Building a Web-Enabled Multimedia Data Warehouse

Hyon Hee Kim* and Seung Soo Park

Department of Computer Science and Engineering
Ewha Womans University, Seoul, Korea
(heekim,sspark)@ewha.ac.kr
(*on leave at) Department of Computer Science, IPVS
University of Stuttgart, Stuttgart, Germany
kim@informatik.uni-stuttgart.de

Abstract. Data warehousing has drawn attention as a useful approach to integrate heterogeneous data sources. Since most of data warehouses have been developed based on the relational database technology, however, difficulties are encountered, when we integrate multimedia data sources, which need a flexible data model and a content-based query language. In this paper, we study a framework for multimedia data warehousing, which integrates diverse types of multimedia data from disparate sources. Because of the interoperability and flexibility of the XML, we use the XML technology for our framework. The multimedia data warehouse supports content-based integration and retrieval of multimedia data, and manages changes of data sources efficiently in a distributed environment. Therefore, our framework significantly simplifies development of Web-based advanced multimedia application.

1 Introduction

Advanced multimedia applications require integrated access to diverse types of multimedia data. Data warehousing has attracted a great deal of attention as a useful approach for integrating business data, because it offers high performance of query processing and supports data analysis. Much research into data warehousing has been done in the context of the relational data model and the relational algebra, i.e., select-project-join queries [5]. However, the conventional warehousing approach has difficulties in integrating heterogeneous multimedia data. The relational data model does not support a natural and flexible way to describe multimedia contents. In addition, the relational query language has its limitations in querying multimedia contents.

The object-oriented approach has been widely used for integrating heterogeneous data because of the flexibility of object-oriented data models [3]. For the same reason, the object-oriented approach has been also used for managing multimedia data into database systems [8,10]. However, their strong typing system is not appropriate for integrating heterogeneous multimedia data. Since multimedia data have diverse types

C.-W. Chung et al. (Eds.): HSI 2003, LNCS 2713, pp. 594-600, 2003.

and their characteristics are highly type-specific, it is difficult to define contents of multimedia data in advance.

The aim of this study is to provide a framework for multimedia data warehousing. Particularly, we emphasize that the multimedia data warehouse should be built based on multimedia contents, because content-based retrieval is essential in managing multimedia data [4]. We assume that multimedia data sources are stored in conventional databases or exist in legacy formats. We adopt the XML technology for our framework, because the data model is simple, its self-describing feature is appropriate for describing multimedia contents, and its interoperability efficiently supports integration of external data sources on the Web into our framework.

In this paper, we present a novel approach to build a Web-enabled multimedia data warehouse. First, we propose a multimedia warehouse model and a view definition language, named the XML/M and the XQuery/M (M for Multimedia). Second, we develop a meta-knowledge manager that is responsible for change control. Finally, we provide SOAP-based wrappers, which transform heterogeneous multimedia data into our warehouse model and communicate with multimedia data warehouse. The proposed approach supports semantic-based integration and content-based retrieval of multimedia data. We expect that the Web-enabled multimedia data warehouse is essential for developing web-based modern applications.

The remainder of this paper is organized as follows. In Section 2, we provide an overview of the architecture of the multimedia data warehouse. In Section 3, we describe the media view generator, and in Section 4, we explain the meta-knowledge manager. In Section 5, we describe the wrappers, and finally, in Section 7, we give concluding remarks.

2 System Overview

Figure 1 illustrates the basic architecture of the multimedia data warehouse. In the center of the Figure, there is a multimedia data warehouse, which is composed of four components: media view generator, media object repository, meta-knowledge bases, and meta-knowledge manager. The media object repository is storage of diverse types of media objects. The media view generator is a query engine for generating media views that can be seen as a set of materialized views of media objects. It generates intra-media views and inter-media views. The intra-media views integrate homogeneous media objects, while the inter-media views integrate heterogeneous media objects. The meta-knowledge bases and manager are responsible for managing changes of underlying data sources.

The bottom of the Figure depicts heterogeneous multimedia data sources. Data sources might be multimedia data stored in (object-)relational databases, legacy formats with MPEG-7 description, or semistructured textual data. On top of each data source, there are wrappers that are in charge of data translation and communication with the warehouse. The wrappers transform diverse types of multimedia data represented by different structure into a common warehouse model. They also generate XML messages about information on changes of their sources, and send it to the warehouse using SOAP protocol.

Finally, at the top of the Figure are clients that let users interact with the multimedia data warehouse through the Web.

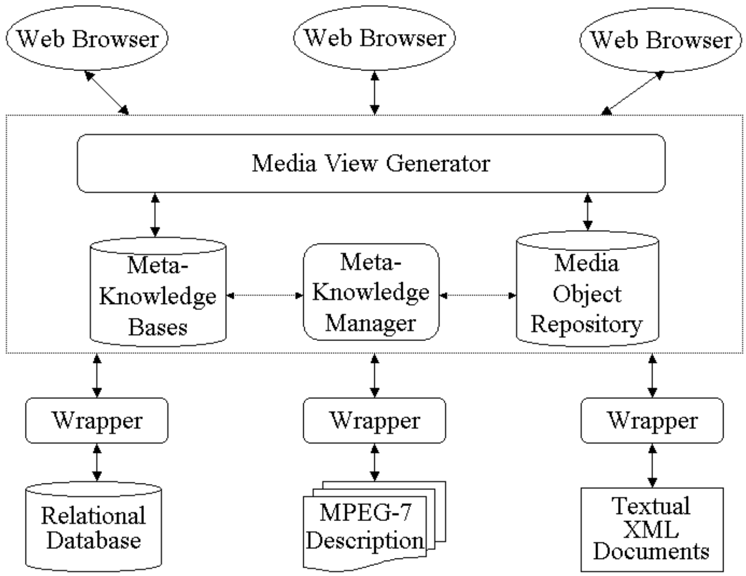

Fig. 1. Architecture of the Multimedia Data Warehouse

3 Media View Generator

The media view generator creates a set of materialized views of underlying data sources, which is named media views. As a common multimedia warehouse model, we adopt XML [2], which is becoming rapidly accepted as a standard data exchange format on the Web. The main deficiency of XML as a multimedia warehouse model is lack of managing relationships like foreign key constraint in the relational data model. For multimedia contents modeling, however, the management of relationships between objects is essential. Therefore, we extend XML with core object-oriented modeling concepts.

The XML/M Warehouse Model: The basic unit of the XML/M is a media object. To model media objects, we take two main features of the standard object-oriented data model [9]: *object identity* and *object nesting*. Each media object has a unique object identifier, and its tree structure describes media contents. The XML/M does not need fixed schemas, because it is self-describing. For example, image object might be described by color, texture, shape, etc., whereas video object might be described by meaningful scene, description of the scene, starting frame, and ending frame, etc.

Media objects with semantic relationships can be integrated into a multimedia object that can be considered as a complex object. The complex object is generated based on object-relationship graphs (ORGs), which specify semantic relationships between media objects. In the standard object model, relationships between objects are modeled as properties of each object. Our media objects focus on describing their contents, and therefore relationships are separated from the objects. While class hierarchies in OODBMSs capture Is-A relationship between classes [7], ORGs capture diverse types of semantic relationships between media objects. The ORG is also represented by XML, and therefore it provides a flexible way to specify semantic relationships.

The XQuery/M View Definition Language: The main goal of the XQuery/M query language is to provide a mechanism for generating media views. We adopt the XQuery, which is a W3C standard language for XML documents, and extend it with special operators to handle media objects. We classify these operations into operations for accessing media objects and operations for creating new media objects.

Accessing media objects has been done using FLWOR expression [11], which uses path expression and pattern. To aggregate media objects with semantic relationships, special operators have been implemented as programming constructs like *CopyTree*, *SelectSubtree*, and *MergeTree*. We are currently developing more operators to edit multimedia contents and to manipulate them.

The Layered View Mechanism: The media views are divided into intra-media views and inter-media views: The intra-media views restructure and integrate homogeneous media objects, e.g., a set of images and a set of video clips, while the inter-media views generate multimedia objects that is composed of different types of media objects with semantic links. For example, an intra-media view might be generated as a multimedia object containing photo images of a movie star, movie video clips of the star, and the article about the actor or the actress. Because a simple query is evaluated against intra-media views and a complex query is evaluated against inter-media views, improvement of query performance is expected.

4 Meta-Knowledge Manager

The meta-knowledge manager controls changes of data sources. A materialized view gets dirty whenever the underlying data sources are modified, and thus it is necessary to process update of the materialized view in response to changes to the underlying data. We develop a rule-based view maintenance technique. In the meta-knowledge bases, three groups of rule sets are stored: Rules for intra-media view maintenance, Rules for inter-media view maintenance, and Rules for contents maintenance.

Rules for Intra-media View Maintenance: If changes of underlying data sources are either creation of a media object or deletion of a media object and the media object belongs to an intra-media view, *Add* operation and *Remove* operation are executed against the specified intra-media view. *Add* operation of an object identifier adds the OID as value of leaf, and *Remove* operation of an object identifier removes the OID.

Rules for Inter-media View Maintenance: In the case of inter-media views, a change of data sources affects an ORG. Therefore, an ORG that is related to the change is updated and then intra-media views that refer to the ORG are recomputed. First, if a media object is created or deleted and it belongs to an ORG, *Add* and *Remove* operation are executed against the specified ORG. Next, System checks whether an update of the ORG is related to instance of inter-media views. If an inter-media view is detected, maintenance statements are generated and the inter-media view is recomputed.

Rules for Contents Maintenance: Because intra-media views have only OIDs of media objects, changes of multimedia contents are totally up to inter-media views. In inter-media views, multimedia contents are represented by XML trees, and therefore their changes are managed by the modified tree comparison algorithm [12]. In [6], our experience shows that the rule-based approach needs user inter-action, because deeply nested tree structure and their variation can induce several errors. To manage several extra-ordinary cases, the basic rule sets can be extended in a semi-automatic way.

5 Wrappers

A wrapper provides two major services in our multimedia warehousing environment. First, a wrapper transforms the underlying data sources into the warehouse data model. We develop two types of wrappers: the relational wrapper and the MPEG-7 wrapper. Most of relational databases currently support XML views of relational data. The relational wrapper restructures these XML documents as the XML/M model using XSLT. In XSLT, input documents are prepared as source trees. The transformation is achieved by associating patterns with templates. If a pattern is matched against elements in the source tree, a template is instantiated and result trees are created. As a result, special tags are filtered and the XML/M model is generated. Since MPEG-7 description is represented by XML, the MPEG-7 wrapper transforms the XML documents into the XML/M data model using XSLT in the same way.

Second, a wrapper supports communication with the multimedia warehouse using SOAP [1]. SOAP provides a simple and lightweight mechanism for exchanging structured and typed information between peers in a decentralized, distributed environment using XML. First, if data sources change, wrappers generate a SOAP message and send it to the warehouse. Next, if the meta-knowledge manager in the warehouse receives this notification message, it sends a request message to the wrapper. Third, wrappers send a reply message containing change information to the meta-knowledge manager. Finally, the meta-knowledge manager extracts the change information from the SOAP message, and reflects it to media views using the rule-based approach explained in Section 4.

6 Conclusions

With recent progress in computing technology, the area of multimedia database systems has attracted much attention. A main focus of multimedia databases is modeling and querying multimedia contents. Much work takes the object-oriented approach because of the flexibility, but most work supports content-based retrieval of specific media type. The object-relational databases are also developed for multimedia data, which extend the relational data model with nested tables, user-defined types and functions. However, this approach provides management of metadata, e.g., compression types, file format, and bit rate, but does not support content-based management of multimedia data directly.

Recently, XML or semistructured data model is widely used as a middleware model because of its simplicity and flexibility. Work on XML warehouses or XML-based mediators is still in its infancy, with many problems remaining to be solved.

Indexing techniques, query evaluation, and optimization should be improved in these approaches. Concerning multimedia data, we also should solve the same problems.

In this paper, we presented an XML-based framework for warehousing heterogeneous multimedia data from disparate sources. The XML/M data model unifies diverse types of multimedia data. Since the XML/M data model does not have strict types and class concepts, new data types and their contents can be easily modeled. The ORGs capture semantic relationships between different media objects, which makes it possible to retrieve semantically related multimedia objects and therefore significantly simplifies application development. The XQuery/M and media views support content-based retrieval and integration of multimedia data. The rule-based view maintenance technique and wrappers provide an efficient method for change control in a distributed Web environment.

Acknowledgements

This work was supported in part Brain Korea 21 Project of Korea Ministry of Education. We thank Prof. Bernhard Mitschang for his encouragement, hospitality, and facility supports. We also acknowledge Dr. Won Kim for suggesting this problem and for stimulating interest in it.

References

[1] D. Box et al. Simple Object Access Protocol (SOAP) 1.1, `http://www.w3c.org/TR/SOAP`

[2] T. Bray, J. Paoli, and C. M. Sperberg-McQueen, Extensible Markup Language, `http://www.w3c.org/TR/XML`

[3] M. J. Carey et al. Towards Heterogeneous Multimedia Information Systems: The Garlic Approach, In Proceedings of the 5th International Workshop on RIDE-DOM, pp.124-131, Taipei, Taiwan, 1995

[4] W. I. Grosky, Managing Multimedia Information in Database Systems, Communications of the ACM; Vol: 40, No. 12, pp. 73-80, 1997

[5] A. Gupta and I. S. Mumick. Maintenance of Materialized Views: Problems, Techniques, and Applications, IEEE Data Engineering Bulletin, Vol. 18, No. 2, pp. 1-16, 1995

[6] H. H. Kim and S. S. Park, Semantic Integration of heterogeneous XML Data Sources, In *Proceedings of OOIS*, Montpellier, France, 2002

[7] W. Kim, Object-Oriented Databases: Definition and Research Directions, IEEE Transactions on Knowledge and Data Engineering, Vol. 2, No. 3, pp. 327-341, 1990

[8] E. Oomoto and K. Tanaka, OVID. Design and Implementation of a Video-Object Database System. IEEE Transactions on Knowledge and Data Engineering, Vol. 5, No. 4, pp. 629-643, August, 1993

[9] Y. Papakonstanninou, H. Garcia-Molina, and J. Widom, Object Exchange Across Heterogeneous Information Sources, In Proceedings of ICDE, pp. 251-260, Taipei, Taiwan, 1995.

[10] D. Woelk and W. Kim. An Object-Oriented Approach to Multimedia Databases, In Proceedings of ACM SIGMOD Conference, pp. 311-325, Washington, D. C., 1986.
[11] XQuery, `http://www.w3c.org/TR/XQuery`
[12] K. Zhang and D. Shasha, Simple Fast Algorithms for the Editing Distance Between Trees and Related Problems, *SIAM Journal of Computing*, Vol. 18, No. 6, pp. 1245-1262, 1989.