© 2015 IJSRSET | Volume 1 | Issue 2 | Print ISSN : 2395-1990 | Online ISSN : 2394-4099
Themed Section: Engineering and Technology

XML Enabling Homogeneous and Platform Independent Data Exchange in Agricultural Information Systems

Bhavesh Kataria
Department of Computer Engineering and Information Technology, LDRP Institute of Technology and Research, Gandhinagar, Gujarat, India

ABSTRACT

At present, there are large quantities of domain specific agriculture databases are available and farmers are accessing these data, in addition many agricultural information systems are also exchanging data; It could be broadly used if databases and computer programs were friendly-user and accessible through Internet in any platform. The Documents XML (extensible markup language) are being widely used for the storage and exchange of data and XSLT documents (extensible stylesheet language transformation) are being developed for transforming XML documents. Through XML the exchange of data during application and documentation processes, such as agricultural supports, traceability, and quality assurance are simplified and automated, so once entered into the system, is available for all data exchanges and efficiency of data processing is improved for all involved in agricultural production.

Keywords: Agriculture, Farmers, XML, XSLT, ADO, JSON, Java Technology, Operating System, Web Services

I. INTRODUCTION

In Agricultural Information Systems documentation of agricultural practices is becoming more and more of an issue for farmers. They are increasingly obliged to it by legislation and other side integrative planning of agricultural production requires through information about measures and events in the past. For data exchange processes in agriculture up to now only individual interfaces between different communication partners were available. Even if the farmer had electronic systems to record production data, the required data had to be transferred by hand from one software to another or from screen into paper forms.

The eXtensible Markup Language (XML) was developed by an XML Working Group formed under the auspices of the World Wide Web Consortium (W3C) in 1996. XML documents are made up of basic units called “elements” according to a set of restrictions specified on an independent specification called Document Type Definition (DTD) (BRAY et al., 2004).

XML is becoming the standard data exchange format among Web applications, providing interoperability and enabling automatic processing of Web resources. An XML document is a hierarchically structured and self-describing piece of information, and consists of atomic elements or complex elements (ABITEBOUL et al., 2000).

The XML documents can easily be transformed to a webpage by means of the Extensible Stylesheet Language Transformation (XSLT). It specifies the

presentation of XML documents by describing how their instances are transformed to an XML document that uses a formatting vocabulary, such as HTML, that can be interpreted by any standard browser (CLARK, 1999).

XML Features for Data Exchange

Extensible Markup Language (XML) is a markup language that defines a set of rules for encoding documents in a format which is both human-readable and machine-readable. The main features or advantages of XML are given below.

1) XML Separates Data from HTML: If you need to display dynamic data in your HTML document, it will take a lot of work to edit the HTML each time the data changes. With XML, data can be stored in separate XML files. This way you can focus on using HTML/CSS for display and layout, and be sure that changes in the underlying data will not require any changes to the HTML. With a few lines of JavaScript code, you can read an external XML file and update the data content of your web page.

2) XML Simplifies Data Sharing: In the real world, computer systems and databases contain data in incompatible formats. XML data is stored in plain text format. This provides a software- and hardware-independent way of storing data. This makes it much easier to create data that can be shared by different applications.

3) XML Simplifies Data Transport: One of the most time-consuming challenges for developers is to exchange data between incompatible systems over the Internet. Exchanging data as XML greatly reduces this complexity, since the data can be read by different incompatible applications.

4) XML Simplifies Platform Change: Upgrading to new systems (hardware or software platforms), is always time consuming. Large amounts of data must be converted and incompatible data is often lost. XML data is stored in text format. This makes it easier to expand or upgrade to new operating systems, new applications, or new browsers, without losing data.

5) XML Increases Data Availability: Different applications can access your data, not only in HTML pages, but also from XML data sources. With XML, your data can be available to all kinds of "reading machines" (Handheld computers, voice machines, news feeds, etc), and make it more available for blind people, or people with other disabilities.

6) XML can be used to Create New Internet Languages

A lot of new Internet languages are created with XML. Here are some examples:

- **XHTML**
- **WSDL** for describing available web services
- **WAP** and **WML** as markup languages for handheld devices
- **RSS** languages for news feeds
- **RDF** and **OWL** for describing resources and ontology
- **SMIL** for describing multimedia for the web

II. METHODS AND MATERIAL

Extensible Markup Language (XML) provides a nonproprietary way to label data objects. These objects may be elements of a database, pieces of a Web page (including links, text, numbers, meta data and graphics) or the contents of a spreadsheet -- almost any form of coded content. Combined with style sheets and Java programs or other program, XML allows sharing and use of data across applications, organizations and platforms. It supports authentication, advanced data handling (including realtime updates and superior searching) and personalization of output. XML is easily applied to any implementation that involves filling out forms and is expected to replace electronic data interchange (EDI), which is widely used in many industries.

How this Happens : HyperText Markup Language (HTML) is the common language of the Web, but it has important limits, particularly as the Web becomes more interactive, with applications such as e-commerce, e-agriculture. One key limit is the lack of information HTML provides about content. Its focus is on format (font, layout, color, etc.) rather than meaning. This is one of the reasons why search engines, which rely on keywords, can be so inefficient. Ambiguities, such as a "Biogas" being both a sword and a car, are inherent to HTML. The second limit is the inflexibility of HTML,

which has one standard set of markups to which all users, in theory, must adhere. XML is an attempt to resolve these difficulties. First, its focus is data, rather than format. XML looks at a Web page and virtually any other electronic document as a collection of data objects. Cutlass can be unambiguously marked up in readable text as, for instance, <fuel>Biogas<fuel>. Second, XML is extensible, so customized sets of markup tags (vocabularies) can be developed for specific agriculture domain, industries or even individuals.

A. Design of XML-based Data Exchange of Heterogeneous Database

XML has the formidable data performance ability, strong ability of self-description and expandability, while the realization of the separation of data and the form of expression. These characteristics decided that XML can be used as media of data exchange, thus realizing the information exchange of heterogeneous system. The study of data exchange of heterogeneous database based on XML by nature is to actually study the mutual conversion between the XML model and relational database schema. For accessing data from heterogeneous databases, Client/User just needs to specify the desired data without taken care for Data Schema, data extraction, data synthesis and data collection method. User can access heterogeneous databases transparently through accessing sharing platform.

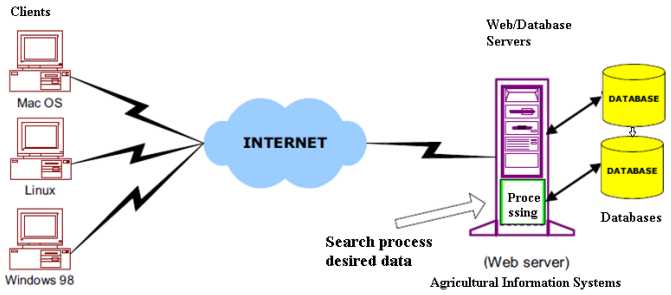

Figure 1 : Accessing data of agriculture Information System from different platform

As shown in fig. 1 how data can be translated in different platform. Most of the agriculture information database is distributed in the agricultural research institutes and agriculture sectors. Databases of all departments are independent entities which have the characteristic of independence. In the sharing, users only have the privilege of query, can't modify the sharing database. In such system we can share data through C/S mode (Client/Server) therefore three tire architecture adopted here as shown in fig. 2.

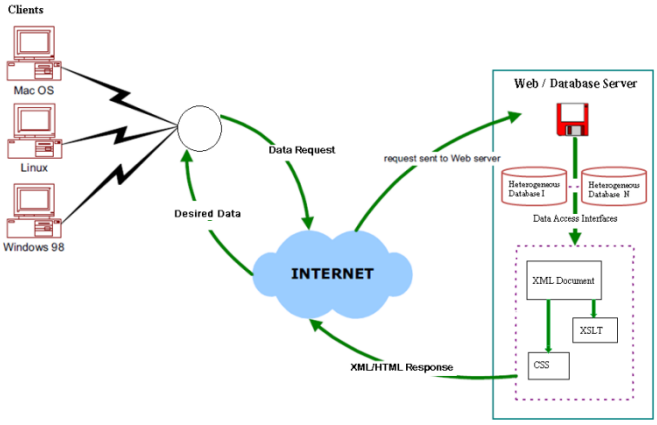

Figure 2 : Three Tire Architecture of Agriculture Information System

(1) User layer

Through web browser clients can submit query to Agricultural Information Systems.

(2) Middle layer

A Web services middle layer established between multiple heterogeneous data sources and users acting as the communication platform for sharing information, which connect with heterogeneous data sources by using data access technology and so on. The layer is the core part of the three-layer structure, which is responsible for receiving the remote or local of the query request and then converts the query into XML documents which will be uploaded to the application server. The application server deal with the logic request, deliver the corresponding SQL statement to the database server and obtain data from heterogeneous database; The query results are converted into XML documents, and translated into HTML or a variety of Script language, and then sent to the query browser. Similarly, the browser will also send request of changing, deleting and adding records to Web server, Web server complete these tasks through accessing the database.

(3) Data layer

Data layer consists of heterogeneous databases and files.

Model of Converting Heterogeneous Database Data into XML Data

In order to provide with user heterogeneous data of variety of different structures in a standardized format, the first thing we should do is to convert these data into

standard XML data model format. Conversion process is shown in Figure 3:

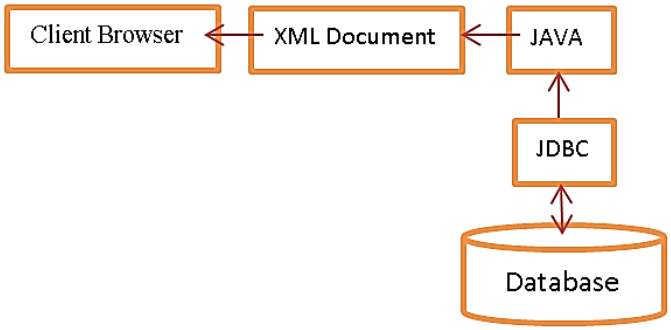

Figure 3 : Steps for Data Conversion into the XML document

The reference model of converting the query results for heterogeneous database into XML data through Web. The application program of Java extracts data from the database through JDBC, convert JDBC Resultset into XML document and then pass them to the Web server. The data is processed and displayed through Web browser as soon as data is stored in the server with XML standard format. The XML is shown exactly in DTD-defined format on client browser, so user can access heterogeneous database through Web.

B. Sharing Data through Web Services

A web service is a collection of open protocols and standards used for exchanging data between applications or systems. Software applications written in various programming languages and running on various platforms can use web services to exchange data over computer networks like the Internet in a manner similar to inter-process communication on a single computer. This interoperability (e.g., between Java and Python, or Windows and Linux applications) is due to the use of open standards.

Web services are XML-based information exchange systems that use the Internet for direct application-to-application interaction. These systems can include programs, objects, messages, or documents.

To summarize, a complete web service is, therefore, any service that:

- Is available over the Internet or private (intranet) networks
- Uses a standardized XML messaging system
- Is not tied to any one operating system or programming language
- Is self-describing via a common XML grammar
- Is discoverable via a simple find mechanism

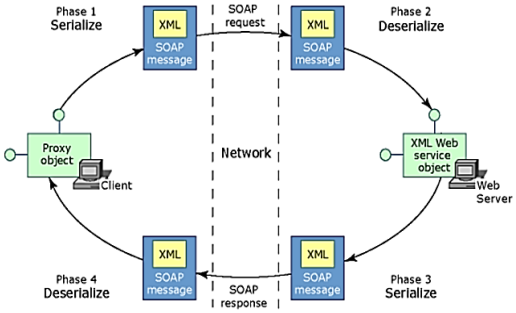

Figure 4 : Web Services

C. XML - Managing Data Exchange/AJAX/JSON

AJAX is nowadays one of the most common used words in the WEB 2.0 era. While the historic remains of it are not really clear (similar logic to manipulate parts of a webpage was already thought of as DHTML, long before the term AJAX existed and surprisingly even using some type of DOM later on) it is now one of the most important technologies used by modern web designers.

But what does AJAX mean? - In short, AJAX stands for Asynchronous JavaScript and XML. It describes a concept of asynchronous data transfer (here: data encapsulated in XML) between the client (usually a web browser) and a server to only exchange/ alter a part of the webpage without the need of a full page reload. That means the browser will issue an XMLHttpRequest in the background and receive only a part of the page - usually tied to one or more html-tags holding uids.

JSON: Javascript Object Notation (JSON) is a fancy name for a simple idea: A web page can download data stored as javascript variables. It's caused a buzz in the tech world because JSON is much easier to load, read and manipulate compared to XML. Also, parsing large XML files can be a large performance hit – JSON gives you an object, packaged up and ready to go.

III. RESULTS AND DISCUSSION

The relational data source converts into XML data source using JAVA or other programming, rather than creating the static XML document. In addition The

XML document can generate dynamically from database and this can update automatically along with the update of database.

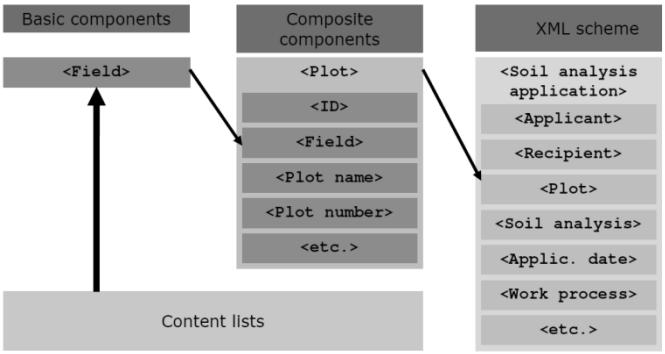

Figure 5 : Sample: Elements of XML

As shown above component of agroXML are the content lists. They provide the functionality of XML Schema enumerations; however the mechanism of how they are included in the schema allows adding to their content dynamically without effecting a change in the schema itself. In addition, they not only contain the enumeration values themselves but also a name and a description of the item at hand. The lists conform to a unified schema and can be downloaded at http://www.agroxml.de/ content. Several lists exist containing e. g. soil types, machine types, fertilizer types, pesticides and plant variety names.

Sample : Example of XML Document of Weather Information

```
<?xml version="1.0" encoding="ISO-8859-1"?>
<weather>
<ph>1.2</ph>
<temp>35</temp>
<humidity>3.5</ humidity >
<date>01-01-2010</ date>
<time>01-01-2010</ time>
<body>Due to weather condition It may effect on small
crops </body>
<weather>
```

XML as a data exchange middleware can well achieve the conversion of data between the various databases, so as to achieve the purpose of data sharing and information exchange.

IV. CONCLUSION

As with any hot new technology, XML has generated exaggerated claims. In reality, XML does not come close to eliminating the need for database management systems or solving large organizations' data sharing problems. The heterogeneous database based on XML between information exchanges has all come true. Seen from the implementation process of the entire heterogeneous database exchange, XML as a data exchange middleware can well achieve the conversion of data between the various databases, so as to achieve the purpose of data sharing and information exchange.

V. REFERENCES

[1] Qiulan Wu, Yongxiang Sun, Xiaoxia Yang, Yong Liang,Xia Geng, "Study on XML-based Heterogeneous Agriculture Database Sharing Platform", School of Information Science and Engineering, Shandong Agricultural University, Taian, Shandong Province, P. R. China.

[2] Mario Schmitz, Daniel Martini, Martin Kunisch, Hans-Jurgen Mosinger, "agroXML: Enabling Standardized, Platform-Independent Internet Data Exchange in Farm Management Information Systems", Shandong Agricultural University, Taian, Shandong Province, P. R. China.

[3] Ms. Rezayat,M.: Knowledge-based Product Development Using XML and Computer-A Design, Issue 32, pp.299-309.(2000)

[4] Hao Shaohua, Han Xie. Heterogeneous Relational Database Integration Model Based on XML Technology.Computer Engineering and Design, 2010,31 (24), pp. 5285-5288

[5] Andreas M. Weiner, Theo Härder. An integrative approach to query optimization in native XML database management systems.IDEAS '10 Proceedings of the Fourteenth International Database Engineering & Applications Symposium.2010,3, pp. 64-74

[6] Shi Yeling, Li Jianhui. Relational Database Schema to XML Schema Generic Mapping Model. Computer Engineering. 2009,35(7). pp: 35- 51

[7] Geißner, G., Grimm, S., Frisch, J.: Geodaten in der Landwirtschaft - Bereitstellung mit agroXML. In: agroXML-Informationstechnik f¨ur die zukunftsorientierte Landwirtschaft. Volume 454 of KTBL-Schrift. Munich (2007) 103–118

[8] World Wide Web Consortium: XML Linking Language (XLink) Version 1.0. (2001) http://www.w3.org/TR/xlink

[9] Abiteboul, S, Buneman, P. Suciu, D., 2000. Data on the Web, Morgan Kaufmann, in: Dalamagas, T., Cheng, T., Winkel, K., Sellis, T., 2006. A methodology for clustering XML documents by structure. Information Systems 31, 187-228.