

©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.

Engineering XML Solutions Using Views

Rajugan R.¹, Elizabeth Chang², Tharam S. Dillon¹ and Ling Feng³

¹*eXel Lab, Faculty of Information Technology, 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

³*Faculty of Computer Science, University of Twente, The Netherlands*

E-mail: ling@ewi.utwente.nl

Abstract

In industrial informatics, engineering data intensive Enterprise Information Systems (EIS) is a challenging task without abstraction and partitioning. Further, the introduction of semi-structured data (namely XML) and its rapid adaptation by the commercial and industrial systems increased the complexity for data engineering. Conversely, the introduction of OMG's MDA presents an interesting paradigm for EIS and system modeling, where a system is designed at a higher level of abstraction. This presents an interesting problem to investigate data engineering XML solutions under the MDA initiatives, where, models and framework requires higher level of abstraction. In this paper we investigate a view model that can provide layered design methodology for modeling data intensive XML solutions for EIS paradigm, with sufficient level of abstraction.

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], it is fast emerging as the dominant standard for storing, describing and interchanging data among various Enterprises Information Systems (EIS) and heterogeneous databases. In combination with XML Schema [3], which provides rich facilities for constraining and defining XML content, XML provides the ideal platform and the flexibility for capturing and representing complex EIS data formats. Conversely, as industrial production techniques move towards a distributed model, the need to exchange data between heterogeneous data sources [4] in a seamless

fashion is constantly increasing. These heterogeneous data sources could arise from server groups from different manufacturers or databases at different sites with their own schemas.

In software engineering, many methodologies have been proposed to capture real-world problems into manageable segments, which can be communicated, modelled and developed into error-free maintainable software models and modules [5, 6]. With software engineering moving towards OMG's Model-Driven Architecture (MDA) initiative [7], platform independent models play a vital role in system development and data engineering. Under the MDA initiative, first the model of a system (and sub-systems) is specified via an abstract notation independent of the technical or deployment specifications (i.e. Platform Independent Model or PIM) and then the PIM is transformed into a deployment model (i.e. Platform Specific Model or PSM) by adding platform or deployment specific information into the PIM. To support MDA initiatives in data engineering, data semantics, constraints and model requirements has to be specified precisely at a higher level of abstraction. This presents an opportunity to investigate data views as a means of providing data abstraction and semantics in PIMs for data intensive MDA solutions.

In the context of MDA solutions for XML domains, it is still a challenging task to produce PIMs despite the flexibility and the semantic richness of the semi-structured data/schema languages (e.g. XML). This is mainly due to OO modeling languages such as OMG UML™ [8], Extended-ER [9] etc. provide insufficient modeling constructs for utilizing XML schema like data descriptions and constraints, while XML Schema lacks the ability to provide higher levels of abstraction (such as conceptual models, visual constraints) that are easily understood by humans. We note that models are often abstract representations that only keep so much

of the detail as is relevant to the particular problem being considered [5, 6]. In this context, XML Schema generally is too low a representation to permit users to interact, visualize or understand it. To rectify this situation, many researchers in work such as [10-15], have applied intuitive techniques, notations and transformation methodologies to capture XML semantics at the conceptual level. This presents an interesting challenge to look at engineering XML data, such as data warehouse models, under the MDA initiatives.

In this paper, we investigate how a layered view model [16-18] is successfully applied in engineering XML solutions, and support MDA initiatives such as producing PIMs (and PSMs).

The rest of this paper is organized as follows.. Section 2 briefly looks at some early view models and definitions, while section 3 outlines our layered view model. Section 4 briefly describes some work done in engineering real-world XML solutions that use our view model. This is followed by section 5, which concludes this paper with some discussion on future research directions.

2. Related Work

Here we first briefly look at some of the view models available today and some of the proposals for new view mechanisms supporting semi-structured data, namely XML, RDF [19] and OWL [20] and view constraint specification. A detailed, comprehensive discussion on view models can be found in [21].

We can group the existing view models into three categories, namely; (a) classical (namely relational) views, (b) OO view models, (c) semi-structured (namely XML) view models and (d) views for SW. Early view models includes relational [9, 22] and OO [23, 24]. An extensive set of literature can be found in both academic and industry forums in relation to various view related issues such as (i) models, (ii) design, (iii) performance, (iv) automation and (v) turning/refinement, mainly supporting the 2-Es; data Extraction and Elaboration (with and some research directions towards 3-Es, i.e. 2-Es with Extension).

Since the emergence XML [2], the need for semi-structured data models that have to be independent of the fixed data models and data access, which go against fundamental properties of the classical data models. Many researchers have attempted to solve semi-structured data issues by using graph based [25] and/or semi-structured data models [26, 27]. Again, the actual view definitions are only available at the lower level of the implementation and not at the conceptual

and/or logical level [21]. One of the early discussion on views for XML was by Serge Abiteboul [28] and later more formally by Sophie Cluet et al. [29]. They proposed a declarative notion of XML views. Abiteboul et al. pointed out that, a view for XML, unlike classical views, should do more than just providing different presentation of underlying data [28]. These concepts, which are implemented in the Xyleme project [30], provide one of the most comprehensive mechanisms to construct an XML view to-date. But, in relation to conceptual modeling, these view concepts provide no support. Other view models for XML include; (a) the MIX (Mediation of Information using XML) view system [31, 32] and (b) an intuitive view model for XML using Object-Relationship-Attribute model for Semi-Structured data (ORA-SS) [33]. This is one of the first view model that supports some of abstraction above the data language level.

In related work in Semantic Web (SW) [34] paradigm, some work has been done in views for SW [35, 36], where the authors proposed a view formalism for RDF document with support for RDF schema (using a RDF schema supported query language called RDQL [37]). This is one of the early works focused purely on RDF/SW paradigm and has sufficient support for logical modeling of RDF views. The extension of this work (and other related projects) can be found at [38]. RDF is an object-attribute-value triple, where it implies object has an attribute with a value [11]. In related area of research, the authors of the work propose a logical view formalism for ontology [39] with limited support for conceptual extensions, where materialized ontology views are derived from conceptual/abstract view extensions. Another area that is currently under development is the view formalism for SW Meta languages such as OWL, namely views for OWL in the "User Oriented Hybrid Ontology Development Environments" [40] project.

3. Preliminaries: The Layered View Model for XML

Our layered view model [17, 18] has three-layers of abstraction (Fig. 1), namely; (1) conceptual, (2) logical or schematic and (3) document or instance level. The view model is based on the postulates 1 and 2, about the real world.

Postulate 1: The term *context* refers to the domain that interests an organization as a whole. It is more than a measure and implies a meaningful collection of objects, relationships among these objects, as well as

some constraints associated with the objects and their relationships, which are relevant to its applications.

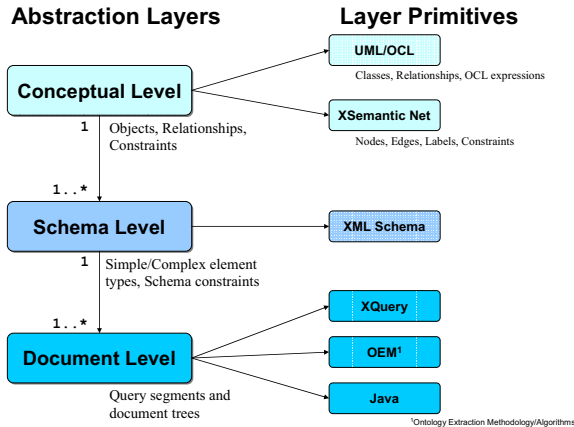


Figure 1: Layered XML View Model (context diagram)

Postulate 2: The term *view* refers to a certain perspective of the context that makes sense to one or more stakeholders of the organization or an organization unit at a given point in time.

The conceptual layer (Fig. 1 & 2) 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 [6], UML [41], XSemantic Nets [42] or Extended Entity-Relationship Model (E-ER) [43]. The conceptual views are constructed using a set of *conceptual operators* [16].

Definition 1: A **conceptual view** is the one which is defined at the conceptual level with higher level of abstraction.

The middle schema (or logical) layer (Fig. 1 & 2) 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 [10-12]. 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).

Definition 2: A **logical view** is an imaginary (XML) document schema which points to a collection of semantically related (XML) tag definitions from a domain and satisfies a conceptual view definition from the target conceptual domain.

The document (or instance) level (Fig. 1 & 2) implies a fragment of instantiated XML data, which

conforms to the corresponding view schema defined at the upper level.

Definition 3: A document (or instance) view is an XML document that is instantiated or materialized (at the document level), where the instance is well-formed and valid document (here XML) that conforms to a corresponding logical view. The instantiation is done by a document view query expression (here native XML query language).

A detailed discussion on this layered view model and formal properties can be found in [16, 17].

The conceptual view models mentioned above are equivalent of the PIMs in the MDA framework. The logical views and the corresponding document views can be considered as the (static) PSM models of the corresponding conceptual views.

4. Engineering XML Solutions with the Layered View Model

In this section, we briefly present some of the architectural constructs that utilizes our view model (Fig. 2), in the context of XML data engineering. Since XML and XML driven solution frameworks are on the increase, it is important to provide models and techniques for XML, which is at a high enough level of abstraction but with rigorously defined standards that are to be more widely understood by both developers and non-technical users. We adopt this concept in our work with XML solutions. All models described below have three levels abstraction (similar to our view model) and each level is modelled and/or designed using the appropriated view (conceptual, logical or document).

4.1. Views for XML Databases and Repositories

In relational DBMS systems, views are used in the context of access control, query refinement, performance enhancement and providing data perspectives for complex aggregate data. In the Enterprise Content Management (ECM) [44] framework, it is a data intensive task, especially when handling large volumes of distributed heterogeneous data, such as data warehousing.

But with increasing heterogeneous database schemas (relational, XML and other formats) and contents challenge the traditional database and view techniques. Therefore for XML database and repository designers can develop their views (PIM) using higher-level modeling languages and use automated tools to implement (PSMs) such views in

their multi-site, multi-platform DB/Repository systems. In doing so, the view definitions are not coupled to a specific query syntax and/or specification standards.

4.2. XML-View based UAC Middleware Design

The proposed growth for XML repositories in the EIS paradigm and their use in either to store data or as an interoperability layer for legacy applications provides the need to investigate user access control in such repositories [45]. The widespread use of XML highlights the need for flexible and expressive access control models for XML documents to protect sensitive and valuable information from unauthorized access (both by humans and machines/agents).

Traditionally, views provided user access control mechanism in many DBMS. In the work [45], authors present an our layered view-based access control model, which supports access control for both human and machine data users. The design methodology proposed is based on our view formalism and support conceptual level design of UAC (PIM) constraints and

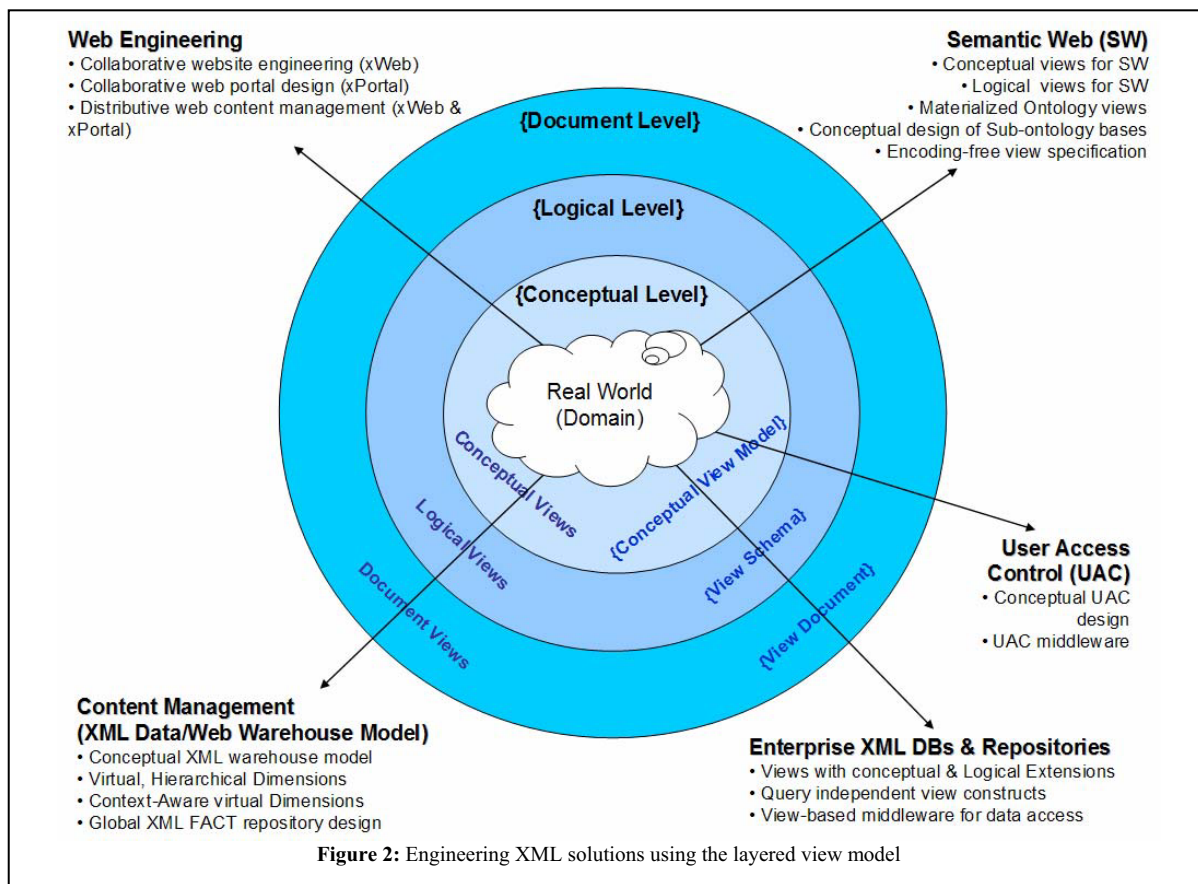
acts as a middleware layer for XML repositories and the databases alike. Such PIM, UAC constraints can be mapped to various enterprise databases and repositories using multiple query/DBMS languages (PSMs).

4.3. A View-Driven XML Document Warehouse Model

Enterprise Content Management (ECM) [44] is a data intensive task, especially when handling large volumes of distributed heterogeneous data, such as data warehousing. To address such an issue, the authors of the work [46, 47], proposed an XML-view based design methodology for modeling and designing XML document warehouses (XDW), using our view model for modeling dimensional (XML) data. Later they extended their work to accommodate web content in Web (XML) document warehouse design [48].

4.4. A View-Driven, Distributed XML FACT Repository

Global XML FACT Repository (gxFACT) [49, 50]



is an *architectural construct* to build and integrate multiple XML FACT repositories [46, 47] and XML data marts into one global XML FACT repository that provide perspectives to an organization at the global level. 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 GxFACT model utilizes the layered view model 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 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 GxFACT, as far as the storage model supports native XML data. Thus we say that the model and design of GxFACT is platform independent (or produce Platform-Independent-Model (PIM)).

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

4.5. Ontology Extraction Using Views for Semantic Web (SW-views)

Views for SW requires some form of abstraction [35, 51], as SW documents and querying are done at the logical level or with logical syntaxes to handle heterogeneous schemas such as in multi-site ontology bases. Therefore we argue that a view formalism for SW requires 3-Es (data extraction, Elaboration and Extension). Though there exists some work in regards to a logical view formalism for SW [35, 40, 52], most of them are tied to RDF specific schema/syntaxes that do not provide conceptual extensions In related work [21], the layered view model (called SW-view) provide support for ontology extraction in the form of materialised ontology views in the MOVE [39, 51] system.

4.6. Collaborative Web Engineering

The increase in enterprise web content in XML and storage of data in XML document format will provide greater semantic clarity and enable easier access and evaluation of the semantically rich web contents. In order to keep the web content descriptive among business partners, yet discrete, where a particular user/staff may want to get an appropriate view of data (warehouse information, storage capacity etc) at a given location or level of the hierarchy.

One way to handle such a complex task is to model and build semantic-aware enterprise websites and web portals [53, 54], using views, where the web content and their associated user interface definitions are captured at the conceptual level (using conceptual views) and mapped to logical view schemas where, additional presentation constraints (such as local company web styles/formats) are applied.

The proposed web architecture is called eXtensible web (xWeb) [55] and eXtensible portal (xPortal) [53]. Na duses views for XML as their core data store that also incorporate a web user interface design methodology and as part of our view model. The methodology is called Web User Interaction Analysis Model (WUiAM) [56, 57] to built collaborative, user-centred websites and web portals.

For example, the design methodology provides three levels of abstraction namely; (1) conceptual, (2) logical or schema and (3) document levels. Intuitively, xWeb methodology provides a 3-step design process to engineer web contents.

At the conceptual level a PIM is developed including; (1) design of the website and its semantics (such as site layout, structure, data, user access control) using a generic model (e.g. UML/OCL), which serves as the XML repository for the website, (2) develop abstract user interface definitions [58] using abstract (web) user interface (AUI) objects or user perspectives [59] and (3) we derive *conceptual views* to build the web pages (i.e. a web page construct corresponds to one (or more) conceptual view/(s) and web portals (view of a view/ aggregate view).

At the logical level; (1) transformation of the XML repository captured in the PIM (e.g. in UML) to XML (schema and document) using the website data (page contents, layouts, resources etc), (2) transform AUI objects to *web user interface* (WUI) definitions such as stylesheet definitions and (3) transform *conceptual views* to *XML view* schema definitions (schema &

constructor or XQuery definitions) using transformation rules described in [11].

And finally at the document level the PIM is deployed. The transformation here is threefold; (1) fill XML repository with web data, (2) transform conceptual view definitions in PIM to PSM (materialized XML views) and (3) generate XHTML documents using materialized XML (view) documents and UI definitions (e.g. XSLT transformations).

The main advantage of the *xWeb/xPortal* design is that, the user does not feel any difference between a classical (HTTP) website and a *xWeb* driven website, as there is no loss of performance traditionally related to XML technologies (XQuery) as the user access directly the X/HTML pages and not native XML documents. The traditional HTTP web server serves as the front end to users and the XML repository serves as the data source for both *xWeb* and *xPortal* servers.

5. Conclusion and Future Work

Though very useful, existing view formalisms (for all data models including XML) lack higher level modeling techniques and abstraction that are needed to describe, model, and communicate complex systems such as data warehouse and e-commerce systems. In this paper, we presented how our layered view model is successfully applied in engineering XML solutions, and support MDA initiatives such as producing PIMs (and PSMs). Our approach is to use views to provide required abstraction for XML data.

For future work, a lot of issues deserve investigation. First is the investigation into dynamic aspects of the XML-view formalism. Second is investigation into formally integrate the layered view model in supporting all aspects of the MDA initiatives, including automation of schema and code generation.

6. 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 World Wide Web Consortium (W3C), 2004.
- [3] W3C-XSD, "XML Schema," vol. 2004, 2 ed: W3C, 2004.
- [4] P. M. S. Poon, T. S. Dillon, and E. Chang, "Transformation of QoS data into XML characterizing data communication in Real Time Distributed Systems," presented at 2nd IEEE International Conference on Industrial Informatics (INDIN '04), Berlin, Germany, 2004.
- [5] G. Booch, *Object-oriented analysis and design with applications*, 2nd ed. Redwood City, Calif. Reading, Mass.: Benjamin/Cummings Pub. Co.; Addison-Wesley, 1993.
- [6] T. S. Dillon and P. L. Tan, *Object-Oriented Conceptual Modeling*: Prentice Hall, Australia, 1993.
- [7] OMG-MDA, "The Architecture of Choice for a Changing World®, MDA Guide Version 1.0.1 (<http://www.omg.org/mda/>)," OMG, 2003.
- [8] OMG-UML™, "UML 2.0 Final Adopted Specification (<http://www.uml.org/#UML2.0>)," 2003.
- [9] R. Elmasri and S. Navathe, *Fundamentals of database systems*, 4th ed. New York: Pearson/Addison Wesley, 2004.
- [10] L. Feng, E. Chang, and T. S. Dillon, "A Semantic Network-based Design Methodology for XML Documents," *ACM Transactions on Information Systems (TOIS)*, vol. 20, No 4, pp. 390 - 421, 2002.
- [11] L. Feng, E. Chang, and T. S. Dillon, "Schemata Transformation of Object-Oriented Conceptual Models to XML," *International Journal of Computer Systems Science & Engineering*, vol. 18, No. 1, pp. 45-60, 2003.
- [12] R. Conrad, D. Scheffner, and J. C. Freytag, "XML conceptual modeling using UML," presented at 19th International Conference on Conceptual Modeling (ER '00), USA, 2000.
- [13] R. Xiaou, T. S. Dillon, E. Chang, and L. Feng, "Modeling and Transformation of Object-Oriented Conceptual Models into XML Schema," presented at 12th International Conference on Database and Expert Systems Applications (DEXA '01) 2001, 2001.
- [14] R. Xiaou, T. S. Dillon, E. Chang, and L. Feng, "Mapping Object Relationships into XML Schema," presented at Proceedings of OOPSLA Workshop on Objects, XML and Databases, 2001.
- [15] Rajugan R., E. Chang, T. S. Dillon, and L. Feng, "XML Views, Part III: Modeling XML Conceptual Views Using UML," presented at 7th International Conference on Enterprise Information Systems (ICEIS '05), Miami, USA, 2005.
- [16] R.Rajugan, E. Chang, T. S. Dillon, and F. Ling, "A Layered View Model for XML Repositories & XML Data Warehouses," presented at The 5th International Conference on Computer and Information Technology (CIT '05), Shanghai, China, 2005.
- [17] R.Rajugan, E. Chang, T. S. Dillon, and F. Ling, "A Three-Layered XML View Model: A Practical Approach," presented at 24th International Conference on Conceptual Modeling (ER '05), Klagenfurt, Austria, 2005.
- [18] R.Rajugan, E. Chang, T. S. Dillon, and F. Ling, "XML Views: Part 1," presented at 14th International Conference on Database and Expert Systems Applications (DEXA '03), Prague, Czech Republic, 2003.
- [19] W3C-RDF, "Resource Description Framework (RDF), (<http://www.w3.org/RDF/>)," 3 ed: The World Wide Web Consortium (W3C), 2004.
- [20] W3C-OWL, "OWL: Web Ontology Language 1.0 reference (<http://www.w3.org/2004/OWL/>)," W3C, 2004.
- [21] C. Wouters, Rajugan R., T. S. Dillon, and J. W. Rahayu, "Ontology Extraction Using Views for Semantic Web," in *Web Semantics and Ontology*, D. Taniar and W. Rahayu, Eds. USA: Idea Group Publishing, 2005.
- [22] C. J. Date, *An introduction to database systems*, 8th ed. New York: Pearson/Addison Wesley, 2003.
- [23] W. Kim and W. Kelly, "Chapter 6: On View Support in Object-Oriented Database Systems," in *Modern Database Systems*: Addison-Wesley Publishing Company, 1995, pp. 108-129.
- [24] S. Abiteboul and A. Bonner, "Objects and Views," presented at ACM SIGMOD Record, Proceedings of the International Conference on Management of Data (ACM SIGMOD '91), 1991.
- [25] Y. Zhuge and H. Garcia-Molina, "Graph structured Views and Incremental Maintenance," presented at Proceeding of the 14th IEEE Conference on Data Engineering (ICDE '98), USA, 1998.
- [26] S. Abiteboul, R. Goldman, J. McHugh, V. Vassalos, and Y. Zhuge, "Views for Semistructured Data," presented at Workshop on Management of Semistructured Data, USA, 1997.

- [27] H. Liefke and S. Davidson, "View Maintenance for Hierarchical Semistructured," presented at Proceedings of the Second International Conference on Data Warehousing and Knowledge Discovery (DaWak '00), London, UK, 2000.
- [28] S. Abiteboul, "On Views and XML," presented at Proceedings of the eighteenth ACM SIGMOD-SIGACT-SIGART symposium on Principles of database systems (PODS '99), Philadelphia, Pennsylvania, USA, 1999.
- [29] S. Cluet, P. Veltri, and D. Vodislav, "Views in a Large Scale XML Repository," presented at Proceedings of the 27th VLDB Conference (VLDB '01), Roma, Italy, 2001.
- [30] Lucie-Xyleme, "Xyleme: A Dynamic Warehouse for XML Data of the Web," presented at International Database Engineering & Applications Symposium (IDEAS '01), Grenoble, France, 2001.
- [31] B. Ludaescher, Y. Papakonstantinou, P. Velikhov, and V. Vianu, "View Definition and DTD Inference for XML," presented at Post-ICDT Workshop on Query Processing for Semistructured Data and Non-Standard Data Formats, 1999.
- [32] B. Ludascher, Y. Papakonstantinou, and P. Velikhov, "Navigation-Driven Evaluation of Virtual Mediated Views," presented at Extending DataBase Technology (EDBT '00), 2000.
- [33] Y. B. Chen, T. W. Ling, and M. L. Lee, "Designing Valid XML Views," presented at Proceedings of the 21st International Conference on Conceptual Modeling (ER '02), Tampere, Finland, 2002.
- [34] W3C-SW, "Semantic Web, (<http://www.w3.org/2001/sw/>)," W3C, 2005.
- [35] R. Volz, D. Oberle, and R. Studer, "Views for light-weight Web ontologies," presented at Proceedings of the ACM Symposium on Applied Computing (SAC '03), USA, 2003.
- [36] R. Volz, D. Oberle, and R. Studer, "Implementing Views for Light-Weight Web Ontologies," presented at Seventh International Database Engineering and Applications Symposium (IDEAS'03), Hong Kong, SAR, 2003.
- [37] W3C-RDQL, "RDQL - A Query Language for RDF, (<http://www.w3.org/Submission/2004/SUBM-RDQL-20040109/>)," W3C, 2004.
- [38] KAON, "KAON Project (<http://kaon.semanticweb.org/Members/rvo/Folder.2002-08-22.1409/Module.2002-08-22.1426/view>)," 2004.
- [39] C. Wouters, T. S. Dillon, J. W. Rahayu, E. Chang, and R. Meersman, "A Practical Approach to the Derivation of a Materialized Ontology View," in *Web Information Systems, Edited by D. Taniar and J.W. Rahayu*, D. Taniar and W. Rahayu, Eds. USA: Idea Group Publishing, 2004.
- [40] HyOntUse, "User Oriented Hybrid Ontology Development Environments, (<http://www.cs.man.ac.uk/mig/projects/current/hyontuse/>)," 2003.
- [41] OMG-UML™, "Unified Modeling Language™ (UML) Version 1.5 Specification," OMG 2003.
- [42] Rajugan R., E. Chang, L. Feng, and T. S. Dillon, "Semantic Modelling of e-Solutions Using a View Formalism with Conceptual & Logical Extensions," presented at 3rd International IEEE Conference on Industrial Informatics (INDIN '05), Perth, Australia, 2005.
- [43] R. Elmasri and S. B. Navathe, *Fundamentals of database systems*, 3 ed: Addison-Wesley, Reading, Mass. Harlow, 2000.
- [44] AIIM, "The ECM Association (<http://www.aiim.org/index.asp>)," AIIM, 2005.
- [45] R. Steele, W. Gardner, Rajugan R., and T. S. Dillon, "Design of an XML View Based User Access Control (UAC) Middleware," presented at IEEE International Conference on Technology, e-Commerce and e-Service (EEE-05), Hong Kong, 2005.
- [46] V. Nassis, R.Rajugan, T. S. Dillon, and W. Rahayu, "XML Document Warehouse Design," presented at 6th International Conference on Data Warehousing and Knowledge Discovery (DaWak '04), Zaragoza, Spain, 2004.
- [47] V. Nassis, R.Rajugan, T. S. Dillon, and W. Rahayu, "Conceptual and Systematic Design Approach for XML Document Warehouses," *International Journal of Data Warehousing and Mining*, vol. 1, No 3, 2005.
- [48] V. Nassis, R.Rajugan, T. S. Dillon, and W. Rahayu, "A Systematic Design Approach for XML-View Driven Web Document Warehouses," presented at International Workshop on Ubiquitous Web Systems and Intelligence (UWSI '05), Colocated with ICCSA '05, Singapore, 2005.
- [49] R.Rajugan, E. Chang, and T. S. Dillon, "Conceptual Design of an XML-View Driven, Global XML Fact Repository," presented at 1st International Workshop on Data Management in Global Data Repositories (GRep '05), held in conjunction with the 16th International Conference on Database and Expert Systems Applications (DEXA '05), Copenhagen, Denmark, 2005.
- [50] R.Rajugan, E. Chang, and T. S. Dillon, "Conceptual Design of an XML FACT Repository for Dispersed XML Document Warehouses & XML Marts," presented at The 5th International Conference on Computer and Information Technology (CIT '05), Shanghai, China, 2005.
- [51] C. Wouters, T. S. Dillon, J. W. Rahayu, E. Chang, and R. Meersman, "Ontologies on the MOVE," presented at 9th International Conference on Database Systems for Advanced Applications (DASFAA '04), Jeju Island, Korea, 2004.
- [52] P. Spyns, R. Meersman, and J. Mustafa, "Data Modeling Versus Ontology Engineering," presented at SIGMOD, 2002.
- [53] W. Gardner, Rajugan R., E. Chang, and T. S. Dillon, "xPortal: XML View Based Web Portal Design," presented at 17th International Conference on Software & Systems Engineering and their Applications (ICSSEA '04), Paris, France, 2004.
- [54] R.Rajugan, W. Gardner, E. Chang, and T. S. Dillon, "xWeb: An XML View Based Web Engineering Methodology," presented at International Workshop on Ubiquitous Web Systems and Intelligence (UWSI '05), Colocated with ICCSA '05, Singapore, 2005.
- [55] Rajugan R., W. Gardner, E. Chang, and T. S. Dillon, "xWeb: An XML View Based Web Engineering Methodology," presented at International Workshop on Ubiquitous Web Systems and Intelligence (UWSI '05), Colocated with ICCSA '05, Singapore, 2005.
- [56] W. Gardner, E. Chang, and T. S. Dillon, "Analysis Model of Web User Interface for Web Applications," presented at 16th International Conference on Software & Systems Engineering and their Applications (ICSSEA '03), France, 2003.
- [57] W. Gardner, E. Chang, and T. S. Dillon, "Two Layer Web User Interface Analysis Framework Using SNN and iFIN," presented at On The Move to Meaningful Internet Systems 2003: OTM 2003 Workshops HCI-SWWA, 2003.
- [58] E. J. Chang, "Object Oriented User Interface Design and Usability Evaluation," in *Department of Computer Science & Computer Engineering: La Trobe University*, Melbourne, Australia, 1996.
- [59] E. Chang and T. S. Dillon, "Integration of User Interfaces with Application Software and Databases Through the Use of Perspectives," presented at 1st International Conference on Object-Role Modeling (ORM '94), Australia, 1994.