



XML Document Management Tools Survey

Vijay Sonawane¹ and D.Rajeswara Rao²¹Research Scholar, Department of CSE, K.L.University, Green Fields, Guntur, Andhra Pradesh.²Professor, Department of CSE, K.L.University, Green Fields, Guntur, Andhra Pradesh.

ARTICLE INFO

Article history:

Received: 24 April 2015;

Received in revised form:

19 May 2015;

Accepted: 25 May 2015;

Keywords

XML,

XML tools,

Change management,

XML DBMS.

ABSTRACT

Now a day's large volume of information is available online. An efficient and effective management of the information on the web is not only valuable to individual users but also to business organizations, especially for decision making purposes. Here challenging task is amongst available information on the web finding business values. The significance of organizations information depends on the applications that deal with it. An organization allows the customer to query online published products information and collects their reviews about the products. XML guarantee about those features of business functions including integration, intelligence and reuse of the content. To complete XML assurance it is very important to maintain XML documents properly. In this paper provides discussion and comparison of various commercial tools and database management systems to create, store, maintain, and query XML documents.

© 2015 Elixir All rights reserved.

Introduction

In today's information era large amount of information is contained online. To get the knowledge useful for business purpose from available online information it is important to manage and maintain those informative documents properly. Online information is useful from the perspective of service provider and service users. Increasing success of XML allows using XML to achieve such business functions including content integration, intelligence and salvage. To achieve those benefits, it is very important to maintain the XML documents properly and know how to efficiently and effectively utilize information in it. XML is often called self-describing data as it is designed in a way that the schema is repeated for each record in the document. XML is used to create vocabularies by storing and exchanging metadata for information exchange on the internet [1]. Changes in XML documents are depend on the application that uses it [2]. Unlike static XML documents, dynamic XML documents changes its structure and content over a time. Applications that use and exchange data in XML format is increased so demand for XML document multi-version support is also increased. It also demands to store and maintain XML documents for future reference. Unlike static XML documents dynamic XML documents need more careful handling strategies. Dynamic XML documents are useful in many fields of information maintenance and management. Need to maintain XML document version is increased but storing all the versions of XML document is not effective as it may create redundancy, which may create the problem in searching and querying system on evolving documents.

This paper discusses various commercial tools which are used to create, store, maintain and query XML documents. Here we also provide comparative analysis of those tools based on various features like change detection, support for versioning, concurrent editing support, document validation support, language support, schema update support, differencing support etc.

Background

Now XML comes into picture as a mission critical platform, with more business applications and processes dependent on the

technology every day, there is a growing need for tools to manage the rapidly expanding volume of XML data and in particular to manage stream of change. A tool is needed which will not only identify changes accurately but also it will represent them in a way which is consistent with an organisation's needs and make enable to process those changes easily. XML is now new standard for data exchange for many organizations across the world. It is used as accepted standard for managing the content of websites and in business to business data exchange. So the volume of XML data increased and with it challenges of change management are also increased. A flexible change control solution is needed to accommodate the multiplicity of XML control requirements, which would be easy to integrate in to existing XML infrastructures and which has the performance and scalability to identify change in large amounts of data and evolving XML file sizes[6]. Any solution also has to be open standards compliant to ensure that it will be able to handle today as well as tomorrow's data and it should be capable of representing changes in a way which is easy to process.

XML Overview

Web browsers use HTML to compose textual, graphical, audible, and visual material in the form of Web pages. HTML has some limitations of fixed set of elements for representing the Web content and its representation. Therefore, to overcome those limitations W3C designed XML (Extensible Markup Language) which is simpler and has self-describing nature. Its format is same as the HTML but more flexible and also used for data exchange and support external data confirmation. It was very difficult with HTML to work with rapidly changing data. The XML "Schema" or "Document Type Description" (DTD) defines a structure of XML documents and helps applications to authenticate imported data. Several standards are available for data sharing and management in XML such as Extensible Stylesheet Language (XSL) to transform an XML document and Document Object Model (DOM) to allow programs to dynamically access or update an XML document. XML and database technology are more corresponding. The task of managing loads of XML data has increased the demand to discover the integration of XML with database technology.

XML databases are equally beneficial for content-centric and change-centric applications. The change-centric applications such as software configuration and cooperative work may exploit dynamic XML documents since the content and structure of a dynamic XML document changes continuously depending on the underlying application, such as the content of a website of an e-commerce web site. Whenever the XML document content changes then mainly two things happen: a) Document size grows b) New version of the same document is created.

Increasing new versions of XML documents are creating need of various version control scheme to manage document content. Most of the time only small changes in the document may creates the new version. XML documents also contain temporal information such as relational databases published in XML [3] and cultural heritage [4].

Commercial Tools

Here we discuss in brief the features of some commercial tools which are used to create, store, maintain and query XML documents. Those provide various features like change detection between XML documents, collaborative editing, versioning support, and validation support. But they will not provide support for temporal XML documents.

1. **Altova's XMLSpy** – It has powerful support for working with all XML-based technologies. It's having unique ability to generate eye catching charts based on XML data. The advanced functionality in XMLSpy is coupled with user-friendly views and entry helpers, wizards, and debuggers designed to create, edit, and optimize today's cutting-edge XML-based applications. With advanced functionality for modeling, editing, transforming, and debugging XML related technologies, the XML editor is well suited for any development task. The XML editor delivers the power needed to create the most advanced XML and Web applications at the same time it allows to with any XML technology in a way that best suits the complexity of the document and user preferences.

2. **Altova's DiffDog**- It is having powerful XML Schema differencing functionality that extends and accompaniments XML-aware file comparison of .xsd documents and eases conversion of XML data. XML developers' often need to modify and enhance XML Schemas as XML-based applications are created. Similarly, many industry standard XML-based data exchange systems need to update XML file formats to maintain compatibility with evolving standards. DiffDog XML schema compare tool along with its XML file comparison capabilities helps to you efficiently update XML data files when XML Schemas change. For this, it automatically maps schema elements with the same names. Its output can be manually modified. It cannot incrementally alter schemas.

3. **DeltaXML Sync** - DeltaXML Sync is a system which performs an intelligent, XML aware merge of three XML datasets into single synchronized XML document. It can apply rule sets to produce different kinds of results for a number of different kinds of applications. Rule sets can be symmetrical (all the input documents are treated as the same way) or asymmetrical, e.g. concurrent-edit which needs to know which document is the 'base document'. It uses DeltaXML Core matching technology so that 'deltaxml:key' is available for fine control of the merge process.

4. **LiquidXML Studio** - Liquid XML Studio is an advanced graphical XML editor containing all the tools needed for designing, developing and testing XML applications obeying with the W3C standards. Liquid XML studio includes many

features like graphical XML Editor, XML Schema Editor, XML Data Mapper, XPath builder, XSLT and XQuery Debugger, WSDL Editor, Web Service Tools, integration with Microsoft Visual Studio and much more, but it is unable to handle reordering of elements.

5. **oxygen's XML editor** - It is the cross platform tool. It is only XML tool that supports all the XML schema languages. In this tool XSLT and XQuery support is enhanced with powerful debuggers and performance profilers. User can use oXygen XML Editor to work with all XML-based technologies, including XML databases, XProc pipelines, and web services. But it provides only a limited versioning support. It can support concurrent edits on a document, validation, and reordering of elements.

6. **SysOnyx's xmlDraft** – This tool provides versioning and document validation support. It is a schema editor offers a very intuitive way to create W3C XML Schemas. It displays a XML document tree. It has many advantages:

1. Visual XML Design –

XML Draft's offers a unique approach to schema design, allowing users to visually design the XML tree as it should look while the application creates the schema.

2. Integrated Development Environment (IDE) –

In addition it has unique ability that it displays to the user the XML documents that the schema is defining. It is complete IDE for XML schema creation and modification.

3. Easy XML Schema Creation –

xmlDraft allows for easy Schema creation through many means. It's visual XML Tree editing, the editor itself, and the ability to import existing XML Documents to automatically create schemas.

4. Documentation for XML Documents

XML schemas of document are not so easy to read as are considered documentation for XML Documents. XMLDraft has built in abilities to export xml schemas into readable HTML documents. It allows keeping XML Documents documented conveniently.

Table 1 shows summary of the comparative analysis of the various tools provided by Altova, Stylus Studio, oXygenXML, LiquidXML, SysOnyx, and XMLmind.

Commercial DBMS for XML

Both types of tools, XML native such as Tamino and XML enabled including Oracle, IBM DB2, and SQL Server DBMSs, are available for storing and querying XML documents. These both types of tools are used for storing and querying XML documents. Native DBMSs rely on a data model specifically conceived for the management of XML, whereas enabled DBMSs are relational or object-relational which have been extended for the treatment of XML. Enabled XML DBMSs are more mature than the native ones because they are supported by big vendors and the integration of XML data with other company data is easier.

Some of the famous databases such as SQL Server, DB2, Tamino, and Oracle 11G provide support for the storage of XML data validated against an XML schema.

In IBM DB2, schema validation occurs on a per document basis. The schema is determined by attributes within the document to be inserted, or by manually specifying a schema as an argument to the XMLValidate function.

Once a document has been validated, it is adorned with metadata that verifies this validation as well as contains information to help optimize query processing.

Table 1. Comparisons of various commercial tools for managing xml documents

| Tool | Versioning support | Temporal support | Manage concurrent edits | Support schema updates | Supported language | Document validation | Differencing support | Difference shows in | Consider document order |
|----------------------|--------------------|------------------|-------------------------|------------------------|---------------------|---------------------|----------------------|--|-------------------------|
| Altova's XMLSpy | No | No | No | Limited Support | XQuery, XSLT, XPath | Yes | Yes | Comments, elements, text formatting, attribute, processing instruction | Yes |
| Altova's DiffDog | | | | | | | | | |
| DeltaXML Sync | No | No | Yes | No | – | No | Yes | All XML documents | No |
| LiquidXML Studio | No | No | No | No | XQuery, XPath | Yes | Yes | Comments, elements, text formatting, attribute, processing instruction | No |
| Oxygen's XML Editor | Yes | No | Yes | No | XQuery, XSLT, XPath | Yes | Yes | | Yes |
| Stylus Studio | No | No | No | No | | Yes | Yes | | No |
| SysOnyx's xmlDraft | No | No | No | Yes | – | Yes | No | – | Yes |
| XMLmind's XML Editor | No | No | No | No | XPath | Yes | Yes | Attribute, text, elements (constrained) | No |

Table 2. Comparative of various commercial dbms

| DBMS | Support validation | Support document modification | Support schema evolution | Versioning support of schema/document | Schema type |
|-------------|--------------------|-------------------------------|--------------------------|---------------------------------------|---------------|
| DB2 | Yes | Yes | Yes | Schema | XSD |
| Oracle 11 g | Yes | Yes | Yes | Document | XSD |
| SQL Server | Yes | Yes | No | Schema | Subset of XSD |
| Tamino | Yes | Yes | Yes | Document | Subset of XSD |

In contrast to Oracle, Tamino allows the stored data to determine reverse compatibility rather than the schema document versions themselves.

One can pass a parameter to the define command to attempt to do some static validation first determining just from the documents themselves whether it is possible for reverse compatibility to be guaranteed but eventually all documents are validated against the new schema at evolution time and, if any validation fails, the change is rejected. Similar to Oracle, Microsoft SQL Server stores a collection of homogeneous XML documents in a relation column. Instances in an XML-typed column or table in Oracle must conform to a specific schema with a specific global element as root, whereas an XML-typed column in SQL Server validates against any schema in a collection of schemas and allows any global element as root. The schema evolution replaces the old schema with a new one whereas the schema modification allows changes in the existing schema while the validity of schema remains unaffected. Oracle offers two ways to evolve an XML schema (1) copy-based (2) in-place evolution.

The copy-based mechanism allows a great deal of flexibility. Data from an XML document collection is copied to a temporary location, transformed according to a specification, and, finally, placed in its original location. The in-place evolution does not require any data copying but only supports a limited set of possible schema changes such as add a new optional element, attribute, or domain value to an enumeration, global element, global attribute, or global type. SQL Server provides the ALTER statement to evolve the XML schema without manually dropping the schema, manually translating instances, and reestablishing the schema.

In DB2, different versions of the same schema are stored as unconnected documents. It does not update instances of a document from one version of a schema to another. In [5] it has been described how to support schema versioning in DB2. It supports a scenario that involves a relational table to correlate

the schema versions. It also allows multiple schema versions to coexist in a single corpus. Documents exist in their original form, associated with their original schema. Tamino allows schemas to evolve under the same restrictions as Oracle's in-place migration mechanism.

Table 2 provides comparative analysis of SQL Server, DB2, Tamino, and Oracle 11 g, based on validation, content modification, schema evolution, content versioning, and schema versioning support.

Conclusion

Data are stored over the internet in the form of XML documents to maintain the critical information. The huge storage of data in XML form demands the efficient management of this data. Such data constantly evolve over time and may require querying over the previous versions of an XML document. In contrast to static XML documents, the temporal, unpredictable, and changing nature of dynamic XML documents requires different handling techniques. The management of multiversioned XML documents hot area. In this paper we discussed features and comparative analysis of various commercial tools and DBMS which are used to create, store, maintain and query XML documents. Comparative analysis of various tools we provided helps the researcher in choosing the tool.

References

1. Pokorny, J. XML data warehouse: Modelling and querying. Kluwer Academic Publishers, pp. 67–80, 2002.
2. Dyreson, C, and Grandi, F, "Temporal XML" Encyclopedia of Database Systems, pp.3032–3035, 2009.
3. Wang, F., & Zaniolo, C. "Temporal queries in XML document archives and web warehouses" In Proceedings of the tenth international symposium on temporal representation and reasoning and fourth international conference on temporal logic pp. 47–55, 2003.

4. Grandi, F, XML representation and management of temporal information for web based cultural heritage applications, *Data Science Journal*, 1, pp. 68–83, 2002.

5. Beyer, K., Oezcan, F., Saiprasad, S., & Van der Linden, B. “DB2/XML: Designing for Evolution”, In *Proceedings of the ACM SIGMOD international conference on management of data* pp. 310–321, 2005.

6. Sidra Faisal, Mansoor Sarwar., “Temporal and multi-versioned XML documents: A survey” *Journal of information processing and management* (50). pp.113-131, 2014.

Further web readings:

Altova XML Spy 2012 Tutorial.
<<http://www.altova.com/documents/XMLSpyTutorial.pdf>>.

Berkeley DB.
<<http://www.oracle.com/technology/products/berkeley-db/index.html>>.

<http://www.deltaxml.com/index.html>.

<http://www.sysonyx.com/xml/Products/xmlDraft/features.asp>.

Introduction to DeltaXML Sync.
<<http://www.deltaxml.com/library/using-sync.html>>.

Liquid XML Getting Started.
<<http://www.liquidtechnologies.com/XmlDataBinding/GettingStarted.pdf>>.

Stylus Studio User Guide.
<<http://www.stylusstudio.com/doc/StylusStudio2011UserGuide.pdf>>.

SVNKit – The only pure Java Subversion library in the world!
<<http://svnkit.com/>>.

User Manual of Oxygen XML Editor 13.2.
<<http://www.oxygenxml.com/doc/Editor-UserManual.pdf>>.

XML mind XML Editor: Features.
<<http://www.xmlmind.com/xmleditor/features.html>>.

Bibliography of Authors



Vijay Sonawane obtained Bachelor Degree in Information Technology from North Maharashtra University (MS), India. Then he earned Master in Technology (Computer science) from Shivaji University, Kolhapur (MS), India. He is currently PhD Research scholar in computer science and engineering in

K.L.University, Vijayawada. He is working as Assistant professor in Sandip Institute of Technology and Research Centre, Nashik. His major area of interest is Data mining, Information retrieval, web information management. He published various papers in Journals and Conferences.



D.Rajeswara Rao :He is a Professor in department of CSE in K L University, Vijayawada. He is acting as heads of knowledge engineering research group and Research policy advisory committee in K.L.University. He is He had published various research papers at National and International Journals/Conferences in the area of

data mining, soft computing.