Workflow Interoperability Using Extensible Markup Language (XML)

Zulkhairi Md Dahalin[1] and Juliana Wahid[2]

[1] School of Information Technology
Universiti Utara Malaysia
06010 Sintok Kedah, Malaysia

[2] School of Information Technology
Universiti Utara Malaysia
06010 Sintok Kedah, Malaysia

Abstract - States of the art workflow management system (WFMS) do not have adequate features of interoperability among WFMSs that cause incompatible "islands" of process automation within an organization or between organizations. This paper introduces method of interoperability that accommodate the variety of implementation techniques in realizing the WFMS interoperability. To realizing the workflow interoperability, this paper uses two WFMS products in the market, i.e. Domino Workflow and Microsoft Exchange 2000 Server. The method that lies in a workflow interoperability framework is a major contribution of this paper. The framework consists of both WFMS products together with two interoperability tools, i.e. Microsoft Exchange Connector for Lotus Notes and Document Object Model (DOM) of Extensible Markup Language (XML) standard that addressing the communication and exchanging document issues between both WFMS products. To demonstrate the workflow interoperability of our approach, a prototype system has been implemented.

1. INTRODUCTION

Recent years have seen the focus of a number of issues for workflow. These include support for ensuring correctness of execution of workflows [4], ability to work with heterogeneous systems and at the same time provides high availability [3], and integrating simulation modeling and analysis capabilities in workflow [5]. These studies have addressed some of the limitations of workflow management system (WFMS) described by [2] and [1] such as lack of interoperability among workflow management systems, lack of support for correctness and reliability, and weak tool support for analysis, modeling, testing and debugging workflows.

Apart from the efforts by individual or groups of individuals, an organization, known as The Workflow Management Coalition (WfMC), was established to address some of the issues in workflow. As stated in [9], the WfMC is founded in August 1993. It is a non-profit, international organization of workflow vendors, users and analysts. Some of the objectives of this organization are to develop workflow standard terminology and enable interoperability between different workflow systems.

The availability of wide range of products within the market has allowed individual WFMS product vendors to focus on their strength on particular functional capabilities and organizations to use different WFMS products within different organizations or departments [7]. Since WFMS products in the market are not standardised in term of software component, system definition and control data used [7], different WFMS products are not able to work together. Furthermore, different products may run on different platforms. So, business processes have to stop at department or organization borders that result in incompatible "islands" of business process automation. To overcome this problem, workflow interoperability among WFMSs should be achieved.

This paper presents workflow interoperability framework between Domino Workflow and Microsoft Exchange 2000 Server. Both products are chosen because they belong to big vendors; Lotus Development Corporation and Microsoft Corporation, which are competitors to each other in the marketplace [6].

The contribution of this paper includes the following:
1) We demonstrate that different WFMSs in the marketplace could achieve interoperability between each other using a middleware or gateway, which is a more immediate applicability without to implement the functionality described by the specification from WfMC.
2) We provide indirect collaboration between Lotus Corporation and Microsoft Corporation by providing a workflow interoperability framework that can be used as guidelines for organisations that use Domino Workflow as their initial WFMS and probably planning to install Microsoft Exchange 2000 Server as a second WFMS and vice versa.

The rest of the paper is organized as follows. Section 2 reviews generally the strategies, models and levels of workflow interoperability that would be achieved when developing interoperability between WFMS products. Section 3 presents the interoperability tools used to address the communication and document exchange issues between both WFMS products. Section 4 discusses in detail both interoperability tools in practice. Finally,

we conclude with the strategy, level and model of workflow interoperability achieved in this paper.

2. STRATEGIES, MODELS AND LEVELS OF WORKFLOW INTEROPERABILITY

According to [8], interoperability is normally achieved using one of the following strategies:

1. Direct interaction - This strategy applies to products that share a common (standard) Application Programming Interface (API) that allows the products to interact with each other directly in order to move work between them.
2. Message passing - Message passing is a strategy of exchanging information between WFMS products by sending packets of data messages between them using an available communication network.
3. Bridging - The strategy of bridging uses gateway mechanism to move work between WFMS products. It may be part of the products that use it or a separate product. Gateways use protocol converters to map data and command formats from one workflow engine to another.
4. Use of shared database - This strategy of interoperability moves work between WFMS products via a shared database. This approach can be viewed as being just another form of store-forward mechanism.

[8] has identified three possible interoperability models that can be achieved when developing interoperability between WFMS products as follows:

1. Chained Model
   According to [7], this model supports the transfer of a single item of work (a process instance or activity) between two workflow environments, which then operates independently in the second environment with no further synchronization.
2. Nested Sub-process Model
   According to [7], this model allows a process executed in a particular workflow engine to be completely encapsulated as a single task within a (superior) process executed in another workflow engine.
3. Parallel Synchronised Model
   This model of interoperability allows two processes to operate essentially independently, across separate workflow engine, but requires the synchronisation points exist between the two processes [7].

[8] also has identified that these three workflow interoperability models can be achieved through eight levels of interoperability:

Level 1 – No interoperability
   No way of communicating between WFMS products.

Level 2 – Coexistence
   WFMS products can be implemented as different parts of a whole process with active participation of human agents to enable the interaction between WFMS products.

Level 3 – Unique Gateways
   WFMS products can work together using some bridging mechanism, i.e. encapsulation, translation or gateway, which performs routing of operations between workflow engines and instances, translation and delivery of workflow relevant data and translation and delivery of workflow application data.

Level 3a – Common Gateway API
   WFMS products can work together using gateways that share a common (standard) API.

Level 4 – Limited Common API Subset
   WFMS products can work together by sharing a common (standard) API. A common (standard) API is the one that are defined in neutral information formats to handle the transport of workflow relevant and workflow application data that most workflow engines can implement it.

Level 5 – Complete Workflow API
   WFMS products can work together by sharing a single standard API that gives access to the full range of possible operations by any workflow management system. The possible operations can be defined and then mapped to operations for each workflow product.

Level 6 – Shared Definition Formats
   WFMS products can work together by having a shared format for process definitions that covers routing decisions; user access rights and the maintenance of workflow system resources.

Level 7 – Protocol Compatibility
   This level assumes that all API client/server communication including the transmission of definitions, workflow transactions and recovery is standardized.

Level 8 – Common Look and Feel Utilities
   This level assumes that in addition to the preceding levels, all workflow products present the user with the same standard user interface and method of operation.

3. COMMUNICATION AND DOCUMENT EXCHANGING MANAGEMENT

In this section we introduce the interoperability tools that addressing the communication and document exchanging issues between Domino Workflow and Microsoft Exchange 2000 Server.

In realizing the workflow interoperability between both products, firstly depends on the method of communication between both products. The failure to realize the method of communication or the strategy would contribute to the failure of both products to

514

interoperate to each other because this will achieve the first level of workflow interoperability, i.e. no interoperability. From analysis of several existing tools in the marketplace, we discovered Microsoft Exchange Connector for Lotus Notes. This tool enables the exchanging of messages between Domino Workflow and Microsoft Exchange 2000 Server.

From the analysis, we also discovered that the document exchanging issues between both products could be addressed by Document Object Model (DOM) of Extensible Markup Language (XML) standard.

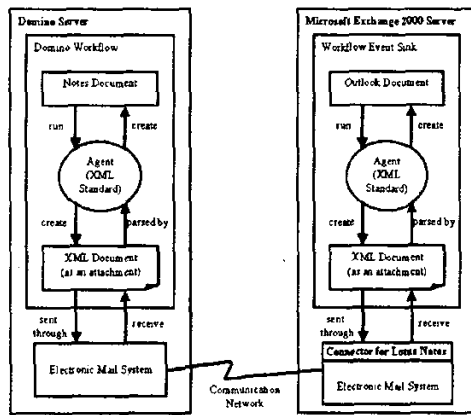These two tools were modeled together with Domino Workflow and Microsoft Exchange 2000 Server as shown in Fig.1

Fig.1. Workflow Interoperability Framework

Based upon the analysis on the interoperability tools, we found that the Microsoft Exchange Connector for Lotus Notes can synchronize user directory between Domino Server and Microsoft Exchange 2000 Server. In addition, it can deliver messages between the two products. The messages delivered could be contained an XML document as an attachment. This framework shows that document in Domino Workflow environment will be generated to an XML document by an agent (the XML standard resides in this agent). The XML document then will be sent to Microsoft Exchange 2000 Server environment as an attachment. The Connector that resided in Microsoft Exchange 2000 Server environment handles the message delivery between Domino Workflow and Microsoft Exchange 2000 Server. The attachment in the message will be extracted and parsed into a document in Microsoft Exchange 2000 Server by an agent where an XML standard resides in this agent. This framework also shows the same flow from Microsoft Exchange 2000 Server environment to Domino Workflow environment.

4. WORKFLOW INTEROPERABILITY DEVELOPMENT

4.1 Microsoft Exchange Connector for Lotus Notes

In order to enables the exchanging of messages between Domino Workflow and Microsoft Exchange 2000 Server, the connector firstly was configured.

Firstly, the Domino Server name has to be specified to enables the connector to connect to the Domino Server. Secondly, the address rule in Microsoft Exchange 2000 Server has to be modified to represent Notes electronic mail in Exchange. Thirdly, specifies the import container that store objects imported from the Domino Server to the Microsoft Exchange 2000 Server such as users, groups and resources. Fourthly, specifies the export container that contains Microsoft Exchange 2000 Server Recipient addresses that are exported to the Domino Directory. Fifthly, configures the message routing in Microsoft Exchange 2000 Server so that it has the correct address space so that messages can be routed to the correct destination in Domino Server. Sixthly, specifies the domino directory database that is used as the source and the target in the directory synchronization. Seventhly, starts the connector service in the Microsoft Exchange 2000 Server. Finally, synchronizes the directory information to export the directory information for Microsoft Exchange 2000 Server users to the directory of Domino Server. In turn, directory information for Domino Server users is imported to directory information for Microsoft Exchange 2000 Server.

When directory synchronization is finished, each system has a complete copy of the directory for entire organization, ensuring that users on Domino Server can send mail to users on Microsoft Exchange 2000 Server and vice versa. The use of the connector has implies the message passing strategy to communicate the Domino Workflow and Microsoft Exchange 2000 Server.

4.2 Document Object Model (DOM) of Extensible Markup Language (XML)

This tool is basically employed in agents in both products application environment. In order to depict the use of this tool, the leave application system was built as prototype.

Both in Domino Workflow and Microsoft Exchange 2000 Server, the DOM of XML standard was applied in two places. The first one is in the process of generating an XML document at the back-end process of a leave application form. The XML document then is sent as an attachment to either the folder in Microsoft Exchange 2000 Server or application database in Domino Workflow. This activity is also considered as the first interface point of interoperability between Domino and Microsoft Exchange 2000 Server. The second place is in the process of parsing the XML document, which is created in scheduled agent. The agent will be triggered when new electronic mail arrives in either the application database in Domino Workflow or the folder in Microsoft Exchange 2000 Server. This is the second interface point between activities in Domino Workflow to interoperate with the activities in Microsoft Exchange 2000 Server.

In Domino Workflow, both the process of generating and parsing an XML document were carried out using an agent written in Java Language. The following scripts fragment shows the process of generating an XML document:

515

```
BufferedWriter bw=new BufferedWriter (new
    FileWriter("c:\\Cuti.xml"));
bw.write("<?xml version='1.0' ?>");
```

The following fragment script below will parse the content of an attachment and creates the DOM tree of the XML:

```
xmlDoc = attachment.parseXML(false)
```

In Microsoft Exchange 2000 Server, Microsoft Visual Basic Scripting Edition (VBScript) was used for both the process of generating and parsing an XML document. The following scripts fragment shows the process generating an XML document in Microsoft Exchange 2000 Server:

```
Set fso =
CreateObject("Scripting.FileSystemObject")
Set f= fso.CreateTextFile("c:\Cuti.xml", True)
f.write "<?xml version='1.0' ?>"
```

The following fragment script below will extract an XML document from an attachment and parse the XML document:

```
Set objxmlDoc=
CreateObject("Microsoft.XMLDOM")
objxmlDoc.async="false"
objxmlDoc.load("C:\Cuti.xml")
```

The use of DOM of XML standard has enables documents in both products environment to be defined in neutral formats, i.e. XML format, that can be handled by each product. So, the use of DOM of XML standard has achieved the fourth level of interoperability, i.e. the Limited Common API Subset level.

By developing the leave application system as the prototype, two models of interoperability could be achieved, i.e. Nested-Sub process and Parallel Synchronised model. The Nested-Sub process model will be achieved if users from Domino Workflow will be the only applicants of the leave application and the users in Microsoft Exchange 2000 Server will be the only approver. In this case all the process of leave application have to wait for the termination of the execution approval process in Microsoft Exchange 2000 Server before carrying on with the decreasing leave balance activity in Domino Workflow.

The Parallel Synchronised model will be achieved by assuming all users could be an applicant and an approver in both environments. Applicants in both environments can execute the leave application process simultaneously where interoperability can be achieved when the approval process is reached. After the termination of execution in the approval activity, both leave application process continues with the decreasing leave balance activity in their respective process.

5. CONCLUSIONS

In this paper, we have demonstrated the workflow interoperability between Domino Workflow and Microsoft Exchange 2000 Server in a running leave application prototype. We concluded that the use of

Microsoft Exchange Connector for Lotus Notes has implied the message passing strategy and the use of DOM of XML has achieved the Limited Common API Subset level. The leave application system that has been developed shown that two models of interoperability are achieved, i.e. Nested Sub-process and Parallel Synchronise model.

6. REFERENCES

[1] Alonso, G. and Schek, H. (1996). Research Issues in Large Workflow Management Systems. *Proceedings: NSF Workshop on Workflow and Process Automation in Information Systems: State-of-the-art and Future Directions.*

[2] Georgakopoulos, D., Hornick, M., and Sheth, A. (1995). An Overview of Workflow Management: From Modelling to Workflow Automation Infrastrucutre. *Distributed and Parallel Databases, 3,* 119-153.

[3] Kamath, M., Alonso, G., Gunthor, R. and Mohan, C. (1996). Providing High Availability in Very Large Workflow Management Systems. *The Fifth International Conference on Extending Database Technology (EDBT 96)* Avignon, France.

[4] Kamath, M. and Ramamritham, K. (1996). Correctness Issues in Workflow Management. *Distributed Systems Engineering (DSE) Journal : Special Issue on Workflow Management Systems, Volume 3,* Number 4.

[5] Miller, J., Sheth, A., Kochut, K.,Wang, X., and Murugan, A. (1995). Simulation Modeling Within Workflow Technology. *Proceedings of the 1995 Winter Simulation Conference,* Arlington, Virginia, pp. 612-619.

[6] Osterman, M. (2001). A Comparison of Exchange and Domino Application Development. *Creative Network, Inc.* Retrieved from the World Wide Web: http://www.cnilive.com

[7] Workflow Management Coalition. (1995). The Workflow Reference Model. *WfMC Standards. Document Number TC00-1003.* UK: Hollingsworth. Retrieved July 2, 2000 from the World Wide Web: http://www.wfmc.org/standards/docs.htm

[8] Workflow Management Coalition. (1996). Workflow Management Coalition Standard – Interoperability Abstract Specification. *WfMC Standards. Document Number WFMC-TC-1012. Version 1.0.* Belgium: Workflow Management Coalition Members. Retrieved July 27, 2000 from the World Wide Web: http://www.wfmc.org/standards/docs.htm

[9] Workflow Management Coalition. (1998). Coalition Overview. *A WfMC White Paper.* Retrieved July 10, 2000 from the World Wide Web: http://www.tims.ch/vmfc.html

516