Chapter 17
System Architecture and Implementation

Ibrahim Dweib
Sultan Qaboos University, Oman

Joan Lu
University of Huddersfield, UK

ABSTRACT
This chapter presents the system architecture, and implementation tools used for evaluating the MAXDOR model. The chapter also presents the main classes created to demonstrate the methodology for mapping XML document into relational database, rebuilding XML document from relational database, updating the content of XML document stored in relational database, XPath-To-SQL query translation, and building the result in XML format. Application on a case study is also presented. XML data sets from selected XML bench marks and XML data repository will be identified to be used for testing and evaluating the model. Finally, the chapter concludes with a summary.

SYSTEM ARCHITECTURE AND THE USED TOOLS

System Architecture

System architecture consists of four main components each of which represents one of the project requirements. Those components are:

1. **Mapping XML document into relational database**: the system loads the XML document and parses it using XML SAX parser as a sequence of events, shreds the document content into tokens, and inserts these tokens into predefined relational database schema. Detail of the relational schema has been given in chapter 4.

2. **Reconstructing XML document from relational database**: this component goes through the relational tables and reconstructs the requested XML document to check the method for lossless of XML document information in Part one or to exchange or export the document to other location.

DOI: 10.4018/978-1-4666-1975-3.ch017
3. **Updating XML document stored in relational database**: by this component, the user is given the facility to update the XML document stored in relational database. Update includes: inserting new tokens as element or element attributes, delete tokens or tokens’ re-allocation within the document, and modify tokens’ name and values.

4. **Retrieving and querying XML document stored in relational database**: throughout this component, XPath queries are translated into SQL statements. The resultant SQL statements are fired against the database engine so as to retrieve XML data results. The retrieved results are reconstructed as XML hierarchical format and returned to the user. Figure 1 gives an overview of the system architecture.

The above listed components will be tested and evaluated in Chapter 6 for the MAXDOR model described in chapter 4. The following points are taken into consideration during system design; i.e. components of the system should be:

- **Testable against requirements**: every requirement should be easy to test.
- **Structured**: the system structure should be clear, read and its code should be easy and understandable.
- **Reusable**: the system design should be reusable and repeatable.

**Tools Used**

The tools used in the project can be classified into:

1. XML interface: both input and output documents are XML format and relational database technology is a target tool for storing XML documents’ contents and structure; so, the relational database capabilities are used for internal processing of data.

2. XPath or XQuery as source languages provided for users to represent their requests. SQL query language is a target language used against relational database to answer users’ queries. XPath is used for the following two reasons:
   a. XPath is simpler than XQuery, and hence would be better to achieve the objective of testing our model in current situation.
   b. Its structure is included in XQuery, so it is easier to be upgraded into XQuery.

3. Visual Basic 6.0 programming language is used as a tool to create the GUI and to implement the system components. It is used for the following reasons:
   a. VB structure is simple, mainly as to the executable code.
   b. VB is easy for building graphical user interfaces.
   c. VB application is easily connected with Microsoft Access database.

4. Microsoft Office Access is used as a relational database management system (RDBMS). It is used for the following reasons:
   a. It can be easily used with visual basic programming language.
   b. Access database can be easily sited on a website for access by remote users. Simple screens can be built in Access, Data Access Pages. Or it can be employed using Active Server Page (ASP) scripting.

**SYSTEM IMPLEMENTATION**

**Requirements for System Implementation**

1. Microsoft Office 2003 or 2007 is required since we are using Microsoft Access as the development DBMS for the system.
11 more pages are available in the full version of this document, which may be purchased using the "Add to Cart" button on the product’s webpage:

www.igi-global.com/chapter/system-architecture-implementation/69140?camid=4v1


www.igi-global.com/e-resources/library-recommendation/?id=1

Related Content

Gravizor: A Graphical Tool for the Visualization of Web Search Engines Results With Multi-Agent Based Modeling
www.igi-global.com/chapter/gravizor/198602?camid=4v1a

Service Discovery Framework for Distributed Embedded Real-Time Systems
www.igi-global.com/chapter/service-discovery-framework-for-distributed-embedded-real-time-systems/198573?camid=4v1a

Innovative Product Design using Metaontology with Semantic TRIZ
www.igi-global.com/article/innovative-product-design-using-metaontology-with-semantic-triz/130007?camid=4v1a

Innovative Product Design using Metaontology with Semantic TRIZ
www.igi-global.com/article/innovative-product-design-using-metaontology-with-semantic-triz/130007?camid=4v1a