IDEA GROUP PUBLISHING
701 E. Chocolate Avenue, Suite 200, Hershey PA 17033, USA
Tel: 717/533-8845; Fax 717/533-8661; URL-http://www.idea-group.com
ITP4691

Distributed Object Based SCM Simulator: LOSIMOPU

Hiroshi TSUJI, Takefumi KONZO, Ryosuke SAGA
Graduate School of Engineering, Osaka Prefecture University
1-1 Gakuen-cho Sakai, Osaka, Japan 599-8531
P: +81-72-254-9353, F: +81-72-254-9915, tsuji@ie.osakafu-u.ac.jp

INTRODUCTION

Under the current Internet era, many companies are interested in supply chain management (SCM) in order to rebuild their business process [1]. Especially, constraints theory on throughput [2] that requires the managers to detect the bottlenecks in the business process has impacted many industries. On SCM, there are two research approaches: (1) OR/MS technology such as mathematical programming that optimizes the objective function in order to allocate resources such as persons, machines, and money [3], and (2) IT technology such as workflow management and electronic payment that integrates the enterprise systems [4].

These researches have been done independently. However, not only to assess the risk and the chance of the business process reengineering but also to evaluate the implementation issues, the SCM designers require the means that integrate both OR/MS technology and IT technology. Therefore, we plan to develop software simulator called LOSIMOPU (LOgistics SIMulator by Osaka Prefecture University) by the distributed object technology [5]. This paper describes the target, the system overview and the design issues of our LOSIMOPU.

TARGET OF SCM SIMULATOR LOSIMOPU

Our purpose is to develop the software simulator designed for the internet based supply chain management that assigns the value of system parameters as stochastic variables and presents the simulation result visually. Although the spreadsheet based risk simulator is well known [6], it is difficult for the designer to express the concurrent and complex supply chain shown in Fig. 1 into the computational model.

Essentially, SCM is an autonomous and distributed system. There are varieties of players for the business: customers, vendors, manufactures, and parts suppliers. While they hide their internal information and behave actively and/or passively, they keep the same information structure such as company property and balance, and the same activity such as "buy" and "sell". Sometimes, they exchange information asynchronously in order to collaborate with each other and update the status for the trade. Each player has its own policy on trade and its own capacity for manufacturing and inventory.

Our strategy for the development is as follows:

(1) We first develop LOSIMOPU as the learning system that teaches the general users what the supply chain management is;
(2) Next we enhance it as the research platform that the experts can include the supplemental components and the special functions in the simulator;
(3) Finally, we expect it can be used in industry.

To realize our strategy, the distributed object oriented modeling technology is suitable to express the supply chain model. In fact this technology is beneficial on the following items [5]:

(1) Improved availability,
(2) Flexible configuration,
(3) Localized control and management,
(4) Incremental system expansion.

To follow the steps, we design LOSIMOPU as follows:

(1) It has basic model that any users can runs easily;
(2) It opens its application interface in order that researchers can implement their OR/MS idea;
(3) It adopts the international standard protocol/interface for the practical use.

SYSTEM CONFIGURATION OF LOSIMOPU

Let us analyze use cases in the SCM model first. In general, the player in the supply chain has the common actions as shown in Figure 2: request, reply, sell, buy and manufacture. Of course, according to the core competence of the player, some activities should be added and the others should be omitted. For example, the customer does not have the function "sell" while the end supplier does not have the function "buy".

Further, each player has the property and the account data in time series. Thus, the participants are expressed as class-subclass relation as shown in Figure 3.

There are five player subclasses in our simulator: (1) Customers, (2) Intermediate suppliers, (3) End suppliers, (4) Electronic payment servers, and (5) Transportation servers. Each player objects in the supply chain runs concurrently [5]. Communication protocol among objects is built in standard formula such as XML and SOAP. Note that the activities strategy of a player is different from each other according to its capacity and policy and is hid in the object. The strategy is often designed by OR/MS technology. Individual information is as follows:

(1) Intermediate Supplier Class
It has information on components that constitutes each goods and information on capacity for manufacturing and inventory. Such information is used to control the inbound of the supply chain. It has the mechanism for the economic order quantity and material requirement planning [1]. On the other hand, it has sales log and customer information that controls the outbound of the supply chain. It has mechanism for the customer relationship management. Further, it keeps management index such as cost, income, real estate and so on. Those are indispensable for calculating the interest, the throughput, and the loss of sales chances.
(2) End Supplier Class
In general, there are the competitors for the parts supply. They have price table and lead time information on parts delivery. For LOSIMOPU, the latter is assigned as stochastic variable.
(3) Customer Class
They play roles for the demanders. The demand information includes goods identification, volumes, and time-limit for the delivery. LOSIMOPU should allow the user to assign a variety of random value for the customer order arrival.
(4) Electronic Payment Server Class
They play roles for the electronic payment among the players. It simulates the time delay on the account DB as business process and charges the payment.
(5) Transportation Server Class
It delivers parts to the manufacturer and delivers goods to the customers. It has price table that is a function of transportation lots and distance.

WORKFLOW IN LOSIMOPU

Let us describe the concept of the workflow in LOSIMOPU. Each player receives the requests from its customers. Then it makes decision whether it accepts the request and replies to the requester.

LOSIMOPU has an e-Market place engine. It received the message from the players, records audit and transferred to the other players. The action of e-Market place is shown in Figure 4. The e-Market place has function that visualizes goods flow, money flow and information flow for the analysis. The visualized information also includes the management index and the risk.

Figure 1. Example of concurrent and complex supply chain

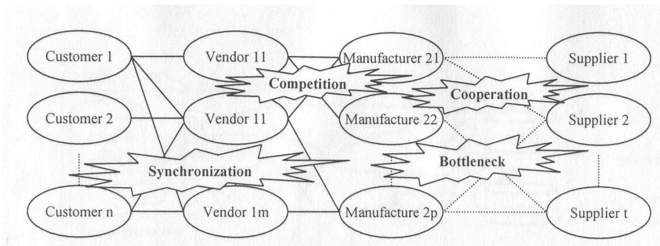


Figure 2. Use Case for intermediate supplier

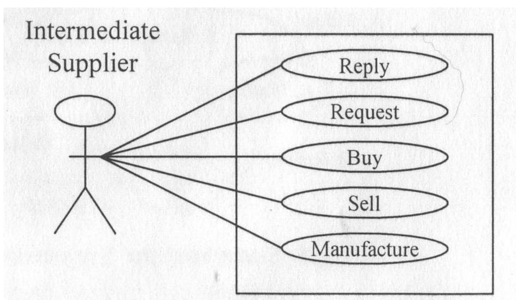


Figure 3. Class diagram of players

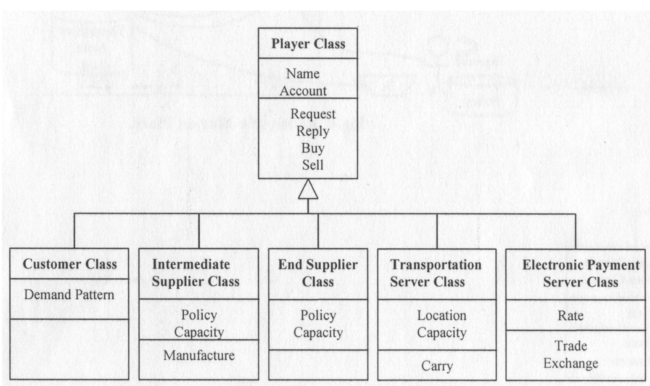


Figure 4. Action of e-Market place

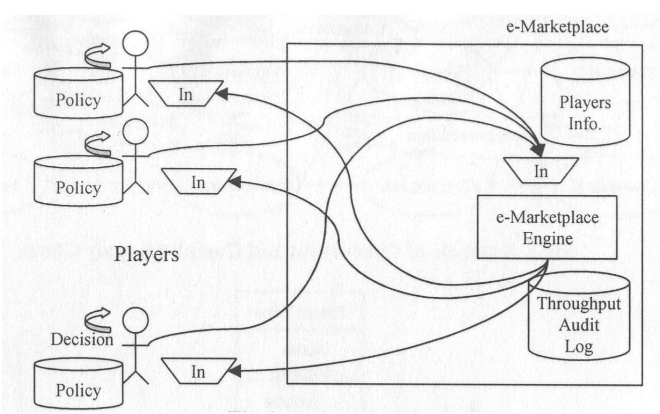
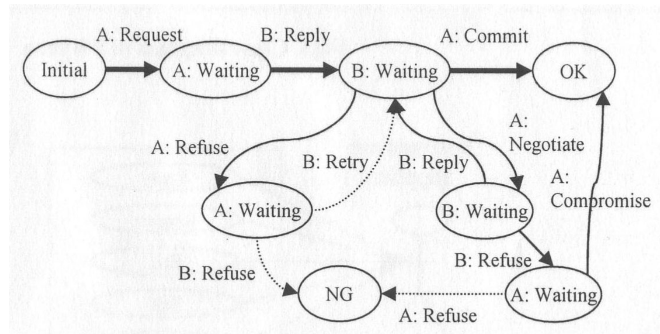


Figure 5. Statechart for transaction



Each transaction has the state in LOSIMOPU and it starts as "initial" state. If the trade is established, the transaction finalizes as "OK" state, otherwise as "NG" state. The state chart for the transaction is shown in Figure 5. The basic idea in this statechart is borrowed from Language/Action theory [7].

CONCLUSION AND DISCUSSION

We have described the target and overview of LOSIMOPU. The project is now on going. We have gotten the use cases, class structure and state chart for the conceptual design. To develop the basic model of LOSIMOPU, the design issues are as follows:

(1) Dynamic modeling (object interaction modeling, dynamic analysis);
(2) Software architecture design (system decomposition, subsystem structuring criteria);
(3) Detail software design (user interface design, object state visualization, performance).

Other issues are to develop the open interface for the following research:

(a) Agent behavior that negotiates between buyers and sellers are implemented as an application model of LOSIMOPU.
(b) It simulates the tracking system that monitors the status of the goods that were ordered by the customers. It is possible to analyze the complexity between the manufacturer and the others by implementing the cancel protocol.
(c) In the case that the demand is greater than the supply, LOSIMOPU detects the bottleneck of the supply chain and presents the alternatives that dissolve the bottleneck. Therefore, LOSIMOPU can be also the infrastructure for analyzing the chain.
(d) It can be platform that the user develop and analyze the campaign model and the pricing model.

Those are application examples of LOSIMOPU. Adding the concurrent and distributed objects (for examples, warehouse and wholesalers objects) to the basic model, the user of LOSIMOPU can continue to enhance the SCM simulator.

REFERENCES

[1] J. Coyle, et al.: The management of Business Logistics, West Publishing Company (1996).
[2] E. Goldratt, J. Cox: Goal: A Process of Ongoing Improvement, Penguin Highbridge Audio Published (1993).
[3] F. Karaesmen, et al.: Integrating advance order information in Make-to-stock production systems, Vol.34, pp. 649-662 (2002).
[4] S. Knoshafian and M. Buckiewica: Introduction to Groupware, Workflow, and Workgroup Computing, John Wiley & Sons, Inc. (1995).
[5] H. Gomaa: Designing Concurrent, Distributed, and Real-time Applications with UML, Addison Wesley (2000).
[6] J. R. Evans and D. L. Olson: Introduction to Simulation and Risk Analysis, Prentice Hall, Inc., (1998).
[7] T. Winograd: A Language/Action Perspective on the Design of Cooperative Work, In: I. Greif (ed.), Computer Supported Cooperative Work: A Book of Readings. Morgan Kaufmann, San Mateo (1988).

Related Content