The Design and Implementation of the Integrated Supply Chain Management System Based on UML and J2EE Technology

Yongchun Wu, Zhenjian Jiang, Na Han

Business School, Shandong Jianzhu University, Jinan 250101, China
E-mail: clarence-wyc@126.com

Abstract: Supply Chain Management (SCM) is a hotspot both in academic field and in modern enterprise. Under the new competition environment of the economic globalization, in order to adapt to the demand that the market has brought forward with the high quantity, the high flexibleness and the low cost, it's necessary to integrate the interior supply chain in business enterprise with the exterior supply chain in business enterprise, so as to attain the global dynamic superior target. Consequently it's important to research and develop the Integrated Supply Chain Management System (ISCMS) based on J2EE. The SCM's theory and its development is studied, analyzed the problems in general in the supply chain as well as the necessity for researching the ISCMS. The architecture of ISCMS is proposed. Finally, the plan to design and implement the ISCMS based on UML and J2EE technology is expounded.

Key Words: Supply Chain, Supply Chain Management, Integrated Supply Chain Management System, UML, J2EE

I. INTRODUCTION

In the modern world the main focus of competition is not only between different companies but also between supply chains. The supply chain is a network of suppliers, factories, warehouses, distribution centers and retailers through which raw materials are acquired, transformed and delivered to the customer. Its diagram is shown in Figure Ⅰ. While the separation of supply chain activities among different companies enables specialization and economies of scale, there are many important issues and problems that need to be resolved for successful supply chain operation – this is the main purpose of SCM. SCM is "the integration of key business processes from end user through original suppliers that provide products, services, and information that adds value for customer and other stakeholders" (Chan & Qi, 2003). SCM emphasizes the integration of activities and information flows while focusing on increasing value for the customer. The diversity of participants in terms of size, technological capabilities, cultural differences, efficiencies etc. makes companies a difficult task.

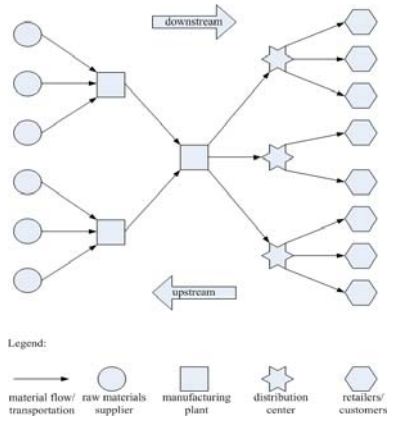


FIGURE Ⅰ AN EXAMPLE OF A SUPPLY CHAIN

This work is supported by Nature Science Foundation of Shandong Province under Grant ZR2009HZ006

II. COMPANIES' PROBLEMS IN GENERAL IN THE SUPPLY CHAIN

Traditionally the different echelons of the supply chain and also the different decisions have been considered separately for planning purposes. Companies have been dealt with on an individual basis rather than collectively. There are a lot of problems in the un-integrated supply chain. Some of them are listed below:

A. Decisions Problems

Firstly, companies face a huge number of problems, such as how to make decisions concerning production planning, inventory management and vehicle routing. These three decisions are managed separately in most organizations because making each individual decision is very difficult, since many constraints have to be satisfied.

Secondly, the problem is yet harder in reality because the decisions concerning production planning, inventory management and vehicle routing are interdependent. Hence, these three decisions should be taken together, which makes the planning problem harder.

Thirdly, companies are not isolated, but impact on and are impacted by their partners. As a result, when a company maximizes its profits, it may disturb other companies, which may result in globally under optimal decisions, because organizations may have different conflicting objectives. The best solution would be to make the decisions together concerning production planning, inventory management and vehicle routing for several companies. As this planning problem is hard for a single company, synchronizing all companies' decisions together is very hard.

B. The Bullwhip Effect

The bullwhip effect occurs when the demand order variations in the supply chain are amplified as they move up in the supply chain. Five possible sources for bullwhip effect are recognized in the literature. They include: demand forecast updating, prize

fluctuation, rationing and shortage gaming, order batching and none-zero lead time.

In addition to greater safety stocks the described effect can lead to either inefficient production or excessive inventory as the producer needs to fulfil the demand of its predecessor in the supply chain. This also leads to a low utilization of the distribution channel. Despite of having safety stocks there is still the hazard of stock-outs which result in poor customer service. Furthermore, the Bullwhip effect leads to a row of financial costs. Next to the (financially) hard measurable consequences of poor customer services and the damage of public image and loyalty an organization has to cope with the ramifications of failed fulfillment which can lead to contract penalties. Moreover the hiring and dismissals of employees to manage the demand variability induce further costs due to training and possible pay-offs.

III. The Necessity for Researching the Integrated Supply Chain Management System

In order to solve those problems and optimize the supply chain performance, supply chain functions must operate in an integrated manner. But the dynamics of the companies and the market make this difficult; materials do not arrive on time, production facilities fail, workers are ill, customers change or cancel orders, etc. causing deviations from plan. In some cases, these events may be dealt with locally, i.e., they lie within the scope of a function. In other cases, the problem cannot be "locally contained"; modifications across many functions are required.

Now the definition of the Integrated Supply Chain Management is a process-oriented, integrated approach to procuring, producing, and delivering products and services to customers. The Integrated Supply Chain Management has a broad scope that includes sub-suppliers, suppliers, internal operations, trade customers, retail customers, and end users. The Integrated Supply Chain management covers the management of material, information, and funds flows.

Consequently, the Integrated Supply Chain Management System must coordinate the revision of plans/schedules across supply chain functions. The agility with which the supply chain is managed at the strategic, tactical and operational levels in order to enable timely dissemination of information, accurate coordination of decisions and management of actions among people and systems, is what will ultimately determine the efficient, coordinated achievement of companies' goals.

IV. The Architecture of the Integrated Supply Chain Management System Based on J2EE Technology

The J2EE platform uses a multi-distributed application model. This means application logic is divided into components according to function, and the various application components that make up a J2EE application are installed on different machines depending on which tier in the J2EE environment the application component belongs. The following is a list of components at different levels:

- Client tier components run on the client machine
- Web tier components run on the J2EE server
- Business tier components run on the J2EE server
- Enterprise information system (EIS) tier software runs on the EIS server

According to the J2EE typical architecture and the characteristics of the integrated supply chain management system, the architecture of the Integrated Supply Chain Management System, which is shown in Figure Ⅱ:

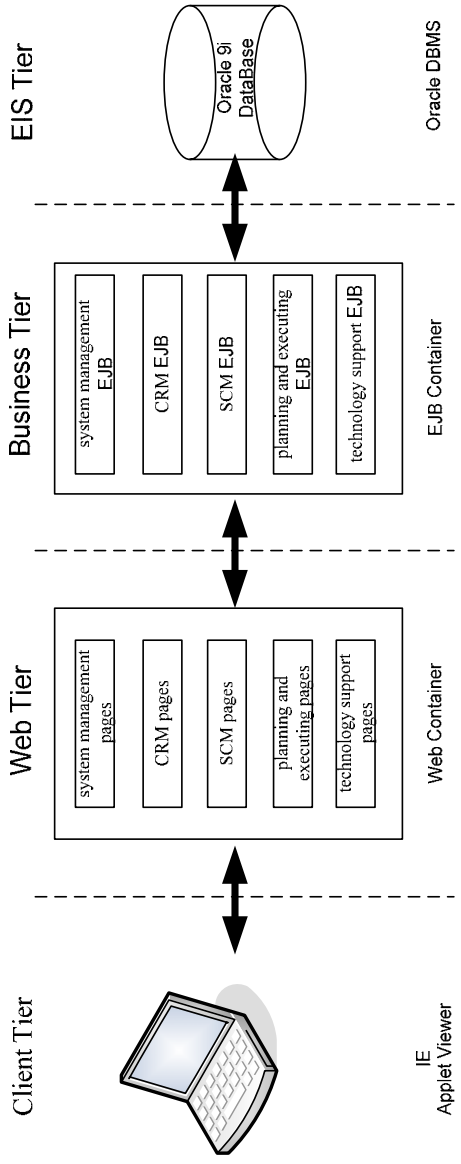


FIGURE Ⅱ THE ARCHITECTURE OF THE INTEGRATED SUPPLY CHAIN MANAGEMENT SYSTEM BASED ON J2EE TECHNOLOGY

V. THE DESIGNING AND IMPLEMENTING OF THE INTEGRATED SUPPLY CHAIN MANAGEMENT SYSTEM

A. Functional designing of the Integrated Supply Chain Management System

The Integrated Supply Chain Management has the main functions, which is shown in Figure Ⅲ.

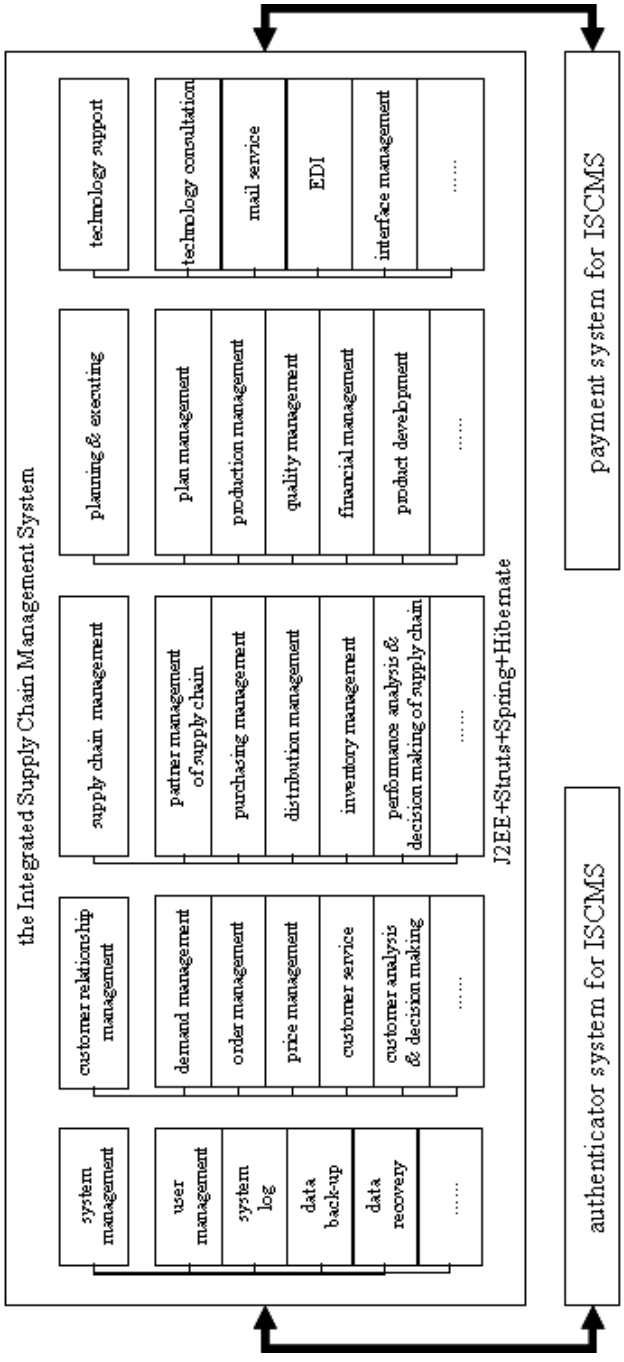


FIGURE III THE FUNCTIONS OF THE INTEGRATED SUPPLY CHAIN MANAGEMENT SYSTEM BASED ON J2EE TECHNOLOGY

The entire system is divided into system management, customer relationship management, supply chain management, planning and executing, as well as technology support. The system integrates the well-known open source framework: Struts, Spring and Hibernate, designing MVC mode by using J2EE technology. Struts is used in the web tier, Spring is used in the business tier; and Hibernate is used in the EIS tier.

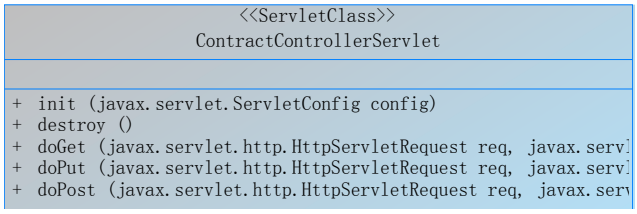
B. Business logic implementation of the Integrated Supply Chain Management System

Due to the complexity of the system and limit the length of this paper, taking the inventory management of supply chain management for an example, the Integrated Supply Chain management System is built based on J2EE technology within the framework: Struts, Spring and Hibernate. Inventory control problem is one of the important elements of supply chain management, through strengthening the inventory control, the systematicness and integrativity of the supply chain can increase, as well as the agility and responsiveness of companies can strength. The joint managed inventory model is applied in this paper, which is a risk-sharing inventory management. The basic idea of the joint inventory management is raw material suppliers and manufacturing plant set up the joint inventory of materials, respectively, manufacturing plant and distribution center set up the joint inventory of the production and sales, which break the tradition of each node of the supply chain having their own inventory models. In this model, the node enterprises corporately formulate inventory plan. This model can be a good solution to bullwhip effect of the supply chain system as a result of each node having their own inventory, which maintains the expected demands of the two nodes to be consistent and lower the risk and cost of inventory. The collaboration diagram is shown in Figure Ⅳ.

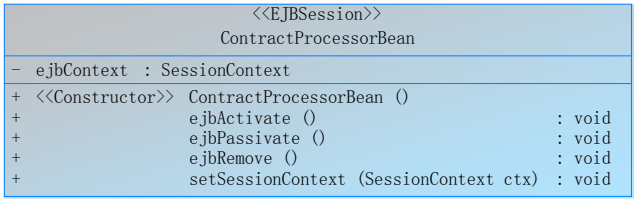
1) Business logic implementation

By Figure Ⅲand Figure Ⅳ, the business processes of inventory management system should be designed to be stateless session bean. For the sake of image, all EJBs and Servlets are described through diagrams.

ContractControllerServlet



ContractProcessorBean



JMIProcessorBean

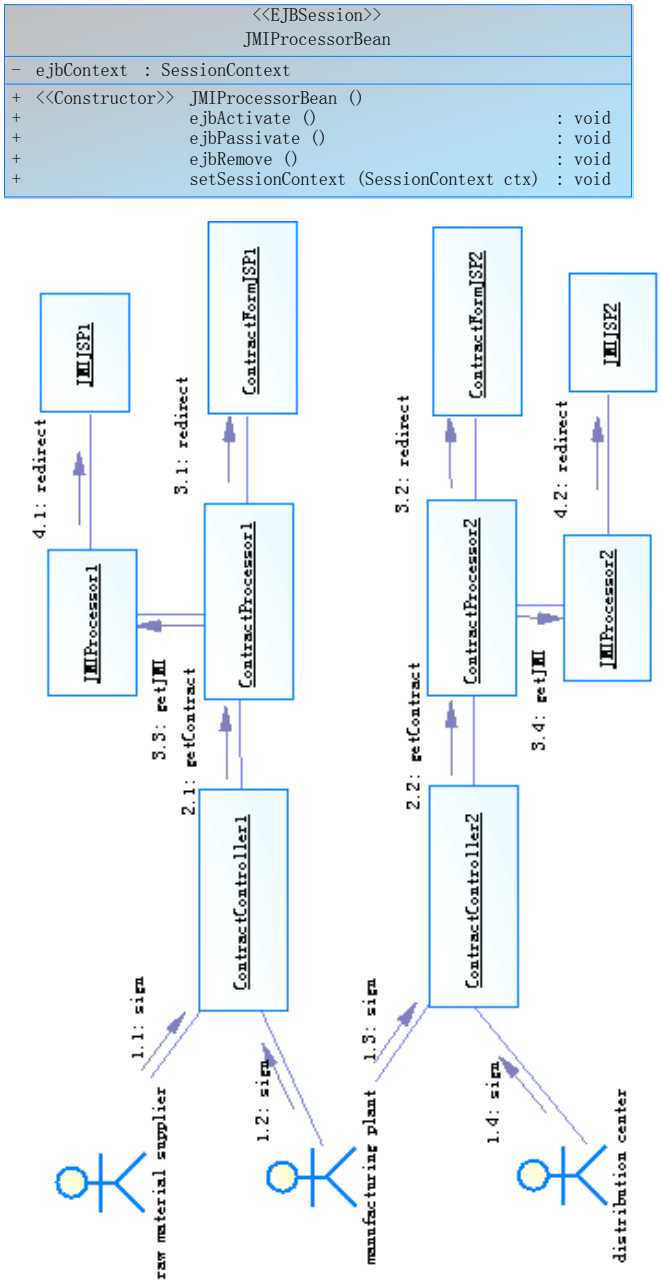


FIGURE IV THE COLLABORATION DIAGRAM OF INVENTORY MANAGEMENT

2) The developing steps of the JMI function

(1) ContractControllerServlet components run in the ContranctControllerServlet containers, which introduce java.util.*, java.servlet.*, javax.servlet.http.*, java.io.*, achieve javax.servlet.http.HttpServlet interface, and go through the javax.servlet.http.HttpServletRequest interface making a request, go through the javax.servlet.http.HttpServletResponse interface receiving response, and ServletClass calling the init (), destroy (), doGet (), doPut (), doPost (). HttpServlet achieves Servlet interface, and in accordance with the request (get or Post) calls the appropriate method (doGet or doPost). Under normal circumstances, the subclasses of HttpServlet can rewrite its doGet methods for dealing with Get request, and rewrite its doPost methods for dealing with Post request. The basic framework of the code is as follows:

```
import java.util.*;
import javax.servlet.*;
import javax.servlet.http.*;
import java.io.*;
public class ContractControllerServlet extends javax.servlet.http.HttpServlet {
  public void init(javax.servlet.ServletConfig config) throws javax.servlet.ServletException {
    // TODO: implement
    super.init(config);
  }
  public void destroy() {
    // TODO: implement
    super.destroy();
  }
  public void doGet(javax.servlet.http.HttpServletRequest req, javax.servlet.http.HttpServletResponse res) throws javax.servlet.ServletException, java.io.IOException {
    // TODO: implement
    res.setContentType("text/html");
    PrintWriter out = res.getWriter();
    out.println("<HTML>");
    out.println("<HEAD>");
    out.println("<TITLE>Add the title here</TITLE>");
    out.println("</HEAD>");
    out.println("<BODY>");
    out.println("HTTP servlet: doGet()");
    out.println("</BODY>");
    out.println("</HTML>");
    out.close();
  }
  public void doPut(javax.servlet.http.HttpServletRequest req, javax.servlet.http.HttpServletResponse res) throws javax.servlet.ServletException, java.io.IOException {
    // TODO: implement
    res.setContentType("text/html");
    PrintWriter out = res.getWriter();
    out.println("<HTML>");
    out.println("<HEAD>");
    out.println("<TITLE>Add the title here</TITLE>");
    out.println("</HEAD>");
    out.println("<BODY>");
    out.println("HTTP servlet: doPut()");
    out.println("</BODY>");
    out.println("</HTML>");
    out.close();
```

```
  }
  public void doPost(javax.servlet.http.HttpServletRequest req,
javax.servlet.http.HttpServletResponse res) throws
javax.servlet.ServletException, java.io.IOException {
    // TODO: implement
    res.setContentType("text/html");
    PrintWriter out = res.getWriter();
    out.println("<HTML>");
    out.println("<HEAD>");
    out.println("<TITLE>Add the title here</TITLE>");
    out.println("</HEAD>");
    out.println("<BODY>");
    out.println("HTTP servlet: doPost()");
    out.println("</BODY>");
    out.println("</HTML>");
    out.close();
  }
}
```

(2)Implementing JMIProcessorEJB class: JMIProcessorBean class is the enterprise bean class, which structure is determined by the EJB class, JMIContractProcessorEJB is designed to have a stateful session bean, so they need to achieve javax.ejb.SessionBean interface, introducing the java.util.*, javax.ejb.*, EJBSession calls the JMIProcessorBean (), ejbActivate (), ejbPassivate (), ejbRemove (), setSessionContext (), ejbCreate () method. The basic framework of the code is as follows:

```
import java.util.*;
import javax.ejb.*;
public class JMIProcessorBean implements
javax.ejb.SessionBean {
 private SessionContext ejbContext;
  public JMIProcessorBean() {
    // TODO: implement
  }
  public void ejbActivate() throws javax.ejb.EJBException {
    // TODO: implement
  }
   public void ejbPassivate() throws javax.ejb.EJBException {
    // TODO: implement
  }
  public void ejbRemove() throws javax.ejb.EJBException {
    // TODO: implement
  }
  public void setSessionContext(SessionContext ctx) throws
javax.ejb.EJBException {
    this.ejbContext = ctx;
  }
}
```

VI. CONCLUSION

The JSP, Servlet, EJB components based on J2EE platform combine with the thinking of the three-tier structure and the design mode, which can simplify the development process, rapidly building the Integrated Supply Chain Management System of a good scalability, maintainability, reliability and high availability. From the analysis to the application, starting from the analysis of key technologies, this paper put forward a complete design proposal of the Integrated Supply Chain Management System based on J2EE. On the one hand, from the management, economic and technical feasibility, this system is analyzed and verified, on the other hand, combining the characteristics of J2EE technology, the practicability is discussed and is proved that this system meets resource integration requirements of companies in supply chain, which lays a solid foundation to study the next-generation integrated supply chain management system based on open platform. It uses a new design concept based on J2EE technology, is bound to affirmation and development in the developing and application of supply chain management system.

ACKNOWLEDGMENT

The authors would like to thank Haizi WANG and Li ZHAO from Business School at Shandong Jianzhu University, as well as the anonymous referees for valuable comments and critique during the review process of this paper. None besides the authors, however, can be held responsible for the result.

REFERENCES

[1] Junhou Li, Research and Implementation of the Integrated Supply Chain Management System Based on J2EE, Huazhong University of Science & Technology, 2004.

[2] Li Tu, Zhongxiang Fang and Feng Chen, Research on the Supply Chain Management System Based on J2EE, Machine Tool & Hydraulics, No.7, pp.215-218, 2006.

[3] Nikhil Ajay Pujari, Integrated Supply Chain Optimization Model Using Mathematical Programming and Continuous Approximation, Ohio University, 2005.

[4] Ming Dong, Process Modeling, Performance Analysis and Configuration Simulation in Integrated Supply Chain Network Design, Virginia Polytechnic Institute and State University, 2001

[5] Haiyan Hu, The Design and Implementation of Supply Chain Management, Tsinghua University, 2004.

[6] Shijun Tang, Design and Implement of Agile Supply Chain Management System Based on Struts and Hibernate 4-layer Framework, Central China Normal University, 2006.

[7] Haofei Zou, Yuming Jiang and Jie Wang, A Study on supply chain management for virtual enterprises, Machinery Manufacturing Engineer, No.4, pp.35-37, 2003.

[8] Xiaomei Wang, Development of Customer Relationship Management System Based on Web, Modern Electronic Technique, No.14, pp.40-41, 2005.

[9] Yong Li, Hongwei Zhao and Jianyi Guo, Design and Realization of the Customer Relationship Management Application System, Journal of Kunming University of Science and Technology (Science and Technology), Vol.29, No.3, pp.44-47, 2004.

[10] Lei Yang, JMI faced to Supply Chain Management, Dalian Maritime University, 2006.

[11] Thierry Moyaux1, Brahim Chaib-draa1, and Sophie D'Amour, Supply Chain Management and Multiagent Systems: An Overview.