

Studies and Implementation of Data Interface based on Domino Application System

Zhihong Ruan^{1,a}, JunFeng Zhang^{1,b}, Yi Jin^{2,c}

¹School of Education Science, Guizhou Normal University, Guiyang, Guizhou, 550001, P. R. China

²Department of Computer Science Guiyang University, Guiyang, Guizhou, 550001, P. R. China

^asmall_russet@126.com, ^bgzsd2008@126.com, ^c16279138@qq.com

Keywords: Domino, Data interface, Application system, XML

Abstract. For solving the shortages of the existing methods of data exchange among the various enterprise information systems, in this paper, the XML documents were used as the carrier of structured data to design and implement a new type of data interface based on Domino application system. By using FTP service to communicate, the loose coupling architecture was built to reduce the dependence and complexity of systems, and the security of the systems was improved. The new type of data interface is more efficient and more practical, and the environment of application systems will be also more efficient and more flexible by using it.

Introduction

At present in many enterprises the application systems based on Domino are used widely. Compared with other relational database systems, this kind of Domino-based application system uses text-type database.

During the process of data exchange and sharing among these databases, general methods are used basically in the following: Lotus domino's ODBC Interface, Lotus LEI tools and Lotus ESB interface. But the shortcomings of these conventional solutions such as the high cost, the platform choosing and unstructured data can not be handled and so on. Therefore, to analyzed and implemented the interface for solving these problems mentioned above is the focus of this paper.

Principle of Interface Development

Principle of Text Type Database Generated XML

There are two methods in Domino environment to process XML, one is using Java agent and the other one is using Lotus Script agent.

(1) For the method of Java Agent to generate the XML document. Because the built-in explanation of the XML capabilities of Domino Notes system, by bringing in the java packages, XML files can be generated through the DOM method. By using this way, there are no specific requirements for the operating system in which the application system run. As the method to generate xml file by using dom, so the special characters of some data fields can be explained automatically by the system.

(2) For the method of using Lotus script agent, XML files can be generated by lotus script invoking MSXML component, as the first method, the special characters of data fields of the XML files can be explained automatically by the system too, but there is one limitation for the operating system—it is effective only for the Windows operation system. Or the XML character string can be spliced directly by using lotus script, and then it can be written into the xml files.

Interactive Data of Domino-based Application Systems

The method of Domino application system to generate XML files mainly is used to help the Domino-based application systems generating interactive data which other application systems need. For the interactive data generated by other application systems, Domino-based application systems will be analyzed.

There are two methods for other application systems to generate interactive data; both of them can be explained as the following below:

Method 1: Other application systems can invoke the application web agent of Domino by using the method of Http&Get ; there are parameters which need to be delivered within URL. Then those parameters will be handled by the application web agent of Domino.

Method 2: XML files can be generated by other application systems, then can be delivered to servers which Domino application systems run in, and those XML files were read, analyzed by Domino application systems with timing java agent.

Analysis and Generation Method of Other Application Systems

The frameworks of other application systems basically divided into three types: (1) java and relational database (2) .NET and relational database (3) PHP and relational database.

For the systems structured by java, there are no specific requirements to operation systems, either Windows or Unix systems. XML files can be analyzed and handled by the following methods:

(1) DOM (Document Object Model) Method

DOM method is a collection of hierarchical organizational nodes or pieces of information. It is a cross-platform and language-independent convention for representing and interacting with objects in HTML, XHTML and XML documents. Aspects of the DOM may be addressed and manipulated within the syntax of the programming language in use. The public interface of a DOM is specified in its application programming interface (API), Figure 1 shows the method of DOM.

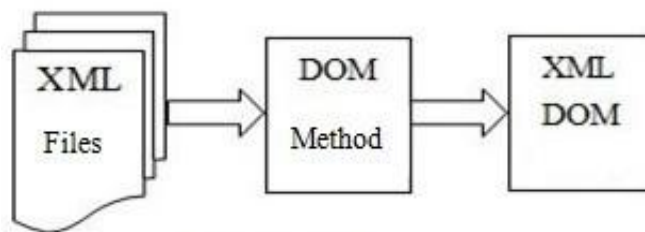


Fig.1 DOM Method

(2) SAX (Simple API for XML) Method

SAX (Simple API for XML) is a sequential access parser API for XML. SAX provides a mechanism for reading data from an XML document. It is a popular alternative to the Document Object Model (DOM).

SAX parsers have certain benefits over DOM-style parsers. The quantity of memory that a SAX parser must use in order to function is typically much smaller than that of a DOM parser. DOM parsers must have the entire tree in memory before any processing can begin, so the amount of memory used by a DOM parser depends entirely on the size of the input data. Because of the event-driven nature of SAX, processing documents can often be faster than DOM-style parsers.

Memory allocation takes time, so the larger memory footprint of the DOM is also a performance issue. Due to the nature of DOM, streamed reading from disk is impossible. Processing XML documents larger than main memory is also impossible with DOM parsers but can be done with SAX parsers. However, DOM parsers may make use of disk space as memory to sidestep this limitation. The method of SAX is shown in Figure 2.

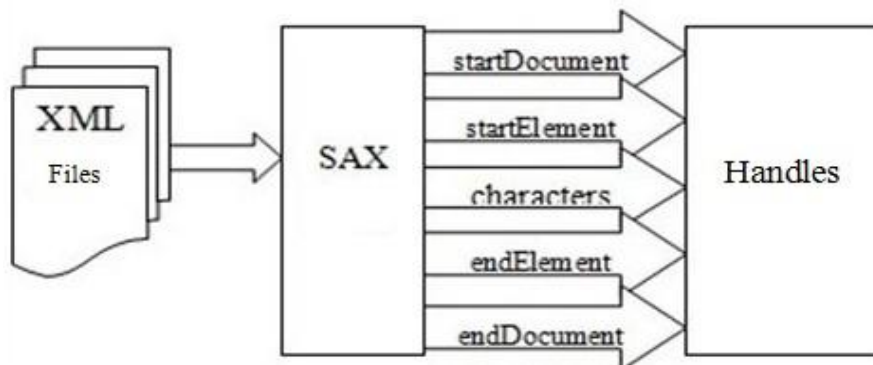


Fig.2 SAX Method

JDOM is an open source Java-based document object model for XML that was designed specifically for the Java platform so that it can take advantage of its language features. JDOM integrates with Document Object Model (DOM) and Simple API for XML (SAX), supports XPath

and XSLT. It uses external parsers to build documents. JDOM was developed by Jason Hunter and Brett McLaughlin starting in March 2000. It has been part of the Java Community Process as JSR 102, though that effort has since been abandoned.

For the systems built by using .Net language, the method to analyze and process XML files is by using System.XML of .NET Framework. As .Net Framework is only compatible with Microsoft's operating system, so this method only can be used in the Microsoft windows operating system.

For the systems built by using PHP language, the method to analyze and process XML files is by using DOM and SimpleXML. Compared with .Net Framework, PHP platform has no special requirements on the operating system.

Based on what we have mentioned above all, one cross-platform interface based on Domino can analyze and process Xml files by using Domino java agent for avoiding the complexity of using lotus script agent. After interface data generated, timing FTP scripts were chosen in Domino-based application systems in java agent to achieve data transmission, and the agent program execution time was set, the time interval is 5 minutes. As a result, this interface can be deployed in both Unix system and Windows system, it provides cross-platform data transmission service.

Implementation of Interactive Interface

Interactive Interface Description

The data which other application systems need were generated as XML files according to the arranging XML format in the Domino-based application systems platforms, then these XML files were uploaded into the specified directory of other application systems by using FTP method.

Other application systems trigger the operation regularly several times a day to read the XML file automatically, obtained data services in accordance with the agreed XML format and provide the data synchronization in those database of other application systems. The interface data flow is shown in Figure 3 as below.

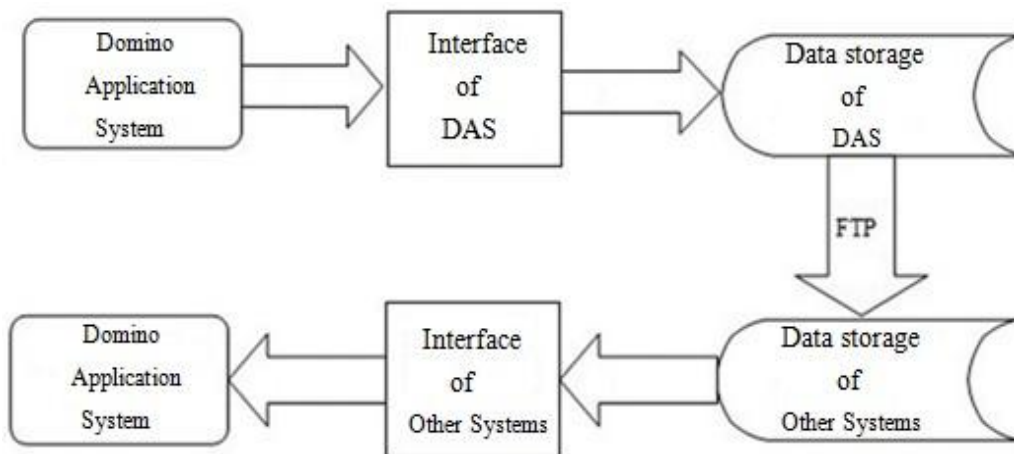


Fig.3 Date Flow of interactive interface

Steps of Interface Development

The following items are the steps of interface developing:

(1)Domino Database:

New database interfaces display database jkzs.nsf, specify the template is empty.

(2) Domino Form:

New form: (interface content | FmContent), it be used to display data for the input interface.

New form: (ftp server configuration | FmFtpSet), it be used to configure the FTP servers of other application systems.

(3) Domino View :

Create view vwContent, the interface was used to display the existing list of data.

Create vwFTP user FTP configuration documentation. New Page: \$ \$ viewtemplate for vwContent. It can be used to embellish the view vwContent.

(4) Resources:

New Image Resources: close.Jpg, editor.Jpg, save and exit.Jpg

(5)New Agent:

New Agent: agFtpDocOpen, to create one new system or modify other servers configuration information.

New Agent: DeleteDoc, to remove the documents, the parameters of this agent were the unid of transmission documents, to delete the corresponding documents.

New Agent: agDocSend, the agent was used to generate XML files and the corresponding attachments, and to upload those data to the servers of other application systems. This agent is the main function of this cross-platforms Interface. This function was achieved by using java agent. Three main classes were created: public class FtpUtil, public class JavaAgent extends AgentBase, public class MyXmlTool.

New agent: WebQueryOpen, it was used to invoked when opening the interface document, the main use is to display the operation toolbar and to provide different operating authority.

Summary

In this paper, One new cross-platform interface was researched and implemented to complete the data transmission between Domino-based application systems and other relational database-based application systems. The architecture of typical domino systems was discussed. For the practical needs, by using XML files as the carrier of structure data is one method to achieve data transmission between all application systems. And XML files were treated as an attachment, identified relations for the unstructured data transmission achievement. This interface, without connecting to the other database directly, uses XML as the carrier by using loosely coupled method to transmit structural and unstructured data. And it is a research and development direction of cross-platform data transmission.

References

- [1] Guisset, Fabian. "What does each DOM Level bring?". Mozilla Developer Center. Mozilla Project. https://developer.mozilla.org/en/docs/DOM_Levels. Retrieved January 10, 2011
- [2] Akl, C.J.; Bayoumi, M.A. Single-Phase SP-Domino: A Limited-Switching Dynamic Circuit Technique for Low-Power Wide Fan-in Logic Gates, Circuits and Systems II: Express Briefs, IEEE Transactions on Volume: 55, 2008 , Page(s): 141 – 145
- [3] Fernandez-Valmayor, A. Building a Syntax Directed Processing Environment for XML Documents by Combining SAX and JavaCC, 2008. DEXA '08. 19th International Workshop on Database and Expert Systems Application 2008 , Page(s): 256 - 260
- [4] Akl, C.J.; Bayoumi, M.A. Single-Phase SP-Domino: A Limited-Switching Dynamic Circuit Technique for Low-Power Wide Fan-in Logic Gates, Circuits and Systems II: Express Briefs, IEEE Transactions on Volume: 55, 2008 , Page(s): 141 – 145
- [5] Koch, Peter-Paul. "The Document Object Model: an Introduction". Digital Web Magazine. http://www.digital-web.com/articles/the_document_object_model/. Retrieved January 10, 2009
- [6] Thavot, R.; Bin Ab Rahman, A.A.-H.; Mosqueron, R.; Mattavelli, M.; Automatic mutli-connectivity interface generation for system designs based on a dataflow description, Ph.D. Research in Microelectronics and Electronics (PRIME), 2010, Page(s): 1 - 4
- [7] Joseph J, Bambara paul. J2EE Technology[M]. Beijing:China Machine Press. 2009:45