

©2005 IEEE. Personal use of this material is permitted. However, permission to reprint/republish this material for advertising or promotional purposes or for creating new collective works for resale or redistribution to servers or lists, or to reuse any copyrighted component of this work in other works must be obtained from the IEEE.

Services Integration Monitor For Collaborative Workflow Management

Leo Pudhota, Andrew Tierney, Elizabeth Chang

PudhotaL@cbs.curtin.edu.au, TierneyA@cbs.curtin.edu.au, ChangE@cbs.edu.au
School of Information Systems, Curtin University of Technology PO Box U1987, Perth
WA 6845, Australia

Abstract

Collaborative workflow management systems in logistic companies require strong information systems and computer network support. These IT integration requirements have expanded considerably with the advent of e-business; utilizing web services and Partner to Partner e-commerce. This paper deals with change management of collaborative workflow in such consortia and proposes architecture for synchronization and monitoring of these services where existing workflow systems are adapted to the changes requested by management. This paper describes conceptual framework of the services monitor which is used to monitor and edit changes resulting in new collaborative workflows.

1. Introduction

In this paper we discuss prototype implementation of adaptation management for dynamic business processes of large logistic consortia. Unpredictable situations may occur as a result of changes in decisions made by the management. Workflow systems have been unable to deal with flexibility, and this inability to deal with various changes greatly limits the applicability of workflow systems in real industrial and commercial operations. This situation raises problems in workflow design and workflow systems development. We propose a workflow adaptation methodology through the process of integration and monitoring techniques for the development of new workflows using existing workflow components.

2. Collaborative Workflow in Logistics Consortium

A consequence of this connectivity and information richness is that one is faced with an increasingly dynamic business environment and marketplace [1]. This environment requires support in the form of collaborative workflow. A

workflow is a sequence of activities that produces a result of observable value. A collaborative workflow focuses on working together towards common goals. They can be used by small groups of companies, project-oriented research teams, or by widely dispersed industries with common interests. Considering a logistics consortium, with many departments, let's take into consideration six operational departments: Management, Warehouse, Logistics, Accounts, Customer Service and Transport. Each department has its own responsibility however they are connected to each other. The Warehouse Departments typically already have their own system, as do the Accounts Departments. The complexity of works become bigger as the customer's orders increase so it is hard to know the progress of the orders, check the warehouse and it is also difficult to schedule the trucks, manpower, etc. A consortium likes to change its internal work (flow of works amongst departments) and its external work (flow of works with its customers and other collaborative organizations). A consortium would like to integrate various departments and also with other logistic network companies. A consortium also wants its customers to be able to book warehouse services, logistic services, place orders and view the status of orders, etc on the Internet or through web services. This is more like the e-commerce way, on the other hand Logistics Management, in its widest definition, is concerned with the strategy and management of the movement and storage of materials and products from suppliers, through the firm's distribution systems to retail outlets and customers. The scope of logistics management for the physical movement of goods starts with the sources of supply and ends at the points of consumption [2, 4]. Logistics management attempts to achieve a balance between holding minimum stock while providing the best services possible to the customer.

3. Challenges of Collaborative workflow

Changes in collaborative workflow have to be incorporated into the integrated enterprise system;

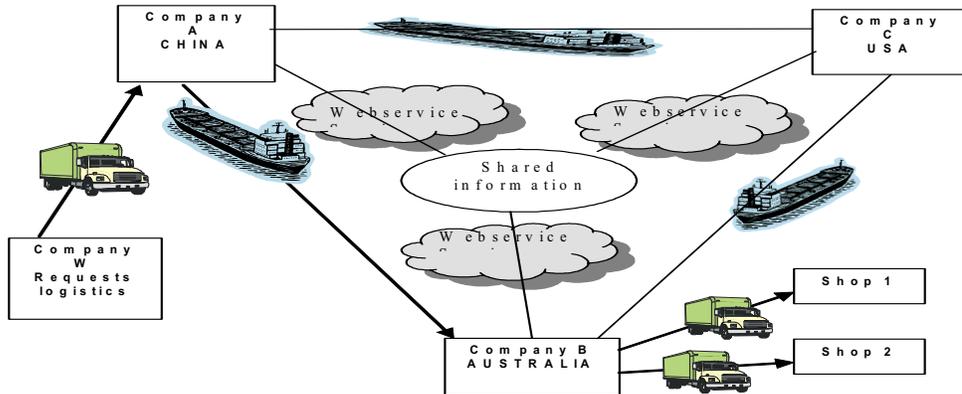


Figure1. Global collaborative logistic companies' network

we have proposed a prototype of its working in our previous papers [8, 9, 10, 11]. In this paper we are concentrating on:

1. Implementation aspects of integration and adaptation of changes of the new workflow into an already existing workflow.

2. Information systems can change at run time hence a new workflow should be able to be monitored for synchronization with an existing workflow in order to adapt quickly.

3.1 Management of data scattered over multiple original systems/legacy systems. For example, a company would prefer to consolidate data into one logical view with a unified architecture, thereby enabling data-source independence. Because application data continues to live and change in the origin systems, the new software layer must be able to retrieve original data on the fly and also propagate changes back to the original systems.

3.2 Provide support for transactions/interaction across multiple back-end systems. "The hard part is getting a transaction model wrapped around those back-end systems; so if it didn't work in the third system, it was able to roll back in those first two systems" [17].

These challenges will help in creating a uniform data processing environment for the whole enterprise, which would lead to changes and improvements in customer services, control of receivables and increased efficiency in communication, sales, marketing as well as minimization of warehouse stocks, streamlining inventory and logistics flows.

It will also provide control to consortium management to monitor the collaborative enterprise's condition, its stock, order and its general financial condition on a routine basis; this is indispensable to the management processes and enhances decision-making for the changes which

need to be taken in the short and long term bases for the consortium to compete in the global market.

4. Workflow Mining Technique for Adaptation

The decisions taken by the consortium board which includes representatives from individual companies is represented as unstructured data which is passed on to the collaborative consortium management. This data which is unstructured is converted into structured data through a process of unstructured data management (UDM), a process of mining, organizing and analysis to extract actionable information. Here, UDM is used as a text mining tool to convert the unstructured information into structured data by considering precision, thoroughness and relevance of the unstructured information provided (please refer to figure 2). Structured information is used to create a new collaborative workflow strategy which helps in the creation of a new dynamic workflow. We propose using a meta-data based application generator to generate the new workflow [9][10]. The dynamic workflow and the existing workflow of an organization are merged to create an adaptive workflow through a synchronization module which helps in the merging of the new workflow to existing workflows without having to disturb existing workflow processes. Figure 2 show three distinct tiers of the adaptation architecture; the technical and communication layer which forms the data store for an enterprise, the middleware consisting of synchronisation and monitor and on top of these layers lies the framework and networking layer which provides services to individual companies within the consortium through web services or an application programming interface This flexibility is useful in

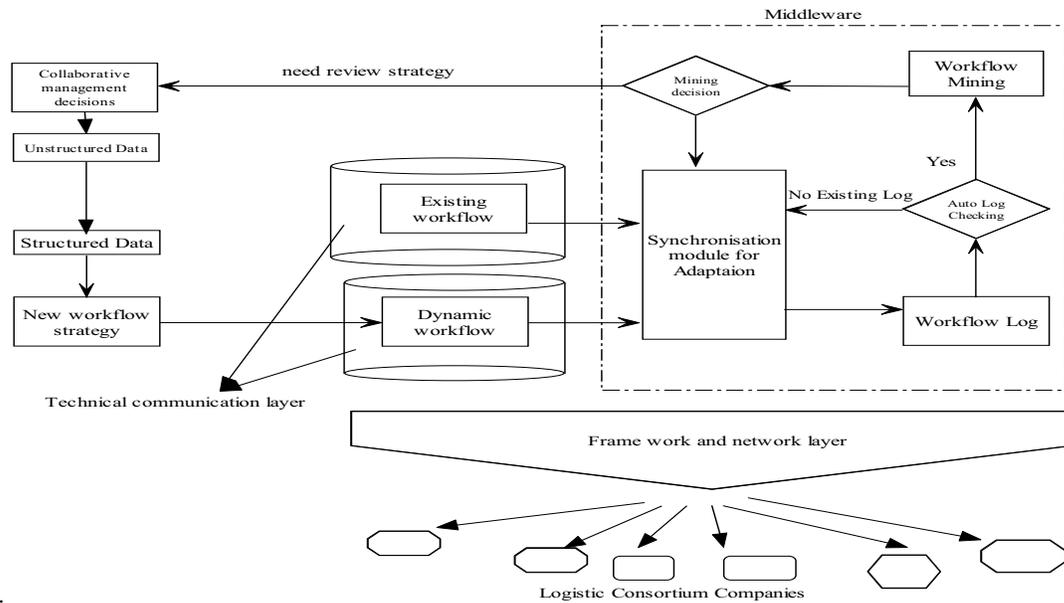


Figure 2. Workflow Mining Strategy for Logistics Consortium and three tier adaptation architecture

the collaborative environment as the business models may also undergo change from time to time as some companies no longer want to be part of the consortium and new companies may want to join the consortium.

5. Conceptual Model of a Monitoring System

Creating workflow processes as per the requirements of management is complicated and time consuming. To support the continuous design process, we propose using a workflow monitoring system which is used to create, edit and delete the workflow processes (refer to figure 4) information collected at runtime and can be used in the diagnosis and redesign phases to derive a model explaining the events recorded in workflow mining. Internal factors are passed on for further consideration in the decision making processes by management; otherwise it is a cyclic process in creating a new workflow model from existing workflow. This service monitor and repository acts as a generic domain model, to build a common understanding and vocabulary amongst logistics domain experts and helps in creating an architectural separation of business functionality from workflow implementation. This separation allows designers to use business rules defined in a UML model to implement such systems. This system is used to monitor the data flow and workflow in the collaborative organization system. The software is broadly divided into two main applications; client side application and server side application.

5.1 Service Oriented Framework For Client Application

We have proposed a conceptual model of the service oriented framework [18]. The web services framework has four layers; 1. Framework Layer, 2. Network Layer, 3. Communication Layer and 4. Technical Layer. In the Framework Layer browsers interact with HTTP servers in their normal way taking advantage of any technologies that enhance this browser-to-web server link. For example in the Network Layer, secure socket layer communication protocols in Netscape and Microsoft browser / server products communicate with HTTP servers, which communicate with the Application Server. The Framework Layer generates web applications at run time, the It is the job of the HTTP server adaptor to communicate with a given HTTP server and forward requests to one or more application "instances" - an instance is a separate copy of a given application process. Enterprise services framework serving a few users may have only one instance.

A large application may have tens or hundreds of instances running on one or more machines. If an application has more than one instance, the Enterprise services framework controller is essentially acting as load balancing agent. If an instance fails, it only affects that particular instance – all other instances and/or the site's web server are unaffected. The controller will forward requests over the network as easily as it will forward requests to applications running on the same box as the HTTP server. In fact, from a load

sharing perspective, it is ideal for the HTTP server and Application servers to reside on separate boxes.

Since client applications are server based, database access happens behind the firewall. Browsers need never make direct connections to a database server. Services access database connections so that they are highly secure (only accessible via actual application API), and conserved (that is, you never have more than one connection per instance regardless of the number of users supported - unless this is specifically something the developers desire). Java Foundation contains fundamental data structures, implementations and utilities used.

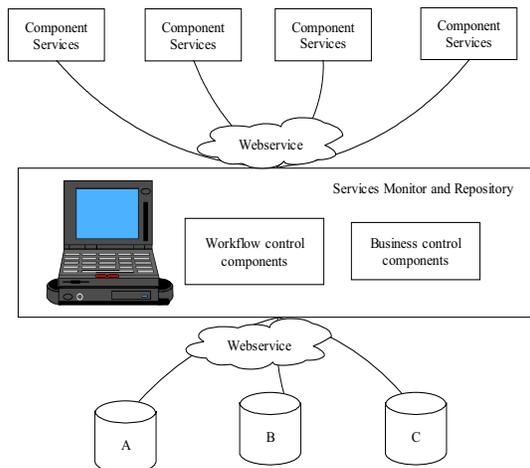


Figure 3. Enterprise monitor repository

The client user will receive commands, guidance and orders from company headquarters through the system and then complete the task. Information is sent back to headquarters directly - therefore headquarters can monitor all workflows and events that happen in each part of the individual companies in real time.

5.2 Service Oriented Framework for Server Side Application

On the server side we propose linking all the databases to corporate database which are accessed by XML-Object Relational Mapper (XML-ORM) that decouples the traditional ORM from both the development platform and proprietary transport layer. The XML-ORM also provides a detached data definition, data manipulation and schema query interface that is database independent and provides the developer and user with a consistent interface to the data [20].

5.2.1 Features Offered By ORM

A Persistence Layer connecting objects in an object-oriented system to data stored in a relational database.

Most commonly, an SQL database is used (though strictly speaking, SQL does not implement the Relational Model -- see SqlFlaws and Relational Language for more details). Using an Object Relational Mapping (ORM) allows one to cleanly apply object-oriented design, analysis, and programming techniques while hiding the specifics of dealing with the relational system. There are commercial and Open Source ORMs that comply to the ODMG and JDO standard APIs for object persistence. Programming against such a mapping is not different from programming against an Object Oriented Database. Using such a ORM may be a good compromise between OO designers/architects insisting on pure object orientation without any data knowledge, and DBAs/managers emphasizing the shortcomings of OODBMSs [3].

6.0 Construction of the Xml-Object Relational Mapper

Although based on much of the functionality of the Object Relational Mapper (ORM), the XML-ORM differs from existing ORM solutions in the way that it exposes the internal ORM objects to the user. Current ORM solutions require the creation and mapping of physical classes to the relational schema--necessarily resulting in code level platform dependencies. For the developer to expose and thus make available these objects to other platforms, additional layers or facades are required to be created and maintained. Deployment and platform choice issues also increase in complication due to the vendor driver or library installation requirements, plus the ever present issues of distributed solutions that need to shield the user from the protocol and traverse corporate firewalls. The XML-ORM exposes its internally created objects to the user via standard Web Services [23] and XML [24] technologies which totally eliminate platform dependencies issues from the development and maintenance cycles [21], [22] [20] (although the XML-ORM itself must still interface to the DBMS systems using the available vendor interfaces)[20].

By utilizing such a standard interface, access to the XML-ORM is greatly simplified and the technology used is already available for most platforms that support Microsoft's .NET or Sun's Java. By accessing in this manner, full transactional database access becomes available to the monitoring system - using XML-ORM we can also these devices can directly utilize the same fully fledged remote database access to client application data as well. Figure 5 illustrates the basic XML-ORM interface architecture between monitor and the DBMS systems [23], [24]. Any application that interoperates with the XML-ORM requires a supporting local API or function set that

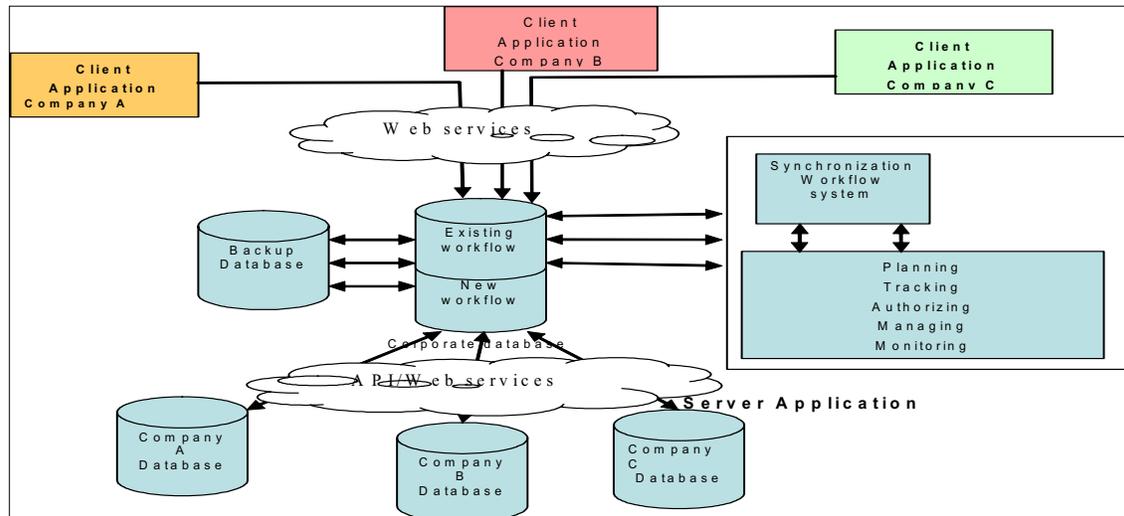


Figure 4: collaborative workflow monitor system architecture

aids in the creation of the XML based XML-ORM commands (optional but desired) and in issuing the command via local Web Services (mandatory but typically already available). Local acceleration of XML-ORM XML commands will require database access application and development environment vendors to provide support for mapping existing database access functionality to their own XML-ORM wrapper classes, using existing functionality. Figures 5 provide an example of a client application interacting with a XML-ORM system, in this case requesting a schema definition via an XSEL request.

There are two methods for creating the XML-ORM generation of tools – designing and creating a new XML-ORM from scratch is the first alternative although potentially costly - perhaps the most cost effective way to initially develop XML-ORM technology is to modify existing (open source) ORM tools to act as XML-ORM systems. To modify a suitable ORM system to become a XML-ORM requires the following steps; 1. Establish a Web Service interface to the ORM system to process the XML-ORM command structure, 2. Develop XML-ORM command mappings to existing ORM methods, plus integrate new functionality as required (and as follows), 3. Develop local compilation options to dynamically generate the standard internal classes as required for the base ORM system based on the XDDL XML-ORM commands (that would otherwise have been created statically during an interactive ORM design session) and 4. Establish appropriate database schema creation methods in support of XDDL commands for ORM systems that do not currently have schema creation functionality.

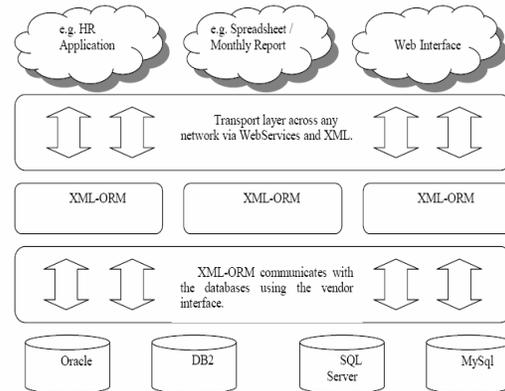


Figure 5. XML-ORM interface structure from application request to DBMS systems

7.0 Evaluation Of Monitor and XML-ORM Mapper

XML-ORM as a backend integrated application server costs time, money and schedule time on a software development project, due to the (substantial) additional effort required to:

1. Procure, install, and administer additional third-party products such as relational database management systems and Object Relational Mapping tools such as Top Link .
2. develop and maintain a relational schema corresponding to the Domain Model;
3. Learn and use an Object Relational Mapping tool, and develop and maintain the mapping between the relational schema and the Domain Model.

XML-ORM costs runtime delayed response time due to the additional inter-process communication, network I/O, object instantiation, and processing that is required

to execute SQL statements that move Domain Object state between the RDBMS and the address space of the process running the software written in the Object Oriented Programming Language. This runtime response time delay has been observed to be anywhere from twice to an order of magnitude more, for certain applications and use cases, than the runtime response time observed for the same applications and use cases running.

Object Relational Mapping costs money. Capital expenditures are required to procure additional third-party products if the project is already using an application server that has Object Oriented Database Management System capabilities. Recurring labor expenditures are required to staff the additional development efforts of administering an RDBMS and developing and maintaining schema and mappings [3]. The potential benefits that offset these drawbacks are:

- a) The possibility of maintaining Database Application Independence within the consortium;
- b) SQL-based access to the data which helps integrate legacy systems ; and
- c) Greater confidence in vendor viability from monitoring the business point of view but also for the consortium to maintain integrity and security of the consortium.

6. Conclusion:

In this paper, we presented a collaborative logistics workflow management and monitoring system based on changes made at managerial levels. We have also discussed benefits, challenges, evaluation and framework of synchronization mechanism for these enterprise systems and have come up with various approaches for the dynamic adaptation of the management workflow system. In our future research we propose to develop a prototype using the discussed methods as outlined above.

7. Reference:

[1] Marshak, R.T.: "Falling in Love with Distinctions", In "New Tools for New Times: The Workflow Paradigm", Future Strategies Inc., 1994
 [2]Miers, D: "The Workware Evaluation Framework", Enix Limited, 1996
 [3] Retrieved on 28 April 05 <http://c2.com/cgi/wiki?ObjectRelationalMapping>
 [4]Haake, J.M., Wang, W.: "Flexible Support for Business Processes: Extending Cooperative Hypermedia with Process Support".
 [5]Denning, P.J.: "The fifteen level", In Proceedings of ACM SIGMETRIC Conference on Measurement & Modeling of Computer Systems, May 1994.
 [6]Sheth A.: "State-of-the-art and future directions", In Proceedings of the NSF Workshop on Workflow and Process Automation in Information Systems, May 1996.

[7] David Neumann, An Introduction to WebObjects. Retrieved: July 30, 2004, from <http://mactech.com/articles/mactech/Vol.13/13.05/WebObjectsOverview>.
 [8] Pudhota L, Chang E. et al. International Journal, Computer Science, System and Engineering, "Extension of Activity Diagrams for Flexible Business Workflow Modeling "volume 18 no3 May 2003, UK.
 [9] Pudhota L, Chang E "collaborative workflow management for logistics consortium" ICEIS April 2004 Porto, Portugal.
 [10] Pudhota L, Chang E "Modelling the Dynamic Relationships between Workflow Components" ICEIS April 2004 Porto, Portugal.
 [11] Pudhota L, Chang E, Venable J "E- Business technology adaptation through workflow mining" MSV June 2004 Las Vegas, Nevada, USA.
 [12] http://www.omg.org/mda/mda_files/UNextMDA4.pdf
 [13] Ulieru. M, Robert W. Brennan Scott S. "The holonic enterprise: a model for Internet-enabled global manufacturing supply chain and workflow management", Canada 2000.
 [14] Ulieru. M, Stefanoiu. D, et al. "Holonic metamorphic architecture for manufacturing" University of Calgary, Calgary, Canada 2000.
 [15] Brandenburger, A. M. and Nalebuff, B. J., (1996), Co-operation, Doubleday NY.Brennan, R. (2000), "Performance comparison and analysis of reactive and planning-control architectures for manufacturing", Robotics and Computer Manufacturing 16(2-3), pp. 191-200.
 [16] Christensen, J.H. (1994), "Holonic Manufacturing Systems: Initial Architecture Standards Directions", Proceedings of the First European conference Manufacturing systems, European HMS Consortium, Hanover, Germany.
 [17] Retrieved on 26 sept 04, http://www.journee.com/n_hl_020703b.html
 [18] Pudhota L, Chang E "Collaborative workflow adaptation through synchronization and clustering" IBIMA Dec 2004, Cozumel, Mexico.[19] Retrieved on 9 Dec 04, [David Wallace Croft](http://www.alumnus.caltech.edu/~croft/research/agent/definition/) Intelligent Software Agents: Definitions and Applications, URL : <http://alumnus.caltech.edu/~croft/research/agent/definition/>
 [20] Tierney A, et al Dec 04 "Requirements for generic transaction based data access services" Australia.
 [21] W3C-Que, 2004. XML Query (XQuery), in <http://www.w3c.org/XML/Query>. 2004, World Wide Web Consortium.
 [22] W3C-Sch, 2004. XML Schema, in <http://www.w3c.org/XML/Schema>. 2004, World Wide Web Consortium.
 [23] W3C-WSA, 2004. Web Services Activity, in <http://www.w3.org/2002/ws/>. 2004, World Wide Web consortium.
 [24] W3C-XML, 2004. Extensible Markup Language (XML), in <http://www.w3.org/XML/>. 2004, World Wide Web Consortium.