XDW data sources and connectivity) and updated constantly in regular intervals to maintain data accuracy [48].

Option 2: Non-persistence (or non-materialized) $\mathcal{G}$xFACT repository with materialized (or persistence) dimensions. This is an unorthodox option, where, $\mathcal{G}$xFACT logical model is implemented (i.e. schemas, environmental parameters etc.), but data is not materialized at the $\mathcal{G}$xFACT level. This situation is comparable to a view definition stored in a relational model. But all the VDims are defined and materialized with data. Here, $\mathcal{G}$xFACT serves as a meta-data repository than a XDW repository.

This option is preferred when an organization has fixed warehouse requirements (at the global level) and wide-range of high-performance storage solutions (not computational power, such as grid or cluster computing). Also, this solution is feasible, if the underlying operational data sources are updated over a longer term than in regular, short (weekly or monthly) intervals. Another advantage of such option is that, since all the dimensions are already materialized (i.e. all complicated query processing is done and data is readily available for end-users), end-users do not require high-performance computing power, thus, suited well for regions that suffer from such issues.

Option 3: Here, it is a combination of option 1 and 2, where predefined sections of the $\mathcal{G}$xFACT repository (i.e. selected views) and selected VDims (i.e. dimensional views) are partially or fully materialized based on business (and performance) requirements.

In making a decision on which option to choose, the following factors should be considered; (1) $\mathcal{G}$xFACT requirements, (2) availability of computing power (and associated resources), (3) end-user computing resources, (4) end-user knowledge, (5) support for in-house/third party analytical tools, (7) estimated size (and predicted growth rate) of the $\mathcal{G}$xFACT and (6) required performance level.

For example, if a section of the business or regional XDW (or data availability) suffers from lack of computing and network resources, $\mathcal{G}$xFACT/VDim sections associated with such data may be materialized to improve data availability and/or performance for the overall global XDW. Note: In deciding option 3, in addition to warehouse and business requirements, warehouse operational requirements must be considered (which section/(s) and/or VDim to materialize etc.) in designing the $\mathcal{G}$xFACT conceptual model.

6. Conclusion and Future Work

In this paper, we presented an intuitive, view-driven *architectural construct* to conceptually model, design and implement a global XML FACT repository, $\mathcal{G}$xFACT. Such repository will integrate organizations' existing data warehouse solutions (XML Document Warehouses and XML marts) to provide an integrated (or global) perspective for DSS, MIS and EIS.

For future work, some further issues deserve investigation. First, the derivation of a formal $\mathcal{G}$xFACT (and associated VDim) model. Second, is the formulation of a valid empirical study to consider

IEEE COMPUTER SOCIETY

model transformation and model mapping formalism between the xFACT models and the $\mathcal{G}$xFACT model.

7. References

[1] "The ECM Association (http://www.aiim.org/index.asp)," AIIM, 2005.

[2] W3C-XML, "Extensible Markup Language (XML) 1.0, (http://www.w3.org/XML/)," 3 ed: The WWW Cons., 2004.

[3] V. Nassis, Rajugan R., T. S. Dillon, and W. Rahayu, "XML Document Warehouse Design," 6th Int. Conf. on DaWaK '04, Zaragoza, Spain, 2004.

[4] Rajugan R., et al., "XML Views: Part 1," 14th Int. Conf. on DEXA '03, Prague, Czech Republic, 2003.

[5] V. Nassis, Rajugan R., T. S. Dillon, and W. Rahayu, "Conceptual and Systematic Design Approach for XML Document Warehouses," *Int. J. of DW and Mining*, 1(3), 2005.

[6] V. Nassis, R. Rajugan, T. S. Dillon, and J. W. Rahayu, "A Requirement Engineering Approach for Designing XML-View Driven, XML Document Warehouses," The 29th Annual Int. COMPSAC '05, Edinburgh, Scotland, 2005.

[7] V. Nassis, et al., "A Systematic Design Approach for XML-View Driven Web Document Warehouses," Int. Workshop on Ubiq. Web Sys. & Intell. (UWSI '05), Singapore, 2005.

[8] R. Elmasri and S. B. Navathe, *Fundamentals of database systems*, 3 ed: Addison-Wesley, Reading, Mass. Harlow, 2000.

[9] L. Feng and T. S. Dillon, "Using Fuzzy Linguistic Representations to Provide Explanatory Semantics for Data Warehouses," *IEEE Trans. on KDE (TOKDE)*, vol. 15, No. 1, pp. 86-102, 2003.

[10] P. Ponniah, *Data Warehousing Fundamentals: A Comprehensive Guide for IT professionals*. NY: John Wiley & Sons Inc., 2001.

[11] P. Gray and H. J. Watson, *Decision Support in The Data Warehouse*. USA: Prentice Hall PTR, 1998.

[12] M. Humphris, et al., *Data Warehousing: Architecture & Implementation*. USA: Prentice Hall PTR, 1999.

[13] R. Kimball and M. Ross, *The data warehouse toolkit : the complete guide to dimensional modeling*, 2nd ed. NY: Wiley, 2002.

[14] R. Kimball and J. Caserta, *The data warehouse ETL toolkit : practical techniques for extracting, cleaning, conforming, and delivering data*. Hoboken, NJ: Wiley, 2004.

[15] D. Theodoratos and T. Sellis, "Dynamic Data Warehouse Design," 1st Int. Conf. on DaWak '99, Italy, 1999.

[16] V. Gopalkrishnan, et al. "Star/Snow-flake Schema Driven Object-Relational Data Warehouse Design and Query Processing Strategies," 1st First Int. Conf. on DaWaK 1999.

[17] A. Gupta, et al.,, *Materialized views: techniques, implementations, and applications*: MIT Press, 1999.

[18] J. Trujillo, et al., "Applying UML For Designing Multidimensional Databases And OLAP Applications. 13-36," *Advanced Topics in Database Research, Idea Group Publication*, vol. 2, pp. 13-36, 2003.

[19] J. Blazewicz, et al., *Handbook on Data Management in Information Systems*: Springer, Berlin ; New York, 2003.

[20] W. H. Inmon, et al., *Building the operational data store*. New York, USA: John Wiley & Sons, 1996.

[21] R. Elmasri and S. Navathe, *Fundamentals of database systems*, 4th ed. New York: Pearson/Addison Wesley, 2004.

[22] J. Trujillo, et al., "Designing Data Warehouses with OO Conceptual Models," in *IEEE Computer Society, "Computer"*, 2001, pp. 66-75.

[23] S. Lujan-Mora, et al., "Extending the UML for Multidimensional Modeling," 5th Int. Conf. on UML '02, Dresden, Germany, 2002.

[24] S. Luján-Mora, et al., "Advantages of UML for Multidimensional Modeling," Proc. of the 6th Int. Conf. on ICEIS '04, Porto, Portugal, April, 2004.

[25] S. Lujan-Mora, et al., "Multidimensional Modeling with UML Package Diagrams," Proc. of the 21st Int. Conf. on Conceptual Modeling (ER '02), 2002.

[26] S. Luján-Mora, et al., "Data Mapping Diagrams for Data Warehouse Design with UML," 23rd Int. Conf. on Conceptual Modeling (ER '04), Shanghai, China, 2004.

[27] A. Abelló, J. Samos, and F. Saltor, "Understanding facts in a multidimensional object-oriented model," 4th Int. Workshop on Data Warehousing and OLAP (DOLAP '01), 2001.

[28] W. Rahayu, et al., "Object-Relational Star Schemas," 13th IASTED Int. Conf. on PDCS '01, LA, USA, 2001.

[29] Lucie-Xyleme, "Xyleme: A Dynamic Warehouse for XML Data of the Web," Int. Conf. on IDEAS '01, Grenoble, France, 2001.

[30] Xyleme, "Xyleme Project (http://www.xyleme.com/)," 2001.

[31] P. Fankhauser and T. Klement, "XML for Data Warehousing Changes & Challenges," DaWaK 2003, Prague, 2003.

[32] M. Golfarelli, et al., "Data warehouse design from XML sources," Proc. of the 4th ACM Int. workshop on Data warehousing and OLAP, Atlanta, Georgia, USA, 2001.

[33] E. Medina, et al., "Handling Conceptual Multidimensional Models Using XML through DTDs," Proc. of the 19th British National Conf. on Databases: Advances in Databases, 2002.

[34] V. Nassis, et al., "Goal-Oriented Requirement Engineering for XML Document Warehouses," in *Processing and Managing Complex Data for DS*,: Idea Group Publishing, 2005.

[35] T. Debevoise, *The Data Warehouse Method: Integrated Data Warehouse Support Environments*.: Prentice Hall, USA, 1999.

[36] T. Johnson, "Data warehousing," in *Handbook of massive data sets*: Kluwer Academic Publishers, 2002, pp. 661-710.

[37] J. Trujillo et al., "A UML Based Approach for Modeling ETL Processes in Data Warehouses," Int. Conf on ER '03, 2003.

[38] Rajugan R., E. Chang, T. S. Dillon, and F. Ling, "A Three-Layered XML View Model: A Practical Approach," 24th Int. Conf. on ER '05, Klagenfurt, Austria, 2005.

[39] T. S. Dillon and P. L. Tan, *Object-Oriented Conceptual Modeling*: Prentice Hall, Australia, 1993.

[40] OMG-UML™, "Unified Modeling Language™ (UML) Version 1.5 Specification," OMG 2003.

[41] Rajugan R., et al., "Semantic Modelling of e-Solutions Using a View Formalism with Conceptual & Logical Extensions," 3rd Int. IEEE Conf. on INDIN '05, Perth, Australia, 2005.

[42] L. Feng, et al., "A Semantic Network-based Design Methodology for XML Documents," *ACM Transactions on Information Systems (TOIS)*, vol. 20, No 4, pp. 390 - 421, 2002.

[43] L. Feng, E. Chang, and T. S. Dillon, "Schemata Transformation of Object-Oriented Conceptual Models to XML," *Int. Journal of Com. Sys. Sci. & Eng.*, vol. 18, No. 1, pp. 45-60, 2003.

[44] R. Conrad, et al., "XML conceptual modeling using UML," 19th Int. Conf. on Conceptual Modeling (ER '00), USA, 2000.

[45] E. Chang, et al., "A Virtual Logistics Network and an e-Hub as a Competitive Approach for Small to Medium Size Companies," 2nd Int. Human.Society@Internet, Korea, 2003.

[46] E. Chang, et al., "Virtual Collaborative Logistics and B2B e-Commerce," e-Business Conf., Duxon Wellington, NZ, 2001.

[47] ITEC, "iPower Logistics (http://www.logistics.cbs.curtin.edu.au/)," 2002.

[48] M. K. Mohania, et al. "Data Warehouse Design and Maintenance through View Normalization," 10th Int. Conf. on DEXA '99, Italy, 1999.

1529-4188/05 $20.00 © 2005 **IEEE**

IEEE COMPUTER SOCIETY