Asian Journal of Information Technology 4 (12): 1131-1136, 2005

The Web-based Database Integration Approach: The Experiment of the Composite Approach to Integrate E-commerce Databases Using the Extensible Markup Language (XML)

[1]Sang Hyun Kim,[2] Hee Jung Jung
[1]College of Business Administration The University of Mississippi, USA
[2]Department of Education Washington State Univeristy, USA

Abstract: Virtually, all businesses are moving to online services in which customers have easier access with more choices when buying products than other brick-and-mortar. Online stores have their own database so customers can search and compare one store to others to find the best choice. Because each database of online stores is different from others, integrating databases is a crucial factor for successful electronic commerce (E-commerce). The extensible Markup Language (XML) is one way to integrate such heterogeneous databases. This paper introduces one approach for integrating databases by applying the eXtensible Markup Language (XML) to the composite approach. Online movie store databases, for example, were used for demonstration purposes because it is initially difficult to understand how XML can be used for database integration. This XML subset is called a MovieStore Markup Language (MSML) which is used for integrating three different movie store databases. The result of this study found that the approach is very simple, quick, convenient, and saves costs for organizations and provides better services for customers.

Key words: Extensible markup language, E-commerce, web-base database

INTRODUCTION

The growth of the Internet has been huge for the last two decades. Now, virtually every business is moving to online. In this trend, E-commerce has had a tremendous impact on the way people conduct business and on our economy as a whole[1,2]. Revenue from the Internet economics was more than $500 billion and was estimated to reach $850 billion by 2001[3]. The increasing trend of online business revenue would be continuing according to the prediction by Daikoku *et al.*,[4]at the Gartner, Inc. Despite the relative slowdown in the economy, the global business-to-business (B2B) Internet commerce will reach $8.5 trillion by the end of 2005[5]. However, the success of E-commerce depends on various factors, including customer-base needs, support of substantial E-commerce initiatives, development of E-commerce applications, competitive advantage of web sites[6], and how the organization's culture is reshaped[7]. However, the more important factors for successful E-commerce are database integration systems, and communications and computer technology[8].

Gathering and storing information is getting more complex. Many companies place databases on the web for public usages, which does not mean that customers can access such databases easily. Customers still have to go through multiple steps to access databases because many online companies do not have integrated database systems[9]. As a result, using multiple web-enabled databases is often exhausting for customers because customers must perform multiple queries every time they search for desired products online. Customers want to access multiple integrated web-databases simultaneously during a search, which should be provided by online companies for their success. This study presents a usage of the eXtensible Markup Language (XML) as a new method of integrating online company databases. XML is an easy way to integrate several different databases. However, not many companies have recognized that XML is used for database integration because it is a new concept of markup language and has not been standardized in all industries. By using one example of E-commerce databases, online movie stores, it demonstrated how easily XML can be used for database integration.

Related works: Database integration plays a crucial role in the management of heterogeneous databases. However, integrating two or more different databases is not an easy task because of differences and conflicts of existing database structures, such as models, objects, identity conflicts, representation conflicts, scope conflicts, etc.[10-14]. Even though there are some studies that discussed the conflicts and integration of heterogeneous database systems[10,14,15,18], a few studies have been done to determine common methods that solve such conflicts and integration of databases. There are many different database integration approaches because

Corresponding Author: Sang Hyun Kim, College of Business Administration The University of Mississippi, USA

many online companies have used different application specific databases[19]. Each approach may be used in different circumstances depending on the nature of business and long-term strategies for the organization.

Database integration approaches: Several integration approaches have been studied and introduced to make interoperable database systems. Missie *et al*,[20] introduced four broad approaches: (1) translation, (2) integrated, (3) loose-coupling/decentralized, and (4) broker based. Malaika[21] classifies online database integration approaches into two types, depending on in which the processing happens: (1) server-side processing, and (2) client-side processing. Other approaches are also introduced by several studies: the system overhaul approach, the application cover-up approach[2], the federated approach[21], the composite approach[15,21], the mediated approach[22], the global schema approach [23]and the data warehousing approach[11].

Each approach has pros-and-cons and can be used differently based on long-term managerial strategies of a company. For example, the system overhaul approach[15] requires the creation of totally new systems that consolidates the existing system into one. As a result, this approach is very costly and time-consuming, but it gives more potential benefits in the long-term. The global schema approach by Hughes[23] provides the conceptual global schema as a centralized database so that users can use only one database language, which may not provide more choices to users. The composite approach[15,21] is basically creating a virtual data warehouse that provides the look and environment of a single repository while allowing the original data to remain in its original location. The eXtensible Markup Language (XML) can be applied to this approach in order to get information from multiple databases. Each approach may be modified based on the company's requirements for its business activities.

Customer Communication with E-commerce Databases : Most of online businesses have some kind of database used for management within a company or for outside business activities. Customer interaction with a database will always be involved in either case. Customers may interact with one or more databases. On the other hands, database servers may interact with many clients. A typical example of this interaction between client and server is that customers access databases of online company, which uses database management, data retrieval, and the infrastructure of the Internet. That is, when customers shop online, they search shops and retrieve product information. These steps involve the interaction between customers and databases. According to Inmon[24], developing web-enabled databases merges these concepts into cutting-edge technology which will transform the way information is delivered for public usage. This cutting edge technology in many other industries can be easily found in many industries, such as tourism[25], microbiology and biotechnology[26].

In a case of online movie stores, inventories of the stores are stored in one or more databases. Customers on the web can query these databases in order to get their desired results. Because there are so many choices on the web customers can check the price of a product and availability at more than one movie store when they want to buy products. Customers will go through several steps before they make a final purchase decision. First, customers on the web must search for locations of many different online movie stores. Each store may have different resources and searching functionality. Secondly, customers sort through movie inventories at each store. After repeating the above two steps for each movie store, customers will decide where to buy their desired movie. This multi-step process is very time consuming, inefficient and annoying for customers on the web.

This problem can be generally applied to any type of E-commerce, especially online stores, having database systems. Companies simply need a centralized querying system that can integrate data from various resources so that customers can get query results at one search rather than searching at many different stores. Data provided through this centralized system should be reliable, coherent, and current. This integration of databases may save costs associated with time and efforts on behalf of both customers and companies. One simple solution for efficient and affordable database integration is applying the eXtensible Markup Language (XML) to the composite approach[15,21] that is basically creating a virtual data warehouse. XML for database integration purposes in this approach does not create an entire new database system, but it plays as a center point that integrates all related data from various sources. As a result, the original data remains at separate locations, which provides a single data repository look and feel, so there is no need for additional setup or format for each existing database for each location.

EXtensible markup language (XML): XML stands for the eXtensible Markup Language recently emerged as a new standard for data representation and exchange on the Internet[27]. According to the World Wide Web Consortium (W3C)[20], XML is an very simple language of the Standardized General Markup Language (SGML),

which is used for capturing the structure data by using tags. In fact, XML is similar to the HTML in its format; they are closely related to SGML markup language. However, there are two fundamental differences between them: separation of form and content and extensibility (Software AG, 1969). The HTML uses predefined tags to display text whereas the structures and contents of the data, including actual appearance specified by a specific application or an associated style sheet are normally defined by tags in XML.

Another difference is extensibility. XML users can define tags and structure based on requirements from their business. Additionally, tags can be created from the pre-defined structure that is used to specific schema definitions whereas W3C defines the HTML standard tag-sets. Regardless of the limitations of the HTML, many web developers misuse the HTML to not only format data, but also to structure it, which leads to trouble in parsing, inefficient searching, and high design cost. As defined by XML Working Group, XML is an application profile or restricted form of the SGML[28]. XML has been served to work with the data structuring portions of the SGML. It shows the exact meanings of the data itself. Fig. 1 shows an example of XML Code:

```
<? xml Version = "1.0"?>
<guest book>
   <entry data =" 05/07/05 04:44:50" >
   < name > chris eastwoody </ name >
   <homepage>http://www.vbcodeliberary.co.kr>/homepage >
   < comment >This is the only comment in the guestbook </comment>
   </entry>
</guestbook >
```

Fig.1 : Example of XML code for guestbook

As with the SGML, users are able to construct their own tags that are specific for their application. In the above example, new tags corresponding to the structure of a *guestbook* document have been developed. XML has tagged only the structure of the data. The <name> tag indicates the guest name while the <homepage> tag represents the guest's homepage address. Th <comment> tag simply represents any comment if there is. With this structured data, any reader, machine, or human can identify what the part of the guestbook is only by searching for the proper tag, which is what structuring the data means. There is another application or language that is used to describe the data to the specific condition of end-users. For example, the font for the <comment> tag could be Times New Roman for one application and Verdana in another. By removing the formatting information from the data, users can make the document less complex, which provides a better performance in parsing and searching.

Benefits of XML: Even though XML has not been standardized by all industries, many companies are recognizing potential benefits of XML. As mentioned earlier, the extensibility of XML is one advantage over the HTML. According to RainingData[27], some other benefits of XML includes:

- Platform independent and reusable
- Self-description and machine-readable context information
- Separation content from presentation
- Support of multilingual documents and unicode
- Facilitation of the comparison and aggregation of data
- Ability of embed multiple data and existing data types

The benefits and flexibility of XML not only can solve many existing business problems more efficiently, especially interpretative database system, but also will give new business opportunities taking advantage of XML. For example, XML has already developed content management, and publishing and news syndication. Furthermore, XML will transform many other business areas, particularly those suited to automation[27].

E-commerce Applications: W3C has developed a number of applications by taking advantages of XML. These are just a few examples. Other XML applications are used in on-line banking, push technology, web automation, database publishing, and software distribution.

Table 1. Examples of Applications Using XML

Applications	Description
CDF (Channel Definition Format) (W3C, 1994)	The Channel Definition Format is an XML application that has been developed by Microsoft
PGML (Precision Graphics Markup Language) (W3C, 1994)	A proposed 2D imaging model in XML of the PostScript language and the Portable Document Format (PDF)
RDF (Resource Description Framework) (W3C, 1994)	A framework for describing metadata applications
OSD (Open Software Description) (W3C, 1994)	A suggested XML application for automated distribution and updating software
SMIL (Synchronized Multimedia Integration Language) (W3C, 1994)	An XML application that integrates independent multimedia objects into a synchronized multimedia presentation

Applying XML into Database Integration:

Scenario: It is difficult to understand how XML is used for database integration so it will be helpful to use an example, online movie stores in this paper, to demonstrate the connection between database integration and XML. The earlier illustration of buying movies from online stores is a typical process in which customers search multiple online movie stores before they make a final purchase decision. Using this example again, XML provides a good solution for searching for movies. Customers' search time and efforts could be notably reduced because they can do multiple search processes simultaneously.

In the example, there are three small, independently owned online movie stores, and each store uses a relational database to store inventories. A relational database of each store has its own unique type of data, the numbers of tables/entities, the number of attributes, and a relationship among tables. Each of these small movie stores look for a simple, fast, and cost-effective solution to integrate their databases while they keep their business entity and database independent. Using XML with the composite approach of data integration can provide what each movie store wants to have and, at the same time, provide better quality customer service.

XML Solution: The problem arises in the scenario because each movie store has unique database systems to store its information. The differences of databases of each store possible will be simple, for example, the design of the tables and the number of attributes, or they can be more complex, such as different platforms and/or application software. Using XML, the solution involves two steps. First, it can be recognized that even if the databases are very different, each may describe a same entity sharing a basic set of attributes. For instance, movies have a title, volume number, starring names, and price. XML solution is to develop object definitions (in this scenario, Movie), which contains these four attributes. As mentioned before, one benefit of XML is extensibility. We can create a new markup language for the movie store example to describe the entity, Movie. The Movie Store Markup Language (MSML) provides a set oftags in order to describe the attributes of a movie. XML is the tool that can be used to define the new markup language. The below is an implementation of the Movie Store Markup Language

```
< MOVIE >
    < TITLE > When a men loves women </ TITLE>
    < VOL > 1 < VOL >
    < STRARRINGS >
        < S1 > Sandro Blulocky </S1>
        < S2 > Dena Washington </ S2>
        < S3 > Arnoldme Kim </S3 >
    </ STRARRINGS >
    < PRICE > $15.99 </ PRICE>
</MOVIE >
```

Fig. 2: MovieStore Markup Language (MSML)

(MSML): MSML does not have any data yet but is only template by which each entity is to follow the rules. The <MOVIE> tag signifies the beginning of a new document. Within the <MOVIE> tag, other tags, such as <TITLE>, <VOL>, <STARRINGS>, and <PRICE>, are used to describe the attributes of this movie. The *value* contained between the tags, <TITLE> *value* </TITLE>, represents the title of the movie. Next tag, <VOL> represents volume of each movie. Figure 2 in order to list starring, first we need to begin a new section with the <STARRINGS> tag. Then, the actual name of actors/actresses is listed within the <STARRINGS> tag. This is possible because an object can contain other objects in XML. Each of the three movie stores keeps track of this essential information on each movie. With an agreement from three movie stores to use this standard model/template, each movie store easily can exchange data. The final step in database integration using MSML is to create custom programs in order to generate XML tagged data from three databases, which involves some programming. Then, this data conforms to the standard template that is described in the object definition above.

In the example, three independent movie store databases that have different database format are created. Then, the MovieStore Markup Language (MSML) is defined using the eXtensible Markup Language (XML). The final step involves writing a specific program (C programming language or Java with embedded Oracle 9.1 SQL statements can be used.) for each database in order to replicate the data in the MSML. This approach is valid for the combination of different platforms or applications. By combining the three separate MSML documents into one single larger document (by simple computerization), we can successfully integrate these separate databases into one single system. Figure 3 shows the entire picture when using XML to integrate databases for the example; However, XML is not limited to the web. It can be used in many other circumstances. For instant, a large

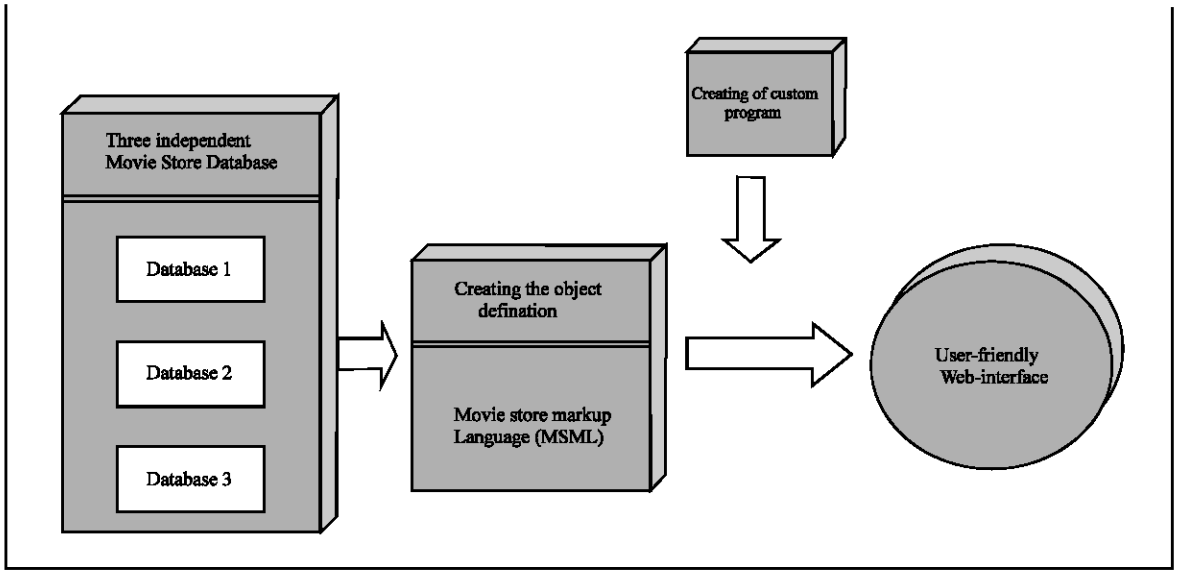

Fig . 3: The process of using XML to integrate movie store database

organization having several subsidiaries with many departments and thousands of employees may have different information banks on all employees in different database systems at each department. In this case, the problem is reproducing one database that contains all employees' directory listings without developing a new standard object definition with which each department can agree. Using XML, we can create a standard object definition; in this case, EMPLOYEE. Then, a comprehensive listing of all employees can be produced, which allows the company to create the directory of all employees.

CONCLUSIONS

Integrating different existing databases is not an easy task, but database integration is very important for all E-commercees if they want to initiate new applications or restructure the existing information system for greater profitability[19]. This study introduced a new and simply approach that is used to integrate heterogeneous database by using XML applied to the composite approach (Neild, 1999; Aren et al., 1993). The usage of XML for database integration has some strategic benefits because it allows for a fast, inexpensive, and convenient integration process. Furthermore, XML gives diverse environments a feasible alternative, especially if a company wants to avoid a high investment and large resource usage. This study illustrated the strategy by creating a subset of XML called, the MovieStore Markup Language (MSML), and used it to integrate independent movie store databases as an example. The MSML may address an important issue associated with network-accessible databases and demonstrates XML-based solution to the issue.

However, other issues, such as standardization, should be considered for future study. Without standardizing XML in all industries, some of XML benefits, including high flexibility and extensibility, can be risk[29]. For example, if an individual small company creates its own unique tags that are not identifiable to its related business partners, it will be too hard to integrate different databases. Another area of future research would be examine other integration approaches such as the data warehousing approach[11], the mediated approach [22]and the global schema approach .[23]Because of the potential benefits of XML, future research will contribute to many business areas.

REFERENCES

1. Keeney, P. L., 1999. The value of internet commerce to the customer, Management Science, 45: 533-542.
2. Litan, R.E. and M. Rivin, 2001. Projecting the Economic Impact of the Internet, Am. Eco. Rev., 91: 313-317.
3. Barua, A., P. Konana, B.A. Whinston and F. Yin, 2001. Managing E-commerce transformation: Opportunities and value assessment, Sloan Management Review, 34: 36-44.
4. Daikoku, G., L. Shu, E. Cronin, P. Gartzen, L. Leskela and D. Siddall, 2001. The economic downturn is not an excuse to retrench B2B Efforts. Gartner Group.

5. CyBiz - BizNews, 2001. Retrieved July 2, 2005 from http://www.cybiz.com/NewsLetter_June2001.asp
6. Jennex, M.E. and D. Amoroso, 2002. E-commerce and technology issues for developing economies: A ukraine case study. EJISDC. 10: 1-14.
7. Amoroso, D., 2001. E-commerce success factors. Working Paper, San Diego State University.
8. Lin, Z., G.D. Anderson and T.J. Anderson, 2002. Managing multi-organization data for real estate E-commerce on the web– motivations, implementation, and implications. special issue on data management for E-commerce, J. Database Manage.,
9. Goodhue, D. L. And M.D. Wybo, 1992. The impact of data integration on the costs and benefits of information systems. MIS Quarterly, 16: 293-19.
10. Abdulla, K., 1998. A new approach to the integration of heterogeneous databases and information systems. Unpublished Ph.D Thesis University of Miami, Florida pp: .
11. Goh, C., S. Madnick and M. Siegal 1994. Context Interchange: Overcoming the challenges of large-scale interoperable database systems in a dynamic environment. 3rd International Conference on Information and Knowledge Management. (Eds.) In Nabil R Adam, Bharat K Bhargava, & Yelena Yesha. Gaithersburg, MD, pp: 337-346.
12. Holowczak, R.D. and W.S. Li, 1996. A survey on attribute correspondence and heterogeneity metadata representation. Institute of Electrical and Electronics Engineers.
13. Kim, W., I. Choi, I. Gala and M. Scheevel, 1993. On resolving schematic heterogeneity in multi database systems. J. Distributed and Parallel Database, pp: 1-251.
14. RainingData, 2000. Retrieved June 10, 2005, from http://www.rainingdata.com
15. Neild, T.H., 1999. The virtual data integrator: an object -oriented mediator for heterogeneous database integration. Unpublished Ph.D Thesis, Northwestern University
16. Phijaisanit, W., 1997. Dynamic meta-data support for information integration and sharing across heterogeneous databases (federated database). Unpublished doctoral dissertation, George Mason University.
17. Srinivasan, U., 1997. A framework for conceptual integration of heterogeneous databases. Unpublished doctoral dissertation, University of New South Wales.
18. Yu, T.F., 1997. Information modeling and mediation languages and techniques for information sharing among heterogeneous information systems. Unpublished Ph.D Thesis, University of Florida.
19. Parent, C. and S. Spaccapietra, 1998. Issues and Approaches of Database Integration. Communication of the ACM., 41: 166-178.
20. Missier, P., M. Rusinkiewicz and W. Jin, 1999. Multidatabase languages. In A. Elmagarmid & M. Rusinkiewicz and A. Sheth (Eds.), Management of heterogeneous and autonomous database systems. CA: Morgan Kaufmann Publishers, Inc.
21. Aren, Y., C.Y. Chee, C. Hsu and C. Knoblock, 1993. Retrieving and Integrating data From Multiple Information Sources. Intl. J. Intelli. Cooperative Inform. Sys., 2: 127-158.
22. Ensink, B., S.H. Kim, T. Schavey and M. Shrestha, 1999. XML code generator for web database integration. 13th National Conference on Undergraduate Research, Rochester, NY
23. Hughes, J.G., 1991. Object-Oriented Databases. New York: Prentice-Hall.
24. Inmon, W.H., 1996. Building the Data Warehouse. Wiley Computer Publishing. 2nd (Edn.)
25. Jones, C.B., 1998. Cutting Edge Technology-If It Works It's Obsolete. Economic Research Associates, presented at PATA Conference.
26. Pearson, G.S., 2000. Maximizing Security Benefits from Technical Cooperation in Microbiology and Biotechnology. Report of the NATO Advanced Research Workshop Piestany, Slovakia.
27. Carmel, D., Y, Maarek and A. Soffer, 2000. XML and Information Retrieval: A SIGIR Workshop, IBM Research Lab in Haifa.
28. Bray, T., J. Paoli and C.M. SperbergMcQueen, 1997. Extensible Markup Language (XML): Part I Syntax. World Wide Web Consortium Working Draft.
29. Anonymous, 2000. Explaining XML. Harvard Business Review.