



# INTERNATIONAL JOURNAL OF PURE AND APPLIED RESEARCH IN ENGINEERING AND TECHNOLOGY

A PATH FOR HORIZING YOUR INNOVATIVE WORK

## SQL EDITOR FOR XML DATABASE

MISS. ANUPAMA V. ZAKARDE<sup>1</sup>, DR. H. R. DESHMUKH<sup>2</sup>, A. S. MAHALLE<sup>3</sup>, S. A. KARALE<sup>3</sup>

1. M. E. First Year, Department of Computer Science & Engineering, IBSS College of Engineering, Amravati.

2. Prof and HOD, Department of Computer Science & Engineering, IBSS College of Engineering, Amravati.

3. Asst. Prof, Department of Computer Science & Engineering, IBSS College of Engineering, Amravati.

Accepted Date: 27/02/2014 ; Published Date: 01/05/2014

**Abstract:** XML has emerged as the dominant standard for representing and exchanging data over the Internet. Professional XML editors will help you to write error-free XML documents, validate your XML against a DTD or a schema, and force you to stick to a valid XML structure. For example, a business can easily model complex structures such as purchase orders in XML form. As another example, all of Shakespeare's plays can be marked up and stored as XML documents. With a large amount of data represented as XML documents, it becomes necessary to store and query these XML documents. To address this problem, there has been work done on building native XML database systems. These database systems are built from scratch for the specific purpose of storing and querying XML documents.

**Keywords:** XML- Extensible markup language, SQL – Structured Query Language, SQL Parser



PAPER-QR CODE

Corresponding Author: MISS. ANUPAMA V. ZAKARDE

Access Online On:

[www.ijpret.com](http://www.ijpret.com)

How to Cite This Article:

Anupama Zakarde, IJPRET, 2014; Volume 2 (9): 1052-1060

## INTRODUCTION

XML has emerged as the dominant standard for representing and exchanging data over the Internet. To overcome the first of the above limitations, there have been techniques proposed for storing and querying XML documents using relational database systems. These approaches work as follows. The first step is relational schema generation, where relational tables are created for the purpose of storing XML documents. The next step is XML document shredding, where XML documents are "stored" by shredding them into rows of the tables that were created in the first step. The final step is XML query processing, where XML queries over the "stored" XML documents are converted into SQL queries over the created tables. The SQL query results are then tagged to produce the desired XML result. The wealth of literature in this field makes it clear that there are many possible approaches for relational schema generation.

### FEATURES:

- A] It will provide a visual interface for an XML database.
- B] Users will be able to upload XML Documents into this database.
- C] This Interface will display list of all XML Documents present in database.
- D] When a document is selected it will be opened / displayed.
- F] User can fire a SQL query on the opened document.
- E] The SQL-To-XML converter will automatically convert it into XPATH expression.
- F] The XPATH expression will be used to retrieve data from the document.
- G] The retrieved xml data will be displayed in SQL Editor.

### What is XML?

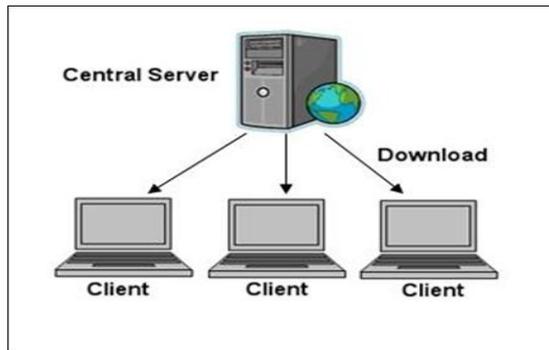
XML is Mark up language. It is Extensible Mark up Language. It is a protocol for containing and managing data. XML is quickly becoming the standard for data exchange over the Internet. This is creating a new set of data management requirements involving XML, such as the need to store and query XML documents.

### What is SQL?

However, the standard SQL commands such as "Select", "Insert", "Update", "Delete", "Create", and "Drop" can be used to accomplish almost everything that one needs to do with a database. SQL (pronounced "ess-que-el") stands for Structured Query Language. SQL is used to communicate with a database. According to ANSI (American National Standards Institute), it is

the standard language for relational database management systems. SQL statements are used to perform tasks such as update data on a database, or retrieve data from a database.

**Client-server model:** In a network, the client/server model provides a convenient way to interconnect programs that are distributed efficiently across different locations. Computer transactions using the client/server model are very common.



In the usual client/server model, one server, sometimes called a daemon, is activated and awaits client requests. Typically, multiple client programs share the services of a common server program. Both client programs and server programs are often part of a larger program or application. Relative to the Internet, your Web browser is a client program that requests services (the sending of Web pages or files) from a Web server (which technically is called a Hypertext Transport Protocol or HTTP server) in another computer somewhere on the Internet. Similarly, your computer with TCP/IP installed allows you to make client requests for files from File Transfer Protocol (FTP) servers in other computers on the Internet.

**SQL parser:** It checks the syntax of sql queries coming from the sql client manager, if query is valid/well formed or if it is correct then it will given to the SQL to X-path convertor, If sql statement is valid then will generate an parse table. Then this parse table will be given to the SQL-To-XPATH Converter.

**PROBLEM ANALYSIS:** We use here SQL editor to fetch data from XML database, in this we will fire the SQL queries on the XML database, after this the appropriate data will be retrieved, firstly we write a sql query in SQL editor then this query will be passed to the SQL-client manager here this manager manages/handle all the queries coming from SQL editor, then it will be passed to the SQL parser. SQL parser is the one of the component of our system. It checks the syntax of sql queries coming from the sql client manager, if query is valid/well formed or if it is correct then it will given to the SQL to X-path convertor, If sql statement is valid then will generate an parse table, Then this parse table will be given to the SQL-To-XPATH Converter.

According to parse table it will generate XPATH expression. This will be given to the X-Path expression Processor it will process the XPATH expression in such a way that this x-path expression will be used to retrieve data from XML database. If client send one or more queries (requests) at the same time to server, then server will take only the first Query means it will get the queries priority wise, but our server can handle such a situation of burdening of queries, when one request is come to server then at the same time when it handle the first request it receives the second request hence we made our server as a multithreaded.

### SYSTEM DESIGN:

#### Modules:

A] Document Manager / XML Database:

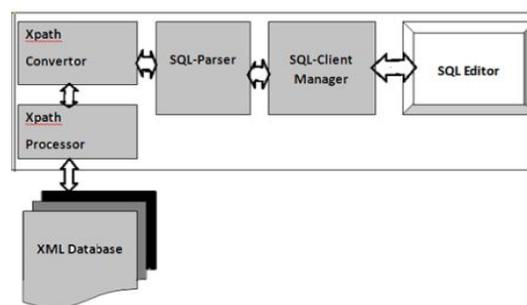
- 1] Maintain XML Documents
- 2] Will store document
- 3] Will Provide document list.
- 4] Will be able to remove documents.

B] SQL Parser:

- 1] Will check syntax of SQL statement.
- 2] If sql statement is valid then will generate an parse table.

C] SQL-To-XPATHConverter: According to parse table will generate XPATH expression.

D] X-Path expression Processor: Will process the XPATH expression to retrieve data from XML document.



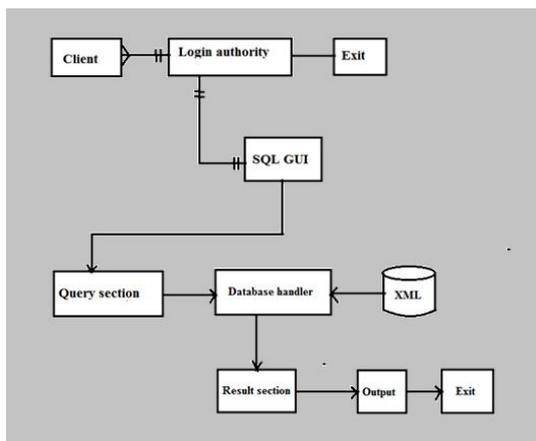
#### DATA FLOW DIAGRAM (DFD):

We have two parts of our system i.e. SQL editor (SQL client) and XML doc/database (XML server) We use here SQL editor to fetch data from XML database, in this we will fire the SQL queries on the XML database, after this the appropriate data will be retrieved, firstly we write a

sql query in SQL editor then this query will be passed to the SQL-client manager here this manager manages/handle all the queries coming from SQL editor, then it will be passed to the SQL parser, SQL parser is one of the component of our system.

It checks the syntax of sql queries coming from the sql client manager, if query is valid/well formed or if it is correct then it will given to the SQL to X-path convertor, If sql statement is valid then will generate an parse table Then this parse table will be given to the SQL-To-XPATH Converter. According to parse table it will generate XPATH expression. This will be given to the X-Path expression Processor it will process the XPATH expression in such a way that this x-path expression be used to retrieve data from XML database. If client send one or more queries (requests) at the same time to server, then server will take only the first Query means it will get the queries priority wise, but our server can handle such a situation of burdening of queries, when one request is come to server then at the same time when it handle the first request it receives the second request hence we made our server as a multithreaded.

#### COMPONENT DIAGRAM:



In the ER diagram of querying xml database using SQL query, the client first long by using his user ID & password. If the user is not registered then, by creating user ID & passed the user can log in. The Login can be done by login authority; if the user ID or password will wrong then the process terminated.

When user has successful login then the control goes to "SQL GUI" which Graphical processing the XLM database by using xml Queries or also by SQL queries. In Query Section, Query get executed .whether the given Query is right or wrong is checked by the query section. Then the Query get executed the by the direction of given Query the Data is fetch from database. Thus Database handler taker the query &by the direction of query fetch data from database which is



### 3] Client connected

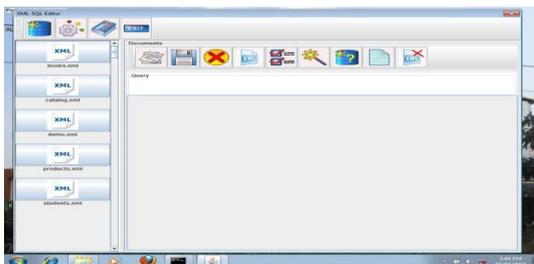


After this we have to connect our client with server which is waiting at port 2020 by typing c in cmd and after typing c type for running the client suddenly a dialog box will create.

### 4] After log in, SQL editor available

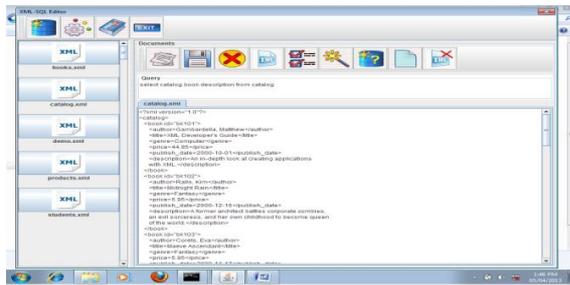


### 5] After opening XML document



### 6] Type a SQL query

For opening the xml document, click on of the xml document on the right side list. After clicking on the xml document, will see that xml file in body part of SQL editor



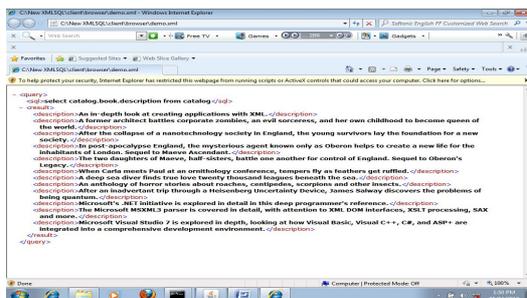
After this we will give the SQL Query in Query Section, for retrieving data from the selected xml document

**7] After giving sql query for xml doc we get this result:**



After giving query we will click on the button having name Sql Query, we will the resulting second tab on the body part which contain only those information which we need, which we specify in sql query.

**8] Click on view button for viewing the result in browser**



**CONCLUSION:**

We have presented a technique for querying XML documents using a SQL queries and using X-path, With a large amount of data represented as XML documents, it becomes necessary to store and query these XML documents, XML has proved hugely successful in the areas of document mark-up, data and meta-data sharing, enabling interoperability, and transparently transporting and storing data. With the current level

of interest in the next generation of enterprise systems, the use of XML is set to grow as it is a core technology to web services, portal development and service oriented architectures. XML is here to stay, and it's future looks bright.

**REFERENCES:**

1. Don Chamberlin, Peter Fankhauser, Daniela Florescu, Massimo Marchiori and Jonathan Robie, "XML Query Use Cases", W3C Working Draft, 15 November 2002. <http://www.w3.org/TR/xmlquery-use-cases/>
2. V. Christophides, S. Cluet, J. Simeon, "On Wrapping Query Languages and Efficient XML Integration", SIGMOD Conference, Dallas, Texas, June 2000.
3. World Wide Web Consortium, "X Query: A Query Language for XML", W3C Working Draft, February 2000.
4. XML related information like xpath, xquery <http://www.w3schools.com/default.asp>