BUSINESS PROCESS SYNCHRONIZATION IN SCM BY USING SOFTWARE AGENTS AND CROSS ORGANIZATIONAL WORKFLOW

[1]Tarini Prasad Panigrahy, [2]Bijaya Kumar Panda, [3]Prasant Kumar Bal
[1]Associate Professor, [2]Assistant Professor, [3]Assistant Professor
[1] Department of Computer Science and Engineering, Gandhi Institute for Technological Advancement (GITA), Bhubaneswar, INDIA

Abstract-In Supply chain Management (SCM) in order to support global competitiveness and rapid market responsiveness, it is required to efficiently integrate different organization's business workflow to provide customized services. Integrating the various business processes in BPM is mainly based on an 'information sharing' and redeployment between the various activities that comprise it. Service oriented Cross-organizational Workflow (SCW) supports the collaboration between several autonomous and heterogeneous business processes, distributed over different organizations. Given the dynamic nature of a supply chain, the adaptation of software agents in SCW represents a possible solution for the composition and enactment of cross-organizational services. This paper addresses the internal coordination and control aspects of such architecture by combining agent technology with industry-standard software engineering processes.

Keywords: Software Agent, Supply Chain Management, Cross-Organizational workflow

I. INTRODUCTION

In a typical a supply chain Management, raw materials are procured from a supplier and items are produced, shipped to warehouses for intermediate storage, and then supplied to retailers or customers. The supply chain management must take into account the interactions at the various levels between the stakeholders so as to reduce cost and improve service levels. In a SCM there are mainly three stakeholders to play their roles. The first one, the Manufacturer or supplier, who produces/provides the products. The second one, the Customer or consumer, who purchases the products. The third one the Retailer or distributor, who acts as an broker between manufacturer and customer to sell and deliver the right product to the right customer and purchase products from different manufacturers.

The Customer places purchase order (PO) to the distributor. The distributor, then, combines the entire PO and other information's from the customers and then calculates a demand plan and sent it to the manufacturer. The manufacturer then design production plan accordingly. It forms a close loop with supply_side and demand_side. As SCM are complicated systems, which involves huge volumes of data and business process which could be very hard to accommodate or resolve. So this problem can be addressed with the help of workflow management which provides a solution to facilitate SCM implementation to meet the needs to certain extent [3]. Also one stakeholder to incorporate the services of another within its own processes we introduced an architecture called as Service Oriented Cross organizational workflow (SCW) [1].

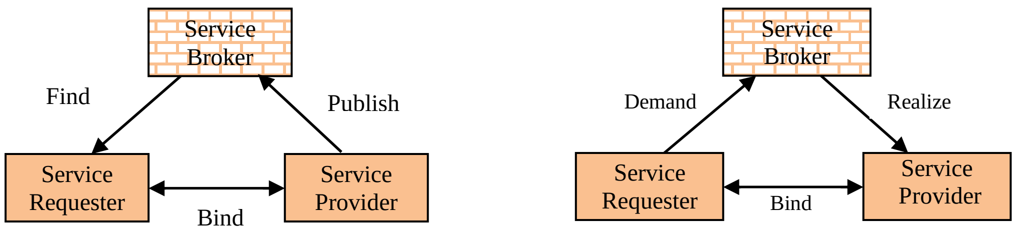
Here in order to address the dynamic aspects of a SCM such as faults or breakdown in utility equipment or an extended delay in taking delivery of raw materials, production facilities fail, customers change or cancel orders, etc. make real-time cooperative operations on the supply chain complex and difficult[2].

The dynamic and distributed nature of the Internet environment is the major problems in this domain. The two main issues to be addressed here are to create systems that operate with respect to these dynamic conditions and to make services those are distributed across different physical and geographical boundaries.

Several related projects that define a solution to address the dynamic problems associated with the Cross-organizational workflow. M.Fox [6] proposes an agent-based supply chain architecture that uses independent reusable components and services which supports complex and cooperative works across organizational boundaries. The CHAIMS [4] project at Stanford University declares a compilation process and text-based composition language. Mennie and Pagurek [5] illustrate a Jini-based architecture for service composition. In their work, they describe an XML-based specification and process to compose services realized as Java Bean components. However, due to the lack of flexible mechanisms to deal with cross-organization business activities, the technology has been continuously under observation. In our work paper we use Software agent technology to problems related to dynamism and distribution. An agent can be defined as an autonomous and goal-oriented software entity, which can operate asynchronously and coordinately with other agents as needed [6,7]. Here we incorporate software agents and workflow automation that provides a flexible, configurable and coordinated approach to achieve efficient and cost-effective management across the entire SCM.The paper proceeds in Section 2 with focus on motivation of the Service oriented cross-organizational workflow with respect to the integration of web services. In Section 3, we focus on SCW environment and the architecture of agent based architecture. In section 4, the Workflow Automation through Agent-based Reflective Processes (WARP) is introduced to support the SCW domain. In Section 5, different modeling view's of WRAP approach and in Section 6, finally we concluded by focusing on the required future work.

II. CROSS ORGANIZATIONAL WORKFLOW AND WEB SERVICES

The SCW environment incorporates the interoperability of web services. For example in a SCM the stakeholders supplier, customer and manufacturer can be considered as components, which can be specified with web services technologies, have the capability of being discovered and accessed from distributed locations. We specify these web services through Web Services Description Language.



(a) Supply_side Model (b) Customer_side Model
Figure-1 Supply_side Model and Customer_side Model

(WSDL) [8], can invoke them using the Simple Object Access Protocol (SOAP) [9].Since we use Software agent technology, we use Web Ontology Language,(OWL-S)[10] as WSDL to a certain extent, do not support the specification of complex interaction(it is used to specify syntactical form of web services), in the use of the web services. Also OWL-S allows automated support for components such as agents to locate, select, employ, compose, and monitor web-based services. In addition, to advertise the specifications of distributed services universally we use Universal Description, Discovery, and Integration (UDDI) [11][12] architectures. When we map the SCM in physical world, with the web service architecture, that has three similar roles which constitute a triangle model, as shown in Figure 1. In this model, the three participants are service provider, service requester, and service broker, corresponding to manufacturer, supplier/distributer and customer (as in Figure) Here the Service provider publishes services to service broker, and the service requester can search the required services. Once a proper service is found, service requester could bind the service with provider directly. This model is from supplier/manufacturer's side we can call such a model as supply_side_model.

III. SCW ENVIRONMENT AND ARCHITECTURE OF AGENT-BASED WORKFLOW

The SCW environment described in this paper incorporates the interoperability of general web services offered by different business organizations, as illustrated in the fig. 2 by taking the scenario of a typical SCM process.

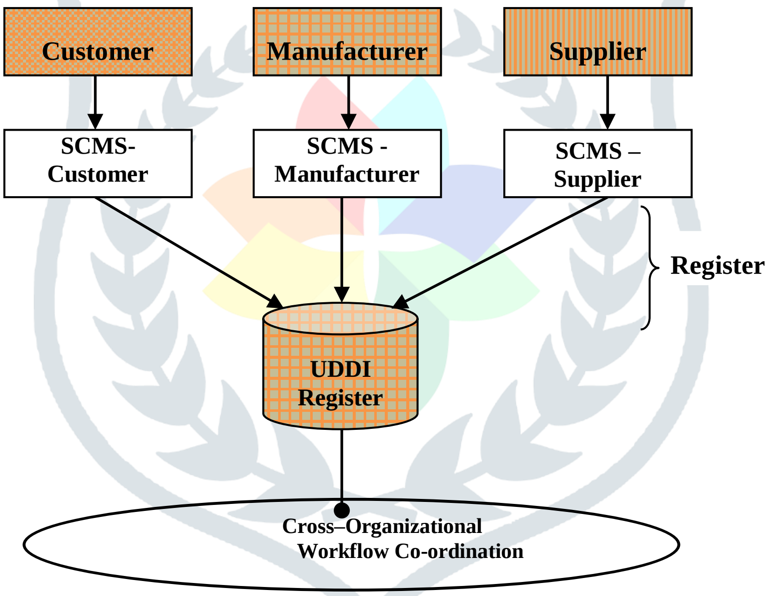


Figure-2 An Example of the SIW Environment

In this typical scenario, the Supply chain uses distributed service from different entities suppliers, customers and manufacturers. These entities register their offerings as web services in a distributed registry, such as a UDDI registry. Our proposed architecture of supply chain management consists of the stakeholder's supplier, manufacturer, and customer. Each entity typically has its own SCM system (SCMS), Hence we have SCMS-Supply, SCMS-Manufacturing, and SCMS-Customer should perform all supply chain activities in a coordinated fashion.

IV. WARP APPROACH

The WARP approach [13][14] is an agent-based middleware architecture. It consists of software agents that can be configured to control the workflow operation of distributed services. The WARP architecture consists of mainly two layers, the application coordination layer and the automated configuration layer.

In application coordination layer the workflow execution takes place by instantiating the workflow instances. The two types of agents that the application coordination layer consists:

1. Role Manager Agent (RMA) act as a broker for the execution of individual roles.
2. Workflow Manager Agent (WMA) act as a broker or middle layer agent for the management and coordination of workflow process, and applicable roles.

The WMAs contain the specification and policy of the workflow process. When a new process is configured, the WMAs accept workflow representations from workflow engine as input and write workflow policies to the centralized database, which is used as the agents' knowledge base. Then a new process specification is deployed to the RMAs. By executing one or more services RMA's full fill their role. The actual workflow execution takes place when the workflow instances are instantiated. At the beginning when the initiation event is written into the event server (based on tuple-space technology) RMAs execute one or more correct services based on its workflow role. WMAs communicate with the first role agent of RMAs via the event server. This makes RMA to invoke correct services based on localized knowledge of services and its workflow role The WMAs do not control the workflow execution, but it adds events to bring about the nonfunctional changes(such as exception handling, atomicity etc.) to the workflow process.

The automated configuration layer consists of the Site Manager Agents (SMA) and the Global Workflow Manager Agent (GWMA).The SMAs discover the service representations from atomic service components and save the service characteristics in the service–oriented data model as illustrated in Fig.3 below.

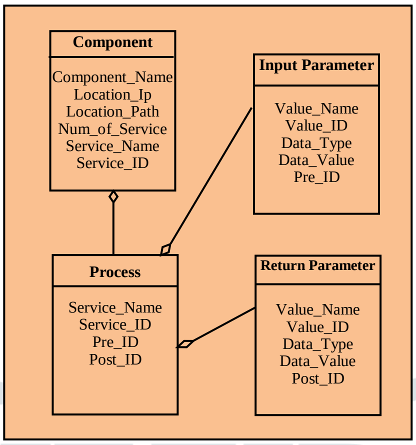


Figure-3 Service representations of atomic service components

The GWMA accepts two inputs:

1) One from SMA, the service representation

2) Other from the workflow designer, the workflow representation. Then write down the workflow policy to the centralized database.

Then the WMAs are configured by the GWMA to play the specific roles. After the completion of workflow-level configuration, the RMAs configured by the SMAs to play certain role as specified in the workflow database. In the WARP process the system takes workflow and the service-based information as input and coordinates the operation of a workflow among previously deployed and new components. The WARP architecture is a reflective process that allows it to discover new service Characteristics by introspection from the existing service components the main focus of the WARP research is to define the components and formalize the method to achieve semi-automated configuration. Here in step-1 the SMA discovers service representations from the existing service component. In step- 2, the SMA populates this Information into a workflow-based repository. In step-3 the workflow designer accesses the available services as Service Representation Views and creates a set of process models. In step-4, the GWMA gathers information from a Workflow Designer (human) and populates this information into the same workflow-based repository. In steps 5(i) and 5(ii), the application-layer agents, WMA and RMA are self configured and are deployed. Step 6 consists of the real-time coordination of the workflow operation

V. MODELLING VIEW'S IN WRAP APPROACH

The WRAP is a middleware agent architecture where the configuration requires the coordination between a workflow designer and many software agents. The configuration process mainly consists of (i) specification of various component based service and (ii) specification of workflow processes. In the top-down approach where web services are developed and described in WSDL and interfaces specified with SOAP, here in WARP where we use bottom-up approach, service representations are introspected from pre-existing services while humans enhance these representations with location and interaction-based information. In WRAP the services and their interaction can be Modelled in three different prospective forms (i) visual (ii) message (iii) database. (For persistence) We use Unified Modeling Language (UML) to model service and their interaction in the visual form, to model the messages we use XML. Finally we use a relational format for persistence storage.

Visual Modelling

Here we use UML to model different roles played in the workflow diagram, workflow structure, the control and the message flow in the workflow diagram and also the exception handling cases. Let us consider the workflow process in a supply chain management, before the modeling the WRAP architecture contain agents which introspect the existing atomic component and develop a base model. Here at the beginning there are few supply chain related services as shown in fig.4 with which the agents start their initial modeling .These are invocation based service described as class diagram. These are web services.

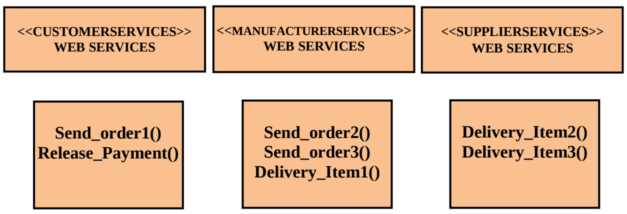


Figure-4 Service Representation View

The first human action in WRAP architecture is to model all the available roles. These roles must be mapped to different services to realize them. The three main roles in SCM are customer interface, manufacturer interface and supplier interface. This model is shown in fig.5 below in UML class diagram, where roles are associated to their corresponding services.

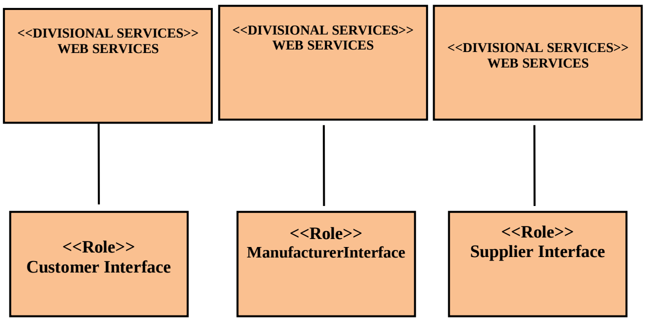


Figure-5 Role Association View

Roles are joined together to form a workflow. A workflow is named and modeled as a UML class called in our case as "supply chain" .The below fig.6 shows all the rolls associated with the workflow process in a typical SCM.

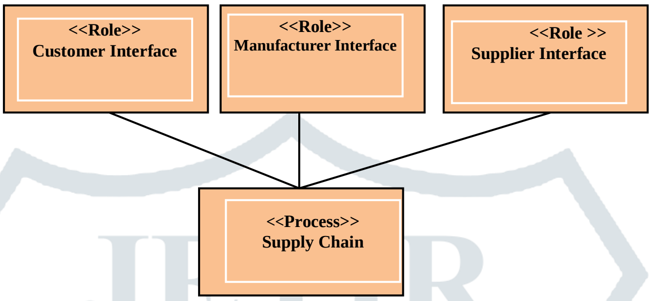


Figure-6 Role Join View

In visual modeling the control flow view and the message flow view are the main focus of attention in the workflow coordination. These models are depicted as UML activity diagrams, both for control flow and message flow. The Role Collaboration view shown in below Fig. 7 shows the different roles played and their services that must be executed throughout the workflow. In this workflow, we have different control flow branches. Activity diagrams shows workflow schema with forks that allow for multiple Paths.

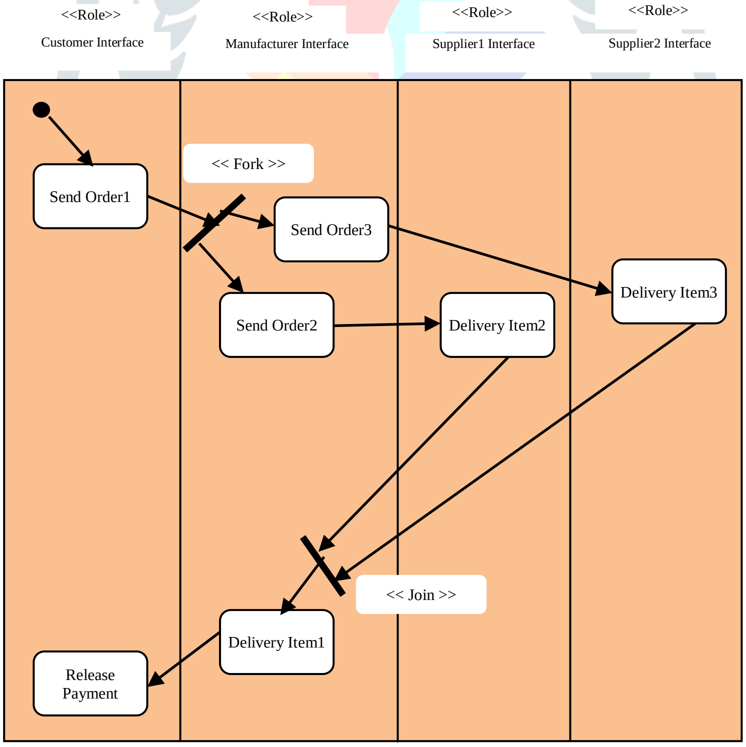


Figure-7 Role Collaboration View

As the workflow executes, the Message Collaboration View shows how the information is exchanged between various roles. The classes that represent the dataflow can map to complex XML schemas. This model is shown in fig. 8.

.

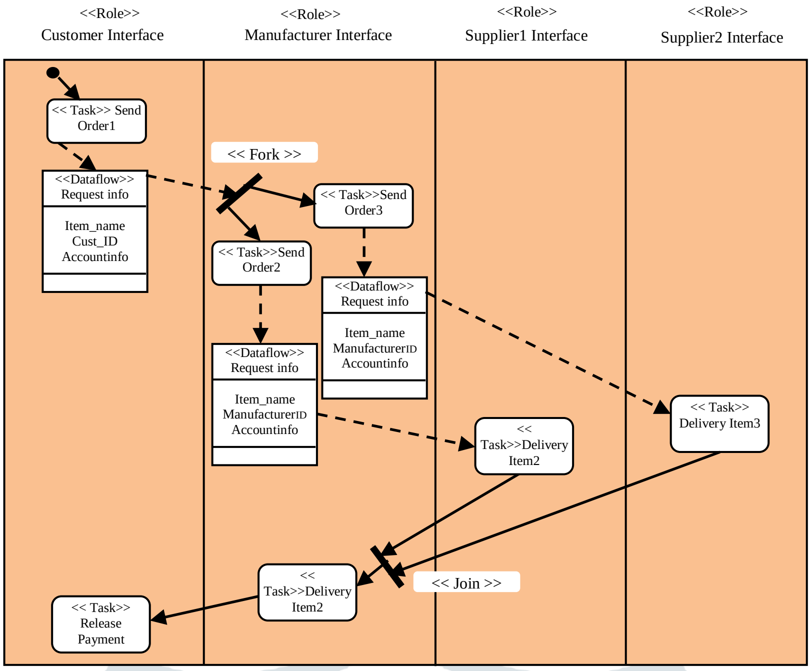


Figure-8 Message Collaboration View

The workflow model detailed in the above shows the information required to model a workflow of web service. Here we conclude that by using visual modeling the interpretation is much clear then compared to textual interpretation. In message modeling and database modelling we map these representations to XML messaging form and relational model which is not in the scope of this paper.

VI. Conclusion

The above studies shows that by using agent based coordination system , we can improve the coordination in a business process efficiently and effectively .In coordinating the tasks in the workflow of the entire Supply chain, the autonomous software agents play a vital role. The proposed software agents extend the capability of a workflow management system across the organizational boundaries. The use of an agent-based middleware in WRAP approach can be comparable to the use of SOAP in the web services. Also the use of XML message format can be seamlessly mapped to WSDL. In this regard some related work has been done by R. Tolksdorf [15], but that approach do not adopt the three modeling views (visual, message and database) of capturing interaction specifications.

In the future work we will study in more details how web services and agent technology is effectively used seamlessly in B2B integration. The main focus is to use modeling languages of WSDL, SOAP and to develop a run-time application in a business process domain.

References

[1] M.B. Blake, H. Gomaa, Object-Oriented Modeling Approaches to Agent-BasedWorkflow Services, in: C. Lucena, et al. (Eds.), Software Engineering to Large-Scale Mult-Agent Systems II, LNCS, vol. 2960, Springer-Verlag, Heidelberg, Germany, 2004Feb

[2] D. Simchi-Levy, P. Kaminsky, and E. Simchi-Levy, Designing and Managing the Suppry Chain, McGraw-Hill, 2000.

[3] Q. Xu, R. Qiu, and D. Russell, “Collaborative Supply Chain Management in Semiconductor Manufacturing Planning,” Proceedings of the 4th Intemational Conference on Control and Automation,Montreal, Canada, pp. 83-87, June 9-12,2003.

[4] The CHAIMS Project (2002): http://www-db.stanford.edu/CHAIMS/.

[5] D.W. Mennie, B. Pagurek, An Architecture to Support Dynamic Composition of Service Components, Proc. of the 5th International Workshop on Component-Oriented Programming –WCOP 2000, Sophia Antipolis, France, 2000 June, pp. 1 – 8.

[6] M. Fox, M. Barhuceanu, and R. Teigen, “Agent oriented Supply-chain Management,” The International Joumal of Flexible Manufacturing Systems, Vol. 12, pp.165-188,2000.

[7] T. Hess, L. Rees, and T. Rakes, “Using Autonomous Software Agents to Create the Next Generation of Decision Support Systems,” Decision Sciences, Vol. 31, No. 1, pp. 1-31,2000.

[8] Web Services (2002): http://www.w3.org/2002/ws/desc/.

[9] SOAP (2002): http://www.w3.org/TR/soap12-part0/.

[10] P. Massimo, K. Sycara, T. Kawamura, Delivering Semantic Web Services, Proceedings of the WWW2003, Budapest,Hungary, 2003 May.

[11] UDDI (2002): http://www.uddi.org/.

[12] UDDI Specification 2.04 (2002): http://www.uddi.org/pubs/ ProgrammersAPI-V2.04-Published-20020719.pdf.

[13] Blake, M.B. " Agent-based Workflow Configuration and Management of On-line Services", Proceedings of the International Conference on Electronic Commerce Research (ICECR-4), pp 567-588, Dallas, TX, November 2001

[14] Blake, M.B. " WARP: Workflow Automation through Agent-Based Reflective Processes ", Proceedings at the 5th International Conference on Autonomous Agents, Montreal, Canada, May 2001 (software demonstration)

[15] Tolksdorf, R. “Coordination Technology for Workflows on the Web: Workspaces. In Proceedings of the Fourth International Conference on Coordination Models and Languages COORDINATION 2000, LNCS, pages 36–50.Springer-Verlag, 2000.