KNOWLEDGE-BASED SIMULATION FOR SUPPLY CHAIN INTEGRATION

CHAN YUEN LING

M. PHIL.

THE HONG KONG POLYTECHNIC UNIVERSITY

2008

THE HONG KONG POLYTECHNIC UNIVERSITY

DEPARTMENT OF INDUSTRIAL AND SYSTEMS ENGINEERING

Knowledge-based Simulation for Supply Chain Integration

By

CHAN Yuen Ling

A thesis submitted in partial fulfillment of the requirements for the Degree of Master of Philosophy

August 2006

CERTIFICATE OF ORIGINALITY

I hereby declare that this thesis is my own work and that, to the best of my knowledge and belief, it reproduces no material previously published or written, nor material that has been accepted for the award of any other degree or diploma, except where due acknowledgement has been made in the text.

______________________________ (Signed)

CHAN YUEN LING (Name of student)

Abstract

In a competitive business environment, it is necessary to adopt a new supply chain strategy from time to time in order to increase competitiveness. However, most small and medium-sized enterprises (SMEs) do not have the guidelines necessary for selecting the most suitable strategy to quickly achieve success. If they use the traditional trial and error approach, a great amount of effort and heavy costs would be incurred. Once an unsuccessful strategy is implemented, the loss to the parties involved would be considerable.

In supply chain management, demand uncertainty is the most critical problem that most parties have to face in their current practices. To cope with the problem, retailers attempt to keep a high level of inventory or order larger quantities than needed from their suppliers. The high inventory costs are charged to the retailers. A vendor managed inventory (VMI) strategy is a way of solving such kinds of problems. Low inventory levels will be kept by the retailers and the lead time will be reduced.

Fast responsiveness to customers is one of outcomes that can be achieved by the VMI strategy. It is also a key factor to drive the supply chain to success. However, the adoption of a new supply chain strategy needs a large amount of effort. Besides, most SMEs lack a simulation platform to mimic the effect of a change of supply chain strategies. It would be better if the SMEs predict the effects, caused by a new strategy, at the redesigning stage. This will help with reduction in the level of inventory, lead time, unfilled rate and stock-out rates. The platform can thus play an important role in the selection of a suitable supply chain strategy, at the preliminary stage.

In the present study, a knowledge-based simulation platform (KBSP) is proposed to enhance the competitiveness of all parties in the supply chain. It aims to accumulate successful experiences that SMEs have had in formulating and implementing competitive supply chain strategies. Based on many unpredictable factors, the KBSP is provided for multiple retailers and a supplier who can simulate the processes of a VMI strategy. The outcomes of the implementation of the supply chain strategy can be derived from the simulation. The potential benefit of the KBSP is to provide a simulated environment for the SMEs, which will reduce the bullwhip effect due to demand uncertainty. A KBSP prototype has been designed and developed, and a trial run for evaluation of its performance has been conducted at a selected reference site.

During the simulation, spectrum analysis is used to transfer patterns of demand to a series of spectra from which features of the patterns are extracted for comparison. History logs of processes in the simulation and terms in an agreement are valuable knowledge. This knowledge is captured, reused and retained in a knowledge repository (KR). The knowledge contains, for instance, problems that are solved, and details of how and why certain decisions were made. New staff are able to learn about a number of successful cases which have been accumulated in the KR.

Angus Electronics Company Ltd. was selected as the site for the case study in which the system was trial implemented. In this case, the VMI strategy was selected for the simulation of the trial implementation. Compared with the traditional approach, that is, the push system, the simulation results show how inventories and the stock-out rate were significantly reduced, and how the fill rate was increased. The change of the strategy from the push system to the VMI strategy indicates that the strategic change is beneficial for Angus in that it can reduce the level of inventory and reduce the lead time.

The theoretical background and know-how of this study forms the basis for further development of the proposed KBSP to handle complex supply chain problems. It is suggested that KBSP should be further developed and implemented in manufacturing company. Once the KBSP is further implemented successfully, the risks, such as demand uncertainties, would be identified and reduced. In addition, it is proposed that the KBSP is integrated with enterprise resources planning (ERP) systems, which provide a more realistic environment for different parties to conduct simulations. It is suggested that the KBSP is further developed so that more supply chain strategies can be made available for parties who plan to redesign their current strategy.

Acknowledgements

The candidate would like to express her sincere thanks to her chief supervisor, Dr. C.F. Cheung, Associate Professor of the Department of Industrial and Systems Engineering, The Hong Kong Polytechnic University for his advice, support, guidance and encouragement during her research study.

The candidate would also like to thank Professor W.B. Lee, Head of Department and Chair Professor of the Department of Industrial and System Engineering, and Dr. S.K. Kwok, lecturer in the same Department, for their helpful advice and support during the research project. Many thanks also to the candidate’s family and friends, for their encouragement, support and help.

Finally, the candidate would like to thank the academic staff and technical staff in the Department of Industrial and Systems Engineering. They have shared their valuable experience and provided technical support for the study.

Table of Contents

List of Figures

List of Tables

Chapter 1 Introduction

In the 1980s, supply chain management (SCM) was first introduced from the field of business. SCM is defined as the integration of business processes from the supply of raw materials, production processes, distribution, to delivery of products and services to end customers. In the early stage of SCM, supply chain parties focused on how to achieve the targets of a low inventory level, fast customer response and high profits in their enterprises. Nowadays, the concept of SCM is slightly different in that the emphasis is on how to work well with other parties rather than competing with them.

Supply chain integration is a strategic concept that is essential for those parties who plan to integrate a part of supply chain operations with their upstream suppliers or downstream customers for business process improvement, achievement of high customer satisfaction, on-time product delivery and low operational costs. In the present study, a number of supply chain strategies for supply chain integration are investigated. Four main problems encountered by supply chain parties in the process of supply chain integration are defined. To solve these problems, the knowledge domain for supply chain strategy, knowledge management and simulation are studied and applied to establish a knowledge-based simulation platform (KBSP) for supply chain integration.

1.1 Background of Study

In SCM, supply chain integration is a way to enhance competitiveness in the marketplace. SCM is defined as the mutual collaboration among supply chain parties for making decisions on strategic, tactical and operational issues (Bagehi & Skjoett-

Larsen, 2005). To achieve the goal of supply chain integration, some supply chain strategies are designed and proposed. These are the operational strategy of Just-in-time (Morash, 2001), managerial strategy of customer closeness (Morash, 2001), vendor managed inventory (VMI) (Barratt & Oliveira, 2001; Waller et al., 1999; Holmstrom, 1998(a); Holmstrom, 1998(b); Cottrill, 1997) and collaborative planning, forecasting and replenishment (CPFR). All are applied to reduce costs and create value-added service to customers.

As is well known nowadays, the industry is operating in an environment that is characterized by intense global competition. Most SMEs are still using the traditional push approach to manage business processes. They process supply chain operations in accordance with their organizational culture, policies and systems. The common problems faced are demand uncertainty, lack of information sharing and invisibility of customer needs and supplier capabilities. In 1990s, new supply chain strategies such as VMI and CPFR emerged from the market. Large enterprises such as Wal-Mart, Campbell Soup and Proctor & Gamble implemented the VMI in the grocery industry (Cottrill, 1997; Cetinkaya & Lee, 2000; Dong & Xu, 2002; Moinzadeh, 2002; Mishra & Raghunathan, 2004). In 1995, Wal-Mart, SAP and Warner-Lambert first adopted the CPFR into their business (Barratt & Oliveira, 2001). These strategies aim to improve business processes in a supply chain, increase efficiency of product flows and provide fast response to customer demands. An integration of applications is one of critical problems encountered by a supplier and its downstream customers. Compared with the CPFR, the scale of VMI is relatively smaller and the resources involved in the processes are fewer. It is suitable for SMEs that wish to change from the traditional approach and take the first step towards a new supply chain mode.

1.2 Scope of Study

Successful implementation of the VMI strategy will lead to the provision of fast response to customers, high product availability and a high level of service. This implies that short lead time, high product fill rate and low stock-out rate will be adhered to by supply chain parties (Chopra & Meindl, 2004). In the present study, the study focuses on how to change a supply chain strategy from the traditional push approach to the VMI strategy, and shows what effects this has on SMEs. There is a need for a simulation tool to mimic processes, including negotiations for an agreement, product replenishment and fulfillment of customer requirements. The outcomes of inventory level, fill rate and stock-out rate are used as indicators for evaluating the feasibility of the strategy.

A close relationship between supply chain strategies and knowledge management is discussed in the study. In this, knowledge capture, adaptation and reuse are embedded in the simulation of supply chain processes. The knowledge for the adoption of an appropriate supply chain strategy is important to supply chain parties if they wish to achieve a win-win situation with their partners and extend the network to a global supply chain. Integrating the concepts of the VMI strategy, knowledge management and simulation is an essential step for solving problems defined in the study. The study aims to simulate operations from a strategic stage to an operational stage, to capture knowledge in such processes and reuse it for solving problems in a real situation. In response to these needs, a knowledge-based simulated platform (KBSP) is proposed in the present study.

1.3 Problem Definition

At the stage of redesigning the supply chain, managerial staff have to possess sufficient knowledge about a new supply chain strategy. However, few managers can predict problems which happen at an operational stage. Integration of applications among supply chain parties is one of problems they have to face. In current practice, there is a lack of guidance on the right way for achieving the goals. Other problems encountered by the parties are:

(i) Both a supplier and a retailer make a large effort at a preliminary stage. It is necessary to wait a long time before successful outcomes are obtained from a new supply chain strategy.

(ii) They would take a high risk if they adopt a new supply chain strategy, if no guidance is provided.

(iii) Knowledge about a change of strategy cannot be acquired, shared, diffused and retained within an organization for further reuse.

To address the problems involved in changing or adopting a supply chain strategy, the KBSP is proposed. This will mimic the supply chain processes and predict the risks, such as demand uncertainty and product stock-out.

1.4 Objectives and Significance of Study

The aim of the research study is to develop a simulation platform for providing guidance and suggestions to supply chain parties who would like to change from the push strategy to the VMI strategy. The objectives of this research are:

(i) To allow suppliers and retailers to negotiate terms of reference in a VMI agreement at the redesigning stage;

(ii) To evaluate the feasibility of the implementation of the VMI;

(iii) To leverage knowledge on managing the VMI processes;

(iv) To establish a prototype of the KBSP for the optimization of a supply chain strategy;

(v) To validate the performance of the KBSP through the trial implementation in a selected reference site.

The research study leads to the establishment of the KBSP for the optimization of a successful supply chain strategy. The significance of the study is:

(i) To identify unexpected problems which will occur during the operational stage;

(ii) To minimize the impact of uncertainty in supply chain operations;

(iii) To provide suggestions for SMEs to establish an appropriate supply chain strategy in accordance with their existing practices;

(iv) To leverage knowledge among supply chain parties that helps them to gain a competitive advantage in the global market.

1.5 Organization of the Thesis

The thesis is divided into seven chapters. Chapter 1 contains an introduction, problem definition, objectives and significance of the research study. Chapter 2 is a literature review, which describes an overview of the supply chain management, its strategies, artificial intelligence and knowledge management. In Chapter 3, the methodology used for development of the KBSP and its design for supply chain integration is proposed. The implementation of the KBSP is demonstrated in Chapter 4. In Chapter 4, two main applications, simulation of the VMI and knowledge retention in the KBSP, are described. The results of trial implementation and performance evaluation of the KBSP at a selected reference site are discussed in Chapter 5. Chapter 6 contains the conclusion and a summary of the achievements of the research. Some suggestions for future research are given in Chapter 7.

Chapter 2 Literature Review

In chapter 1, problem definition, objectives and significance of the study are described. In this chapter, the evolution of supply chain management and a number of supply chain strategies, such as Just-in-Time (JIT), Push-Pull strategy, VMI and CPFR, are reviewed.

2.1 Evolution of Supply Chain Management

Starting from the 1970s, some manufacturing technologies and strategies such as JIT, lean manufacturing and total quality management were developed and applied to production (Mehrabi et al., 2000). These strategies were used in an attempt to reduce lead time, manufacturing costs and inventory level in order to increase profits and the market share. The evolution of SCM is shown in Figure 2.1.

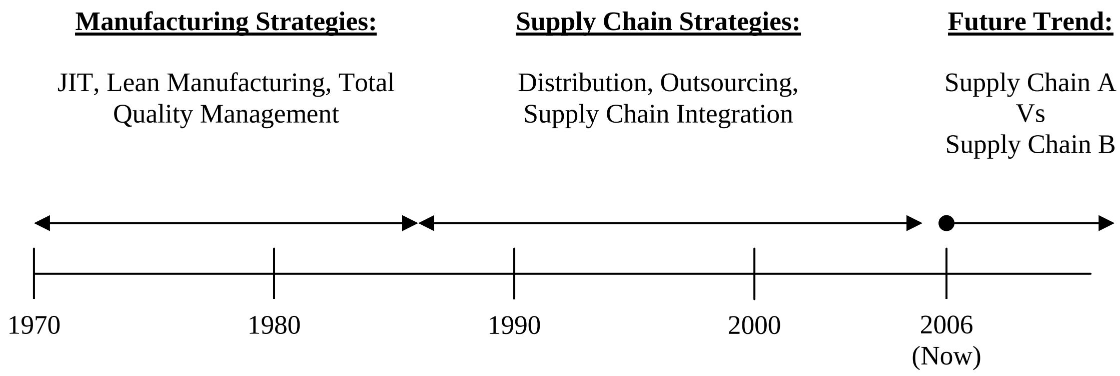


Figure 2.1 - Evolution of supply chain management

In the early 1980’s, “supply chain management” was defined as the integration of business processes, in which products, services and information are provided from suppliers to the end customers (Lambert et al., 1998). The definition of logistics was modified, as a part of supply chain management, by the Council of

Logistics Management (Lambert et al., 1998). This was the first time it was declared that there were differences between SCM and logistics. SCM is also defined as a set of managerial approaches to integrate suppliers, manufacturers, distributors, wholesalers, retailers and customers for production, distribution and product delivery in the right quantities, to the right locations and at the right time (Simchi-Levi & Kaminsky, 2003). Gunasekaran et al. (2004) think that SCM is a major competitive strategy to enhance the productivity and profitability of organizations. Even though SCM integrates supply chain operations and streamlines product flows, demand uncertainty is still a challenging issue which is frequently faced by the supply chain parties (Jung et al., 2004; Hua et al., 2006; Desai et al., 2007; Kwon et al., 2007; Hsieh & Wu, 2008). For instance, a retailer needs to keep a high level of inventory for minimizing the risk of loss due to being out of stock, caused by a large variation of demand. This results in a shift and amplification of variation in demand from a retailer, a wholesaler, a distributor, a manufacturer to a supplier. This is called the bullwhip effect (Lee et al, 1997; Geary et al., 2006) as shown in Figure 2.2. Supply chain integration is one of the strategic approaches, developed by the supply chain parties in the 1990s, to minimize the bullwhip effect,

Also in the 1990s, SCM was considered as the next stage in business management. The strategies were changed from manufacturing strategies to supply chain strategies, including pricing strategy, distribution strategy, outsourcing and procurement strategy, as well as supply chain integration and strategic partnerships (Simchi-Levi et al., 2003). Information sharing is one of initiatives that leads to successful supply chain integration (Kotzab et al., 2003). It facilitates the processes of collaboration among supply chain parties. Some information technologies such as bar coding and radio frequency identification (RFID) are used to enhance the accuracy of data transmission. The technologies provide up-to-date product information at the point of sales (POS) in a supply chain. Adoption of information technologies is able to reduce the bullwhip effect and satisfy the needs of customers, which is an essential factor to drive SCM to success (Preis, 2003; Stank et al., 2001). It shows that successful SCM relies on both information technology and managerial practices.

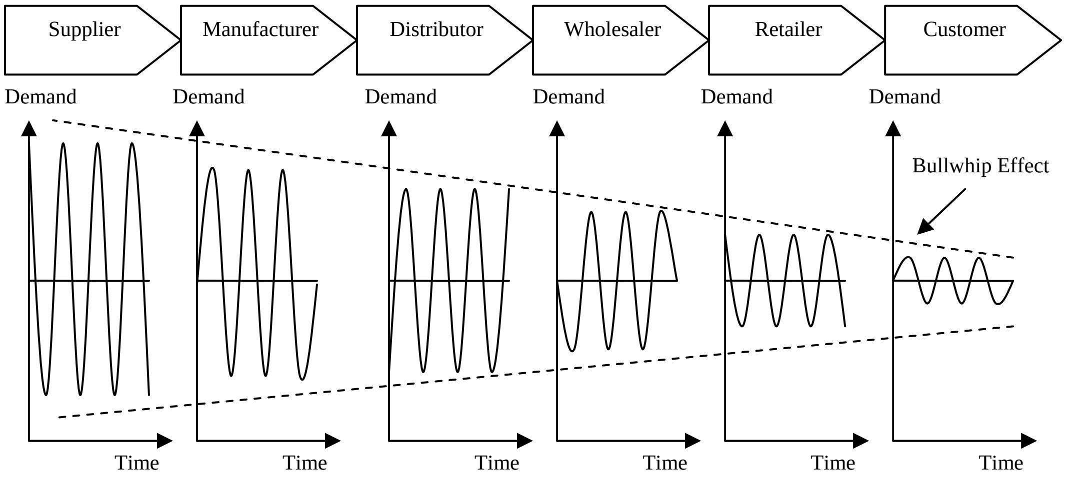


Figure 2.2 - The bullwhip effect in a supply chain

At present, customers require high product quality at low prices, and fast product delivery. VMI is one of the supply chain strategies for achievement of low inventory levels, low stock-out rate and high product availability. It facilitates product flows from suppliers, manufacturers, distributors, wholesalers to a retailer at a high rate with low costs. The strategy not only improves the processes in demand management, but also builds good long-term relationships with partners. Thus, it is a trend for SMEs to adopt such a strategy for improvement of supply chain operations, and enhancement of competitiveness as well. The future trend of SCM is that upstream suppliers and downstream customers will be partners who help each other in increasing the competitiveness in order to compete with other whole supply chains, with success (Cares, 2006).

2.2 Overview of Supply Chain Strategies

In the present study, four supply chain strategies have been reviewed. They are the JIT strategy, the push-pull strategy, the VMI and the CPFR. JIT was invented by Toyota in the 1970s to achieve a target of zero inventory level in manufacturing. In turn, the push-pull strategy was developed in industry, that combined the concepts of the push and pull strategies.

In the 1990s, there was a change of supply chain strategy from an operational level to a tactical level. Efficient consumer response (ECR) is one of the supply chain strategies and is advocated by fourteen trading associations, including the Grocery Manufacturers of America and the Food Marketing Institute, in 1992 (Barratt & Oliveira, 2001). The ECR was first introduced at the Food Marketing Institute Conference in 1993 (Harris et al., 1999). In the same year, ECR was established and published by Kurt Salmon Associates in Europe. It consists of four initiatives which are efficient store assortment, efficient promotion, efficient product introduction and efficient product replenishment. (Harris et al., 1999; Svensson, 2002; Barratte & Oliveira, 2001). The VMI strategy is one of initiatives in the category of efficient product replenishment. It advocates a concept of inventory management that shifts inventory control from the customers to the supplier. Under an agreement using the VMI strategy, the supplier is responsible for providing forecasts and managing the inventory for its customers, and results in reduction of lead time and inventory level.

CPFR is the next generation of the VMI strategy that improves the VMI processes from uni-directional visibility of information sharing to mutually collaborative operations. It is defined by the Voluntary Interindustry Commerce Standards (VICS) Association in 1998, and improves production planning and scheduling, sales forecasting and product replenishment.

2.2.1 Push-Pull Strategy

A push-based supply chain strategy is a long-term forecast for product manufacturing (Donk, 2001; Simchi-Levi et al., 2003). A certain level of inventory is kept to prevent the products from being out of stock. Such a strategy is a suitable approach to forecast products for which there is a stable demand. However, one of the accepted principles of forecasting is that “forecasts are usually wrong and errors are inevitable” (Arnold & Chapman, 2004). In accordance with such a principle, manufacturers estimate the forecasting errors which are likely to occur and produce more or fewer goods to fulfill actual demands in real situations.

In contrast, a pull-based strategy is a demand driven approach in which manufacturing processes are based on actual demands rather than on forecasted ones (Donk, 2001; Simchi-Levi et al., 2003). Neither forecasting nor inventory holding is needed. Thus, the pure pull-based strategy is more likely to be used for controlling dynamic demands. The JIT system is a well-known system which works in a similar way (Kim, 1985), the details of which will be discussed later. It is beneficial for supply chain parties to reduce the inventory level, lead time and reduce effects of variability in demand.

According to Olhager & Ostlund (1990), the working principles of the push and pull strategies in manufacturing processes are called make-to-stock (MTS) and make-to-order (MTO), respectively. The concept of the customer order decoupling point, introduced by Hoekstra and Romme in 1992, was applied for decision-making on a selective problem of MTS and MTO. They claimed that a strategy of MTO can be applied to manufacture products which have a large degree of product customization and need low inventory holding costs (Soman et al., 2004). Li (1992) stated that holding inventory, in the MTS, is one of the ways to compensate for the problem of long production lead time. This strategy helps with a reduction in customer waiting time as a result of fast customer response and an increase in sales.

As for products with high demand rate (Rajagopalan, 2002; Soman et al, 2004), long production lead time (Li, 1992), product life cycle (Soman et al., 2004), setup time and processing time (Rajagopalan, 2002), the MTS strategy is more appropriate for adoption in a company. Consequently, the push strategy is selected rather than the pull strategy.

A push-pull strategy, a relatively modern strategy, combines the push-based and pull-based strategies and is applied in processes from production and assembly to finished products (Simchi-Levi et al., 2003). For generic parts, the push-based strategy is used. The demand is predicted and a certain level of stock is kept. Products can be customized, produced and delivered according to the requirements of customers (i.e. pull-based strategy).

Postponement is one of the supply chain strategies that is applied for such a pull strategy. It is the ability to delay product differentiation or customization until close to the time that the products would be sold (Chopra & Meindl, 2004; Fung et al., 2005). Dell is a computer manufacturing company which has successfully implemented this strategy. As technology is changing rapidly, new models of computer parts are pushed onto the market in a short period of time. Thus, problems related to the short lifespan of parts, the great variety of parts and a high inventory level may be encountered. Once new products are developed and launched in the market, out-of-date products fade out very quickly. The targets, set by Dell, are to reduce losses in fade-out products and achieve good satisfaction in customer requirements.

Dell uses a postponement strategy to manage its business and keeps a low inventory level of generic parts. When customers make an order and customize their needs through the internet, Dell follows this strategy to assemble components and, in turn, delivers finished products to its customer with a short lead time. An excellent relationship is built between Dell and its suppliers that facilities effective product replenishment, high customer expectations, short lead time and low inventory costs.

2.2.2 Just-in-time (JIT)

Supply chain strategies are divided into two levels which are the operational level and the tactical level. As mentioned in the previous section, a whole supply chain is operated by a supplier, a manufacturer, a distributor, a wholesaler, a retailer and customers. JIT is one of the operational strategies to support a successful supply chain through reduction of total cost and an efficient supply (Morash, 2001). Supply chain integration is a consideration of JIT (Frohlich & Westbrook, 2001).

The process of JIT is driven by the concept of Kanban, which makes use of cards to point out which item is stock-out in a production line and needs to be replenished. Once an inventory level of an item drops to zero, a kanban card, including information on the items, is removed and sent to the warehouse. The required quantities of the item and a removable kanban card are delivered to the production line. A supplier needs to replenish items which have been taken from the warehouse. As the amount of inventory is limited in each work station, the replenishment process is stopped when enough components have refilled the line. In Canada, a medium-sized manufacturing company, Waterville TG Inc., has successfully applied the JIT strategy to reduce their inventory level by more than two thirds and in-transit lead time by a factor of seven (Landry et al., 1997). The JIT strategy allows no storage in the production area in order to reduce holding costs and to improve the efficiency of the manufacturing process.

In practice, JIT has a positive effect on production processes through a set of improvement targets. In fact, it is not only limited to manufacturing practice, but is critical to an organization (Sakakibara et al., 1997). The success of JIT relies heavily on on-time delivery by suppliers. To achieve this goal, the responsibility is aligned with suppliers and more inventory is held by them (Grout, 1996). It shows that the inventory holding is shifted from the warehouses of customers to those of suppliers. VMI is one of the most recent supply chain strategies which is becoming popular in supply chain practice. This approach is applied in the food industry, for instance, Campbell's Soup and Barilla (Waller et al., 1999). In VMI, a vendor takes responsibility for product replenishment on time. This is discussed in the next section.

2.2.3 Vendor Managed Inventory

Vendor managed inventory, also known as the continuous replenishment program (CRP), is one of the initiatives of ECR (Andraski, 1994; Cachon & Fisher, 1997; Waller et al., 1999; Toni & Zamolo, 2005). ECR is the next generation of quick response (QR) (Harris et al., 1999) which originated in the textile industry in 1984 (Lowson et al., 1999).

VMI is one of the most effective initiatives for improving the supply chain efficiency of various parties (Waller et al., 1999). It is a process of product replenishment in which suppliers manage inventory for customers. Under a VMI agreement, customers share sales orders and inventory records with their supplier. This assists the supplier to provide accurate forecasts, good production plans and on-time product deliveries. The visibility of information from the supplier to customers, leads to reduction of the bullwhip effect, due to demand uncertainty.

According to Cachon & Fisher (1997), a manufacturer in the grocery industry, Campbell’s Soup, implemented a trial run of the VMI strategy in 1993. It delivered products consisting of several hundred stock keeping units (SKUs) from a number of warehouses to four distribution centers. Each distribution center sent sales demands and inventory to Campbell’s Soup via EDI every weekday. The manufacturer used the information sent by the distribution centers to determine the minimum inventory level of each product everyday. If the product inventory dropped below a certain level, the product was replenished up to at least the minimum inventory level, by full truckloads. This reduces the inventory level and successfully improves the service level.

VMI has been successfully adopted in the grocery sector in Europe, in which a wholesaler sends information such as the reorder point and the minimum batch size to a vendor for managing inventory (Holmstrom, 1998a; Holmstrom, 1998b). In the U.S., a manufacturer, Panduit, applied VMI to cooperate with its distributors for effectively improving service levels to the end customers (Cottrill, 1997). Vergin & Barr (1999) stated that VMI is beneficial for reducing inventory and stock-out rate. This is proven by ten manufacturers with an average of three-year’s VMI experience in the grocery industry. Successful implementation of VMI helps in achieving fast customer response, reducing the inventory level and enhancing the service level. It indicates that the reorder point, the batch size, the service level and the lead time are critical factors in a VMI agreement. In the present study, they are used as parameters for the KBSP to simulate the VMI strategy, which will be described in Chapter 3.

A process of VMI is that retailers send information to a supplier via EDI for managing the inventories of the retailers. The supplier keeps more inventories for the achievement of quick customer response and high service level. For a trade-off between loss and profit, the supplier usually implements VMI to more than one retailer for increasing cost-effectiveness. For products with high demand and low supply, a big challenge faced by a supplier is to fulfill the requirements for each VMI retailer. The supplier determines how to allocate products to them based on relationship and customer closeness. The retailers do not know what decision the supplier will make and how many products will be allocated. It reflects that the invisibility of information from the supplier to the retailers is one of the limitations of VMI (Barratt & Oliveira, 2001).

Moreover, the protocols used for communication in EDI must be the same. For two parties using EDI-based systems with different protocols, the information cannot be exchanged in VMI processes (Cottrill, 1997). In practice, large enterprises do not intend to modify their existing EDI business practices to conform to other partners, which operate their business on a small scale. The partners, which intend to collaborate with the enterprises under the VMI strategy, need to change their existing protocols to the enterprise’s one. The process of changing existing practices is complicated and its outcomes are unpredictable. There is a need for the simulation of VMI and to predict the possible outcomes. Due to these limitations, the CPFR emerged from the grocery industry at the end of the 1990s.

2.2.4 Collaborative Planning, Forecasting and Replenishment

The concept of collaborative planning, forecasting and replenishment was originally generated due to a need for collaborative forecasting and planning in the grocery industry (Holmstrom et al., 2002). CPFR is the first guideline published by the Voluntary Inter-industry Commerce Standards (VICS) Association in 1998. The CPFR strategy represents the next step of the VMI approach in the development of efficient collaborative supply chain networks. According to Seifert (2003), the nine steps of implementation of the CPFR strategy are to:

(i) Develop a front-end agreement;

(ii) Create a joint business plan;

(iii) Create a sales forecast;

(iv) Identify exceptions for the sales forecast;

(v) Resolve/collaborate on exception items;

(vi) Create the order forecast;

(vii) Identify exceptions in the order forecast;

(viii) Resolve/collaborate on exception items;

(ix) Generate the order.

The CPFR is beneficial for enhancement of the relationship between a supplier and a retailer, improvement on the accuracy of the order forecast and reduction in inventories (VICS, 1999). However, there are some limitations for implementation of the CPFR. As stated by Barratt & Oliveira (2001), results from the literature review and the survey show that one of barriers of the CPFR processes is in identifying exceptions for the sales forecast which are not easily managed. The trading partners focus on traditional supply chain operations rather than handling issues of exceptional cases. It found that sufficient information technology supporting to the various parties is an essential requirement for providing an accurate forecast. Thus, it would be inappropriate for SMEs to implement the CPFR if they do not have modern technology to support the processes.

2.3 Knowledge Management in SCM

Knowledge, experience and working skills are assets an organization intends to accumulate and retain for the generation of new knowledge, new ideas or for formulating a new strategy for increasing competitive advantage. The implicit knowledge is learnt by staff when they are working in a supply chain. It is transferred to explicit knowledge such as reports and guidelines. Systematic retrieval and retention of the knowledge help an organization to manage supply chain operations in a timely manner. A long-term supply chain strategy is decided for fast response to customer needs, high product quality and on-time product delivery. It provides guidelines for the subsequent step of production planning and scheduling. At the design stage, there is a need for a huge amount of knowledge in order to make decisions on the selection of supply chain strategies and agreement terms. It implies that there is a strong relationship between knowledge management (KM) and supply chain strategy.

2.3.1 Overview of Knowledge Management

Knowledge is a kind of experience, truth, judgment and guidelines that is transformed from data and information (Davenport & Prusak, 1998). It acts as an asset in an organization. Nonaka & Takeuchi (1995) stated that knowledge is captured from outside and is shared within an organization for generating new products, services and systems. Knowledge creation lets an organization adopt an innovative approach and gain competitive advantages.

Wiig (1997) stated that one of the KM practices that was started at Chaparral Steel in 1975 involving a knowledge-based system was developed for managing knowledge in commercial fields in the early 1980s. Knowledge creation, capture, organization, renewal and sharing exist in every stage of business planning and operation. Knowledge creation is a way to generate new knowledge through organizational learning, research and development, and motivation of employees to innovate and learn from experience. KM is an effective way for enterprises to create knowledge and renew their knowledge frequently so that both the organization and employees gain benefits.

Davenport & Prusak (1998) described KM is a process to generate, codify and transfer knowledge from a human being to an organization and vice versa. Organizations hire knowledgeable workers for generating new ideas on product development, improving services or sharing knowledge with other colleagues. They get advice from a consultant firm or purchase a potentially useful company for adapting outside knowledge to its existing practices. An effective way for sharing knowledge is face-to-face meetings, informal conversations, videoconferencing and presentations. Technology helps knowledge codification, from tacit knowledge to explicit knowledge, and knowledge transfer in an organization. For instance, Microsoft created a knowledge map that guides new employees in finding solutions to their working problems. Lotus Notes, developed by IBM, is a tool for KM. Knowledge created in a discussion or a meeting is retained and diffused through the web. In an organization, employees use the software to seek knowledge they need in real time. In the view of Wiig (1997) and Davenport & Prusk (1998), KM is supported by technology and it is effective for strategic planning and operations. Some expert systems and artificial intelligence (AI) methods and tools such as case-based reasoning (CBR) and neutral networks provide a channel for accessing knowledge effectively.

KM processes consists of knowledge capture, storage and deployment (Preece et al., 2001). A process of knowledge engineering, proposed by Preece et al., includes five steps: (1) defining scope of a knowledge-based system; (2) building a conceptual model; (3) constructing a knowledge-based system; (4) operating it and verifying its performance and (5) refining and providing maintenance on it. The knowledge-based system (KBS), is based on the knowledge engineering process for design and development, for which KBS is able to deal with the challenge of knowledge integration, which is faced by some knowledge tools, such as document management systems, discussion forums, capability management systems and lesson-learning knowledge base systems. It facilitates knowledge capture, knowledge retention and knowledge integration.

2.3.2 The Importance of KM in SCM

Knowledge is stored in the mind of human beings and is not easy to extract. Davenport et al. (1998) studied thirty-one KM projects in twenty-four companies, which include Hewlett-Packard (HP), automobile manufacturers and high-tech manufacturers. A successful KM project implemented in HP, called “Electronic Sales Partner”, provides technical product information, sales presentation and sales and marketing tactics. It is a typical example which shows that KM is useful for enterprises in the manufacturing industry. According to Davenport et al. (1998), it is claimed that a problem encountered in expert systems in the 1980s is that knowledge cannot be managed and structured by using a set of rules. As a result, KBS evolved in KM.

Knowledge management software will increase its share of the worldwide software market (Goldenberg, 2002). KM tools are widely used for fast response to customer requests and improvement of the quality of customer services. Some companies, such as Nike and Cisco Systems, which are implementing those tools in their e-service aspects of SCM, allow customers to search for up-to-date product information or to find answers to their inquiries. This shows that information technology in KM plays an important role within organizations for managing supply chain operations.

A literature survey over the last ten years was conducted to study design, development and implementation of a KM system in manufacturing (Gunasekaran & Ngai, 2007). It shows that an E-commerce enabled supply chain for business-to-business (B2B) is one of the future research directions of KM in 21st century manufacturing. There is a need for the development of the KBSP for managing activities in B2B. Figure 2.3 shows the relationships between KM and supply chain strategies.

A top-down approach to supply chain management is that KM processes start from supply chain strategy, are used in planning, and continue throughout all operations. Through the supply chain operations, staff create working guidelines, documents and working skills. The knowledge is transferred and adapted by them. It is integrated with external information for generating new knowledge that is applied to supply chain strategies.

As a result, knowledge exists in each stage of a supply chain. Knowledge capture, share and retention are important for the success of the supply chain. A web-based supply chain for B2B in KM is a future trend in supply chain management.

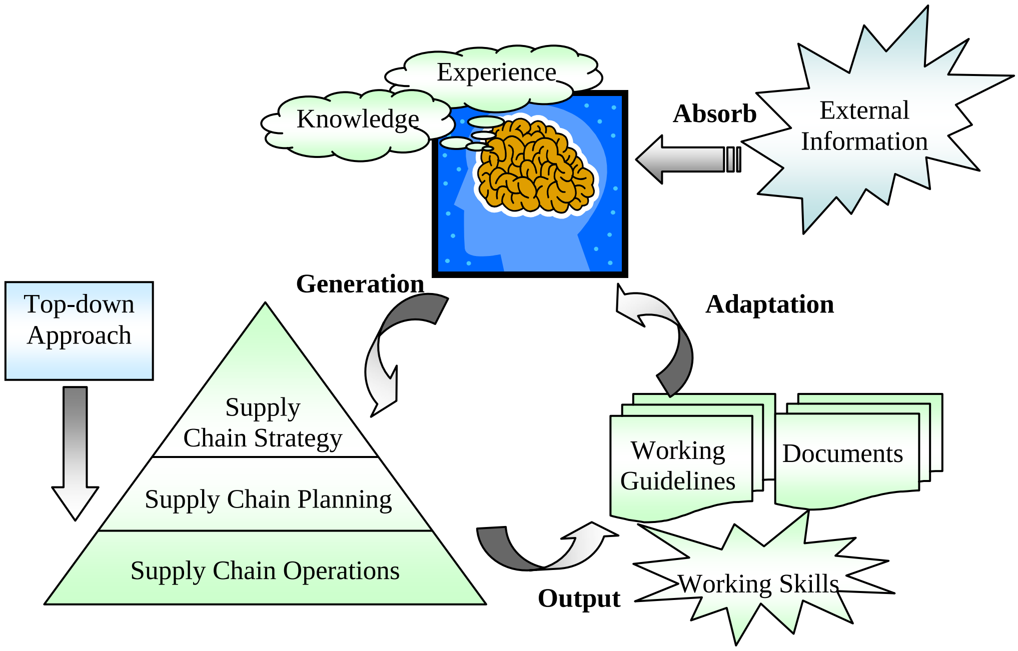


Figure 2.3 - Relationship between supply chain strategies and knowledge management

2.4 Types of Simulation and Applications

In the past decade, simulation models or tools emphasize simulating and streamlining complicated supply chain processes. Most of them focus on the following areas: inventory optimization (Daniel & Rajendran, 2005), problems of the bullwhip effect (Chatfield et al., 2004; Holland & Sodhi, 2004), supplier selection (Ding et al., 2005) or decision support for the supply chain (van der Zee & van der Vorst, 2005). These simulation tools are generally presented in several ways: spreadsheet simulation, system dynamics, discrete-event simulation and business games (Kleijnen, 2005).

Persson & Olhager (2002) presented a discrete-event simulation to evaluate relationships between quality, lead time and costs. Anderson & Morrice (2000) developed a simulation game for illustrating the impact of demand fluctuation. However, it is not practical for various supply chain parties to interactively play with each other. In a rapidly changing business environment, a simulation model which operates via the Internet is needed, so real time information can be shared and acquired.

2.4.1 Beer Game

The concept of the beer game was first introduced in the 1960s (Hieber, R. & Hartel, I., 2003), and simulated the effect of supply chain operations. However, the traditional beer game appears to be over simplified in that the design situation is focused mainly on handling issues of inventory and back-orders. Once the players completed the game, knowledge created in the game playing processes is seldom retained for reuse. The results usually reflect what the players have achieved, rather than how they have performed. It is difficult for the players to learn knowledge about current practices and strategies from the game. The other major shortcoming is that the beer game involves only a single party in each field, that is one supplier, one manufacturer, one retailer and a single customer. In practice, to have no competitor in a whole supply chain is impossible. Thus, the beer game may not reflect the real situation in today's business environment. However, it is useful as an educational game which introduces the basic concepts of supply chain operations.

Simchi-Levi (2003) proposed a beer game to educate students in the concepts of supply chain management, such as lead time reduction, global information sharing and centralized management. Players can choose to be one of the supply chain parties, a retailer, a wholesaler, a distributor or a manufacturer at the beginning of the game. During the game, the player is responsible for placing orders, and looks at the impact on inventory and lead time. Figure 2.4 shows a typical example of a traditional game.

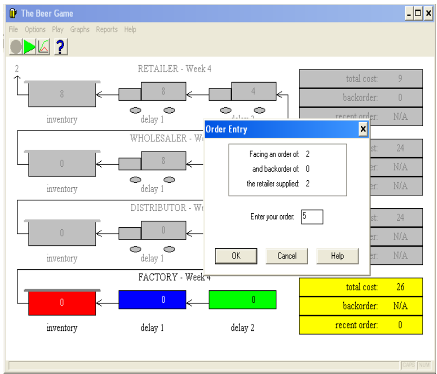


Figure 2.4 - Screenshots of the traditional beer game

2.4.2 Social Simulation Model: Sugarscape

Epstein & Axtell (1996) proposed a simulation model to mimic the behavior of humans, who are represented by agents, located on a landscape of renewable and generalized resources (e.g. sugar). The agents use their individual ability (vision) and needs (metabolism) to perceive, move around in and transform the local environment. The target is to trace and find sugar that is distributed in various places (Terna, 2001). Each agent searches its neighborhood to learn where sugar is the most plentiful and moves it to the destination. Thus, the behaviour of humans is simulated and predicted. The simulation of the sugarscape is shown as Figure 2.5.

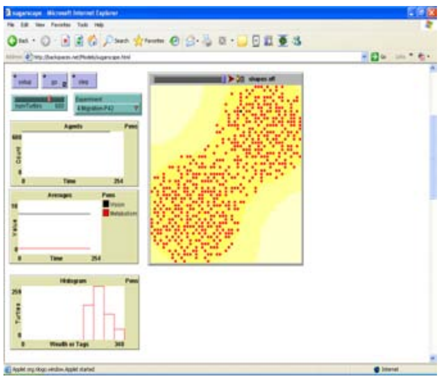


Figure 2.5 - Screenshot of Sugarscape

2.4.3 Knowledge-based Simulation System

A knowledge-based simulation system is defined as a combination of knowledge processing methods and a simulation during which the problems are solved (Kopacsi & Kovacs, 1993). Erraguntla et al. (1994) proposed a knowledge-based simulation for gathering information, answering questions and solving the problems of users. Another type of knowledge-based simulation systems is implemented in online mode. It detects problems and searches for the solutions in a knowledge base first. If the problems cannot be fixed, alternative action is taken in the online simulation (Gonzales et al., 1996).

In the 1980s, a knowledge-based simulation system was proposed to simulate production planning and the utilization of machines (Fox et al., 1988). The goal of the simulation was to minimize inventory. In the simulation, the parameters, which were weighted, were order fill rate, inventory turns and inventory investment. In the simulation process, a set of rules were set up and recommendations were provided at the end.

Most applications of the knowledge-based simulation systems are to provide solutions on issues of production scheduling, planning and utilization of machines. Knowledge captured in the simulation system is related to problems of users. For instance, how many products will be made. However, those systems cannot provide an interactive platform for different participants to join and share knowledge immediately. It is proposed in this thesis that a simulation model for processes of product delivery and product replenishment is built on the interactive platform that provides a chance for the participants to learn from each other.

The systems described above usually simulate internal operations such as product stock-in and stock-out, production planning and scheduling. However, cooperation among supply chain parties cannot be simulated. There is a need for a multi-player simulation in supply chain operations. In this study, the KBSP provides a web-based environment for multiple players to simulate supply chain operations under a VMI agreement.

2.4.4 Simulation of Enterprise Resource Planning (ERP) Systems

In the simulated environment of an enterprise, workflows, daily operations and existing problems of each department, such as human resources, sales and marketing, purchase, finance and logistics, are considered. Srinivasan & Jayaraman (1997) proposed a simulation model with SIMAN, which is a simulation language, to plan assembly processes in an enterprise. The trend is to integrate simulation as embedded components in enterprise applications that can be distributed via the Internet (Harrell & Hicks, 1998). Supply chain planning is also a common topic in the simulation world that focuses on planning, such as demand forecasting, order processing, production planning and distribution planning. Commercial simulation tools have been developed to handle such kinds of problems, including Supply Chain Analyzer by IBM (Archibald et al., 1999), a business process simulator called Virtual Suppliers Manager (Umeda & Jones, 1998) and LOGSIM simulator (Hieta, 1998).

2.4.5 Applications in an Enterprise

In manufacturing plants, many manufacturing tools such as Computer Aided Design (CAD), Computer Aided Manufacturing (CAM) and simulators are used to design products and simulate production processes. The goal of simulation for ERP is to reduce process errors, shorten lead time, increase utilization of resources and improve the manufacturing performance.

Robot Based Agent Oriented Planning and Simulation System (ROPS) is one of simulation tools that enables users to analyse the dynamic behavior of a model and thus make decisions with ERP applications (Kubota et al., 1999). SIMFLEX is a famous simulation tool that works with the Microsoft Windows operating systems to mimic a simulation and analyse the results of the simulation. It provides a strategic plan for a whole supply chain, including manufacturing, transportation, procurement, distribution and sales (Williams, 2003). Besides, Supply Chain Builder is a discrete-event simulation tool to control and simulate a network of production plants and distribution centers. In such a simulation, the materials, inventories and shipments can be customized.

The limitation of traditional simulation tools such as the beer game is that it allows one player to play in simulating a simple process. This reflects an unrealistic situation. In one of the examples of simulation tools, namely Sugarscape, the behaviour of humans is predicted through simulated processes. However, supply chain parties play with the systems in which thousands of rules are pre-defined and fixed. There is a lack of simulation tools to simulate the impact of changes in a supply chain strategy. Once it has changed from the push system to the VMI, an organization would not be able to predict how much effort they need to put in and what results they would obtain. Thus, the KBSP is different from existing simulation tools in that the effects of changing the supply chain strategy are predicted. It provides guidelines for organizations as to what problems would be encountered, how to solve them and what techniques should be learnt to reduce the risk. The effort and costs for designing the strategy are thus minimized.

Table 2.1 summarizes five simulation tools, which are Beer Game, SIMFLEX, Supply Chain Analyzer, Virtual Supplier Manager and LOGSIM. These are compared with the KBSP in five aspects: (1) goals, (2) area of application, (3) number of parties, (4) online simulation and (5) knowledge generation and retention, respectively.

Table 2.1 - A comparison between five simulation tools and the KBSP

	Beer Game (Simchi-Levi, 2003)	**SIMFLEX (Williams, 2003)**	**Supply Chain Analyzer by IBM (Archibald et al., 1999)**	**Virtual Suppliers Manager (Umeda and Jones, 1998)**	**LOGSIM simulator (Hieta, 1998)**	**KBSP**
Goals	To look at the impact of decisions on inventory and lead time	To analysis inventory control policies (MRP & MTO) and logistics services and delivery policies	To quantify the effects of making changes throughout supply chain	To find what impact on total lead time and total productivity.	To analyse a level of inventory and customer service performance.	To reduce inventory level, rate of stock-out and improve fill rate
Area of Applications	Education: Concepts of Lead Time Reduction, Global Information Sharing and Centralized Management	Supply chain operations: production policy, inventory control and delivery policies	Planning and Scheduling, supplier performance management & customer and supplier collaboration capabilities	Planning problems and push-pull strategy	Tele-communications equipment manufacturing	To re-design supply chain strategies from pull approach to VMI approach
Number of parties	One player (retailer, wholesaler, distributor, or manufacturer)	Multiple users	Multiple users	One user	One user	One supplier and multiple retailers
Online Simulation	NA	Yes	Yes	NA	Yes	Yes
Knowledge Generation and Retention	During a process, players learn from the game, but the knowledge cannot be shared and retained.	Information of analysis for policies can be generated. It cannot retain knowledge.	It is for making decision on design and operation and knowledge from user message can be retained.	“Supplier Management Knowledge Data Base” is used to analyse suppliers operations.	NA	Knowledge generated from a VMI process can be retained.

For the aspect of "area of production", these are applied to production, planning and scheduling, supply chain planning and supply chain analysis. Compared with the other tools, the KBSP allows the supply chain parties to change a supply chain strategy from a push strategy to the VMI. Most of the tools are powerful simulation tools for supply chain management. However, the KBSP is valuable for all supply chain parties who are considering using the VMI strategy to predict outcomes at the preliminary stage.

2.5 Expert Systems

An expert system is the computer system in which the knowledge of experts is stored in such a way that users can find solutions from it (Durkin, 1994). It includes an inference engine, a knowledge base and a working memory. A decision for a certain problem is made by asking a number of questions and explaining the reasons for actions (Liao, 2005). In the mid-1960s, expert systems were developed by the artificial intelligence (AI) community and they have been implemented in common applications, such as rule-based expert systems and fuzzy expert systems (Liao, 2005).

An inference engine is operated in the expert systems that derives knowledge by using a set of rules. It matches problems in a working memory with knowledge in a knowledge base (Durkin, 1994). A process of inference determines what knowledge is to be used in the next step, when to ask questions from users and when to get relevant knowledge from a knowledge base. Searching for solutions in the inference engine is based on two methods: forward chaining and backward chaining. Forward chaining is a process for finding solutions from data, a process of inference to a conclusion about problems. In contrast, backward chaining searches for the final nodes first, works backward until a complete path, which is from the results to an initial problem, is traced.

In the airfreight forwarding business, a heuristic approach, which is a technique of optimization, is applied to handle issues of cargo loading (Lau et al., 2004). It is combined with the forward chaining approach, achieving the objectives of optimization of container space and total shipment costs. Wee & Yang (2004) derived a single-producer, multi-distributors and multi-retailers heuristic model for optimization of inventory cost.

Xie et al. (2006) proposed a two-level supply chain coordination algorithm, which combines a heuristic approach with a fuzzy concept, to optimize inventory holding cost and shortage cost. Chern & Hsieh (2006) developed a heuristic algorithm for multi-objective master planning in supply chain networks. Situations regarding outsourcing, production capacity and inventory management are taken into consideration in order to minimize the inventory holding cost and delay penalties.

Nachiappan & Jawahar (2007) proposed a genetic algorithm with a heuristic approach to optimize the sales price of buyers, and an acceptable contract price between a vendor and buyers, under a VMI strategy. These methodologies reflect that the inference engine is a way to derive optimal solutions in supply chain applications. In the research study, the inference engine is applied for optimization of batch size in a VMI agreement. The objectives of maximization of fill rate and minimization of stock-out rate are achieved.

Knowledge representation is a method to represent human knowledge by using rules, semantic networks and frames (Kandel, 1992). Rules are set by both premises and the conclusion, in which promises involve a sequence of causes that work with connectives, such as operators of “max”, “min”, “and” and “or”. A conclusion derived from the promises is an action to be taken for solving a problem. The semantic network is used to represent knowledge by nodes and arcs. Those nodes represent objects, concepts or situations. Arcs show relations between nodes and the weight of such relations. A frame is object-oriented knowledge representation and allows inheritance of concepts from its superclass or subclass. It is constructed by using a data structure and a set of slots, which include procedures, data and pointers to other frames. These knowledge representations work with the inference engine in an expert system to derive solutions from a problem.

2.5.1 Rule-Based Expert Systems

The rule-based expert system has been used in an electronic business that handled a process of placing orders in which agents represent the supplier and the buyer, and these agents communicate with others in such a process (Ahn et al., 2003). Chen & Gao (2004) proposed that an analogy agent and a DB monitoring agent used in a rule-based system reduce inaccurate concerns and the inference of new knowledge of the ERP implementation processes.

Besides, the rule-based systems applied to delivery scheduling optimize the traveling distance from an original point, or a depot, to a terminal point (Chang & Lee, 2004). In such an optimization model, a set of rules about constraints on delivery time, penalty for tardiness and sequences in routing can be modified to achieve an optimal goal. Kim & Rogers (2005) proposed a framework involving integration of a rule-based system and an object-oriented approach for modeling supply chain operations, from business goals to business processes.

Recently, the IF-THEN rule has been built into agent technology for formulating the virtual environment of a manufacturer, in which the agents are responsible for the processes of negotiation of an order, manufacturing, assembly and transportation (Cutting-Decelle et al., 2007). It reflects that a rule-based system is operated mostly in the operational level of a supply chain. In a competitive business environment, new technology and various supply chain strategies are always being developed. Such rule-based systems lack the ability to adapt when working under various supply chain strategies. Once a new supply chain strategy is applied in a company, the rule-based expert system will have to be re-developed again. Knowledge captured in the rule-based system is not easy to retain. Thus, a rule-based expert system is more likely to be suitable for providing decision support on operational work rather than for making decisions on strategic issues.

2.5.2 Fuzzy Expert Systems

Fuzzy expert systems provide a way to make a decision with a certain degree of accuracy that is different from the “True” or “False” decisions made in a rule-based expert system. Human reasoning is not limited to “True” and “False”, and fuzzy sets are embedded into a rule-based system to form a fuzzy expert system and to simulate human reasoning (Liao, 2005). Mikhailov (2000) stated that a combination of fuzzy logic with an analytic hierarchy process, which was proposed by Saaty, T.L. (Saaty, 1990), is useful for solving prioritization problems. This approach is adapted to make decisions for multiple criteria problems concerning the selection of the maintenance strategy in a power plant (Wang et al., 2007).

In supply chain applications, a fuzzy-based approach is used for supplier selection in which multiple criteria, such as technological capabilities and just-in-time purchasing, are considered (Bevilacqua & Petroni, 2007). Usernik & Bogataj (2005) proposed a fuzzy expert system, which is composed of fuzzy logic and a neural network, for application when making decisions on location and inventory strategies. These applications show that the fuzzy expert system is one of the best approaches for solving problems on multiple criteria optimization.

2.5.3 Knowledge-based Systems

Knowledge-based systems (KBS) contain explicit and implicit knowledge in a computer system that use specific situations for reasoning and adapt relevant knowledge of existing cases (Wiig, 2004). Knowledge in KBS is obtained from information that is used to solve problems and support decision making. It has an impact on each level of organizational knowledge from individual, group, organizational and knowledge links (Dutta, 1997). It is the outcome of the knowledge engineering processes and is considered as a framework for knowledge management (Hendrisks & Vriens, 1999).

KBS is commonly applied in knowledge representation adopted in a hybrid expert system (Mitra & Basu, 1997), decision support in business process re-engineering (O'Keefe & Preece, 1996) and even extending to decision support and learning (Mockler et al., 2000). In strategic management, KBS helps to configure the resource allocations and planning in the organization, for fast response to strategic changes (Volberda & Rutges, 1999). Besides this, the KBS is combined with a business game to produce a simulation environment in which the business processes of the manufacturing industry are simulated and decision-making knowledge is kept within an organization (Duan et al., 1998).

O' Leary (1998) found that KBS is broadly used in several areas such as finance, new business development, knowledge management, supply chain management and order management. The survey shows that the number of entries of supply chain management follows just behind that of the finance category as shown in Table 2.2. This reflects that KBS plays an essential role in supply chain management.

Table 2.2 - Number of entries using knowledge-based systems in practice (O' Leary 1998)

Categories	**Number of entries**
Executive processes	1163
Finance	1488
New business development	430
Knowledge management	222
Order management	702
Production and service delivery	4212
Supply chain management	***1172***
Support and shared service	945

2.5.4 Business Intelligence Systems

Business intelligence (BI) is a business decision-making technique that analyses data collected from daily operations and transforms this data into information for decision marking. The BI system, which is a kind of decision support system which emerged in 1970s and it, in turn, rapidly developed from 1980s to 1990s (Siegel, 2000). An intelligent process comprises the framing of questions, gathering data, analyzing data and disseminating intelligence that was applied for collecting and transforming information into knowledge for making business decisions (Siegel, 2000).

The next era of BI is business performance management (BPM), also known as business process management, which is based on a set of processes for optimization of business performance (Golfarelli et al., 2004). A number of gathering techniques in BI are data mining, text mining, Semantic Web and data warehousing. One of the challenges of BI is gathering high quality information from a vast amount of resources (Marshall et al., 2004). They proposed a business intelligence tool named EBizPort, which uses techniques of the summarizer, the Arizona Noun Phazer and the Self-Organizing Map, for extraction of quality content from the web. A number of BI systems on the market including Business Object SA's Enterprise XI Suite, IDV Solutions, Pentaho open source BI Suite and Dundas Data Visualization Inc. support data mining, reporting, dashboard applications and visualization (Hedgbeth, 2007).

2.6 Artificial Intelligence (AI)

In this section, some artificial intelligence (AI) techniques are reviewed. They are neural network, fuzzy logic, genetic algorithm, CBR and hybrid artificial intelligent systems.

2.6.1 Neural Network

The concept of a neural network originates from signal transfer processes via the elements of the brain, involving neurons (Hagan et al., 2002; Russell & Norvig, 2003). In the brain, neurons are composed of the axon, the cell body as well as dendrites. The contact between the axon and dendrites of another cell is called a synapse. The neural networks appear to be a combination of arrangements of neurons and depend on the strength of the synapses.

The backpropagation algorithm is widely used in learning with multiple-layer feed-forward networks (Riedmiller & Braun, 1993). The neural network has the ability to learn complicated multi-dimensional mappings (Hecht-Nielsen, 1989). During the training processes, weightings are added in attempts to fine-tune the system in order to reduce the error between the actual output and the expected output.

In some complicated cases, many steps are needed if an acceptable solution is to be obtained, and so the learning time is relatively long.

Neural networks are applied mostly to represent complex nonlinear relationships and are good at classification. They are commonly used in pattern recognition and face detection (Carpenter & Grossberg, 1988; Rowley et al., 1998; Osuna et al., 1997; Hsu et al., 2002). Based on images of the face, eyes and mouth, humans are recognized by using a neural network algorithm. However, one problem is the speed of the process. A long time is needed from when the pixels are input until the result is generated.

The algorithm of a neural network is generally used in applications of SCM. Forecasting is frequently discussed in SCM for which risk reduction and accurate prediction of demand are provided in the supply chain. Hill et al. (1996) stated that, by use of the neural network concept, the accuracy of forecasting is effectively increased. However, a limitation of a neural network is that there is some bias in the parameters of the model. In a situation without experts, data transformation would fail when the input variables are not defined clearly or are inaccurate.

2.6.2 Fuzzy Logic

The concept of fuzzy logic comes from classical logic, which defines the result of an event as either true or false. For example, a light is switched either on or off. However, in practice, results of most events cannot be represented by only the TRUE-FALSE system. For example, the temperature of water cannot be classified into “hot” or “cold” only. It is usually measured and described by degrees. Thus, the classical logic, which is the TRUE-FALSE system, is most suitable for use in an extreme case. Based on such a concept, a theory of fuzzy logic has emerged. It describes events that are both partially true and partially false (Chen & Pham, 2001).

Going back to 1965, the original concept of fuzzy set theory was proposed by Lotfi Zadeh. Zedeh (1975) stated that fuzzy logic is the logic of approximate reasoning and includes such features as:

(i) Fuzzy truth-values expressed in linguistic terms, e.g., true, very true, false, not very false and so on;

(ii) Imprecise truth tables;

(iii) Validity of inference rules is approximate rather than exact.

In current practice, the fuzzy logic system is commonly used in industrial applications. In designing a fuzzy logic system in the right way, the functions of the basic components of fuzzy logic must be fully understood. One of its applications adopts a fuzzy logic approach to evaluate the capabilities of potential suppliers in a supply chain (Shore & Venkatachalam, 2003). Fuzzy logic algorithms are commonly used in temperature and pressure control systems such as steam turbines (Cox, 1992) and in pattern recognition (Bezdek, 1981). In 1994, a fuzzy logic advisory tool (FLAT) was developed to forecast demand for 10,000 products and to control material purchase of 14,000 components for Nokia, the telecommunication manufacturer (Frantti & Mahonen, 2001). It shows that fuzzy logic theory is adaptable and can be applied in a variety of situations.

2.6.3 Genetic Algorithm (GA)

The genetic algorithm (GA) provides a method for a search for an optimal solution in optimization problems. When searching for an optimal solution, many possible solutions emerge. GA is an approach to select the optimal one from a solution set. During the processes, inputs are mixed and matched to generate a number of possible solution sets. The best solution is matched with a particular problem. The processes of defining the GA are selection, crossover and mutation (Dhar & Stein, 1997; Bui & Moon, 1996).

In supply chain operations, problems of optimization on some parameters, such as lead time and inventory level, are frequently discussed. An application of the GA is to find the optimal set of such parameters. Besides, it is applied for the selection of facilities in manufacturing plants and distribution centers (Syarif et al., 2002) and optimization of production material logistics in simulation games (Yang & Pang, 2005).

2.6.4 Case-Based Reasoning

Case-based reasoning (CBR) is a kind of artificial intelligence (AI) technique that uses a problem-solving approach for learning experience from past cases and reusing this experience for solving new problems (Aamodt & Plaza, 1994). Its concept comes from a dynamic memory model and a memory organization packet theory (MOP) proposed by Roger Schank (Aamodt & Plaza, 1994; Schank, 1982). The first CBR systems called CYRUS, introduced by Janet Kolodner, was developed by this concept (Kolodner, 1983).

According to Wilke et al. (1997), CBR is a system that solves problems by adapting and reusing relevant knowledge of previous cases. Simoudis (1992) stated that a CBR system consists of two components, which are a case base and a problem solver. In the CBR system, problems of new cases are solved by using solutions of previous problems. The purpose of CBR is to provide suggested solutions to problems and to provide a context for understanding or assessing a situation (Kolodner, 1993).

Concepts of CBR are described in many models, such as Kolodner's model (Kolodner, 1993) and Aamodt and Plaza's model (Aamodt, A. & Plaza, E., 1994). Kolodner considers CBR as a process of "remember and adapt". In Figure 2.6, the CBR cycle of Kolodner's model is shown. The process of CBR consists of case retrieval, proposed solution, adaptation, criticism, justification and evaluation.

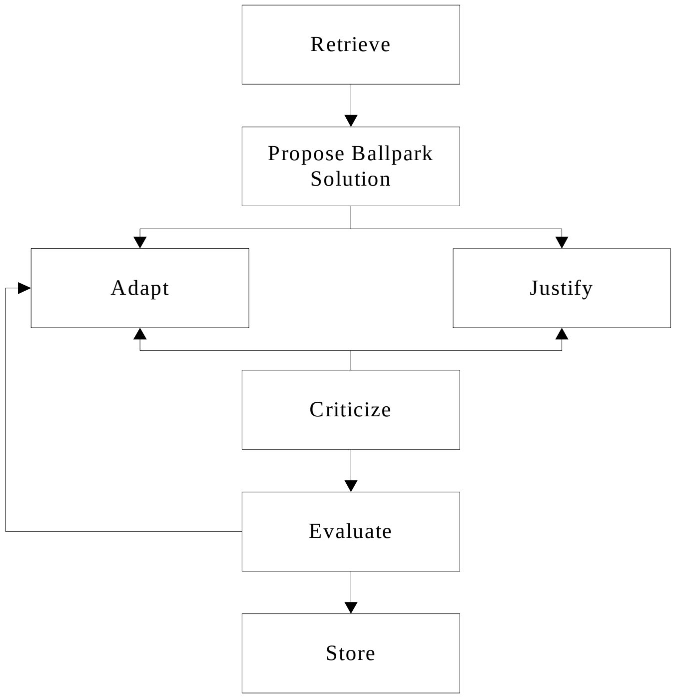


Figure 2.6 - CBR cycle of Kolodner's model

Case retrieval is the first step in the CBR model. It is a process of case extraction by indexing features in a case library. Compared with features of a new case, using matching and similarity assessment, an old case is retrieved. A relevant case retrieval supports making the right inference to a new case. A proposed solution is obtained from a process of case retrieval. The proposed solution is extracted from past cases, based on how relevant portions are needed. For instance, if a part of a solution needs to be derived, focus is a part of the old case.

Adaptation is the next step in which parts of past cases are revised and those parts are fitted into the new case. Its process is to identify which parts need to be adapted. Validation is a process including justification and criticism that is done after validation. A proposed solution is contrasted and compared with other similar cases. If the proposed solution is unacceptable, retrieved cases with similar solutions are recalled. The evaluation is a process to judge a proposed solution. The new case is retained in a store if it is successful.

Aamodt and Plaza presented a CBR model that is composed of case retrieval, reuse, revision and retention, as shown in Figure 2.7. The challenge in CBR is the problem of deciding what to store in a case, how to describe case contents and how to index the case memory for effective retrieval and reuse (Aamodt & Plaza, 1994).

The CBR cycle is described by four steps, which are to retrieve a relevant case in a knowledge repository (KR), to reuse solutions of a past case, to revise a suggested solution and to retain a new solution in the KR as new knowledge for future reuse.

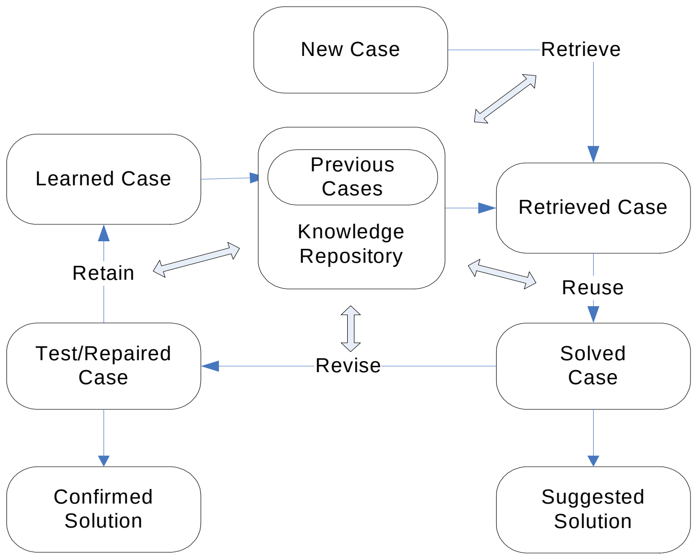


Figure 2.7 - CBR cycle of Aamodt and Plaza's model

(i) Case Retrieval

According to Aamodt & Plaza (1994), a process of case retrieval consists of three steps, which are identification of features, initial match and selection. A problem of a case is first identified. Features of the problem act as inputs that are used for case retrieval. In the process of the initial match, a set of relevant cases are retrieved in three ways, which are by following direct index pointers from problem features, by searching an index structure and by searching in a model of domain knowledge. The cases are taken in the selection process for the extraction of the most similar case. As stated by Allen (1994), nearest-neighbor algorithms, decision trees and connectionist associative memories are used for case retrieval.

(ii) Case Reuse: Adaptation

Case reuse is the second step of a CBR model, which was proposed by Aamodt and Plaza. The solutions of past cases are adapted for reuse in a new case. Transformational reuse is a case adaptation applied to the CBR model. It does not use solutions of old cases in a new case directly. However, knowledge of old cases is transformed into a new case in the process of transformational reuse. Operators index the differences between retrieved cases and the new case. Another kind of adaptation is derivational reuse. Compared with transformational reuse, the derivational reuse is a method to look at how problems are solved. The information in the retrieved case, such as an alternative method used in a successful case, is reused in the new case.

(iii) Case Revision

Case revision is a process to revise a solution to a case. It is followed by a process of case retention. In such a process, the solution of the case is revised to fit a new problem. Evaluation of the revised case is carried out in a real environment. The verified case is transferred to a process of case retention.

(iv) Case Retention

This process is used to retain the new knowledge in the system after solving a new problem. It involves identifying which information in a case needs to be retained, what form it is in, and how to index the case. Knowledge created in a new case is retained in the KR for further reuse.

CBR is a cycle for accumulating successful cases in which knowledge is retrieved, revised, reused and retained. Valuable experiences are thereby shared and diffused among staff within an organization. In addition, there are other advantages of CBR as given below. CBR

(i) suggests solutions of previous similar situations to solve new problems;

(ii) allows staff to learn from previous experiences and helps them avoid repeating past mistakes;

(iii) helps staff to focus on the important parts of a problem.

In practice, CBR has been applied in the Compaq Ltd. for improving customer services and in NEC Corporation for knowledge capture and distribution on the quality control of software (Allen, 1994). A CBR model combined with multi-agent systems for the reduction of uncertainty in a fluctuating supply and demand under a collaborative and information sharing environment (Kwon et al., 2007). The CBR has been applied to reuse and store retailer profiles, supplier profiles, manufacturer profiles and product characteristics, for making decisions on how many products to be ordered and how many products to be manufactured. This helps in reducing the costs of inventory holding, backlog and ordering for both manufacturers and retailers. To sum up, CBR is a process through a cycle of case retrieval, case reuse, case revision and case retention for achieving knowledge creation, diffusion and retention. It can be combined with the existing systems and is beneficial for improving customer services, quality of products, and inventory management processes.

2.6.5 Hybrid Artificial Intelligent System

A hybrid AI system is a combination of different AI algorithms used to match various characteristics, solve complex problems or resolve difficult situations. Kuo & Chen (2004) proposed a combination of fuzzy and neural networks to solve problems of order selection. In this case, a neural network plays a role in training the model when fuzzy logic is used for decision-making. For such kind of issue, a well-experienced project manager mostly uses his own knowledge to decide whether customer orders are accepted or rejected. For other applications, fuzzy logic is combined with a neural network to speed up the learning algorithm to manage complex decision making or diagnosis (Lin & Lee, 1991). A heuristic GA is combined with fuzzy logic for providing decisions on the problems of subcontractor selection (Wang et al., 2001).

2.7 Summary

In the 1980s, manufacturing strategies were mainly JIT, lean manufacturing and total quality management. Since the 1990s, due to rapid business changes, traditional supply chain strategies, such as outsourcing and supply chain integration, have been used. These tend to integrate the supply chain effectively, and competitors are considered as partners in the supply chain. It is envisaged that the integrated supply chain will face competition from other supply chains in future.

After reviewing some supply chain strategies such as push/pull system, JIT, VMI and CPFR, it is found that VMI and CPFR are new trends now being implemented in the supply chain. It is likely that they will reduce the inventory, lead time and increase collaboration and profits among suppliers and customers. Compared with CPFR, the process of implementation with a VMI strategy is relatively simple. Resources involved in the process are comparatively few. It is more suitable for some SMEs who are thinking of changing from the traditional approach to the VMI strategy. This study focuses on simulating the situation after the change to the VMI strategy.

Nowadays, supply chain problems are mostly being solved by mathematical analyses and simulation games, such as the beer game and simulations for enterprise applications. The design of the beer game simplifies traditional strategies so much that they cannot reflect realistic supply chain operations. Customization of the supply chain strategy is needed so that supply chain parties can be adapted to the new business environment. Although some commercial simulation tools are available, they are mainly concerned with how to solve business problems and how to gain maximum profit within the organization. However, to drive supply chain operations to success, knowledge management is also needed. In the view of an organization, knowledge is considered as a treasure. There is a great need for knowledge-based simulation systems, which will generate knowledge and make it available for reuse in the organization.

To develop a simulation platform for SCM, artificial intelligence such as neural networks, fuzzy logic and genetic algorithms can be considered. It appears that CBR is more suitable for the development of this kind of simulation system since it allows for the acquisition, retention and diffusion of knowledge for addressing the problems of SCM.

In the reviews on knowledge management and its importance in the supply chain, it is interesting to note that there is a relationship between supply chain strategies and KM. Integration of the concept of the supply chain strategy, KM techniques and a simulation tool is essential for solving problems. There is a need for developing the KBSP which combines four-domains of knowledge, involves the supply chain strategy, KM, simulation and artificial intelligence.

Chapter 3 Design and Methodology of the Knowledge-based Simulation Platform

Most supply chain parties spend a lot of time planning and implementing supply chain operations, from ordering, inventory checking and re-ordering to delivery. The parties attempt to apply various strategies, such as the JIT system and the VMI strategy for reducing lead time and keeping the inventory at a low level. As the re-designing of traditional operations often involves changes and may produce unpredictable effects, staff may be reluctant to adapt and work in such a changing environment. Besides, the staff do not know how to change the direction of a current situation to a new approach in order to gain more profit. In this study, the KBSP, a web-based platform, is proposed for multiple retailers and a supplier to run the simulation for prediction of effects on VMI. Its design and methodology are described in the following sections.

3.1 Traditional Approach Vs KBSP

In traditional supply chain operations, a retailer is responsible for sales forecasts. A schematic diagram of the conventional approach is shown in Figure 3.1. As for product replenishment, a retailer first checks the inventory status and sales information. Next, the purchase orders are sent to a supplier via electronic data interchange (EDI).

The supplier looks at the available inventory and determines whether the order can be filled. If stock is available, advance shipment notices will be sent to the retailer. Upon receiving invoices from the supplier, the retailer checks the products and makes payment. For completing such a cycle, long process time and complex procedures, from placing orders, sending purchase orders, receiving invoices to stocking-in products, are needed.

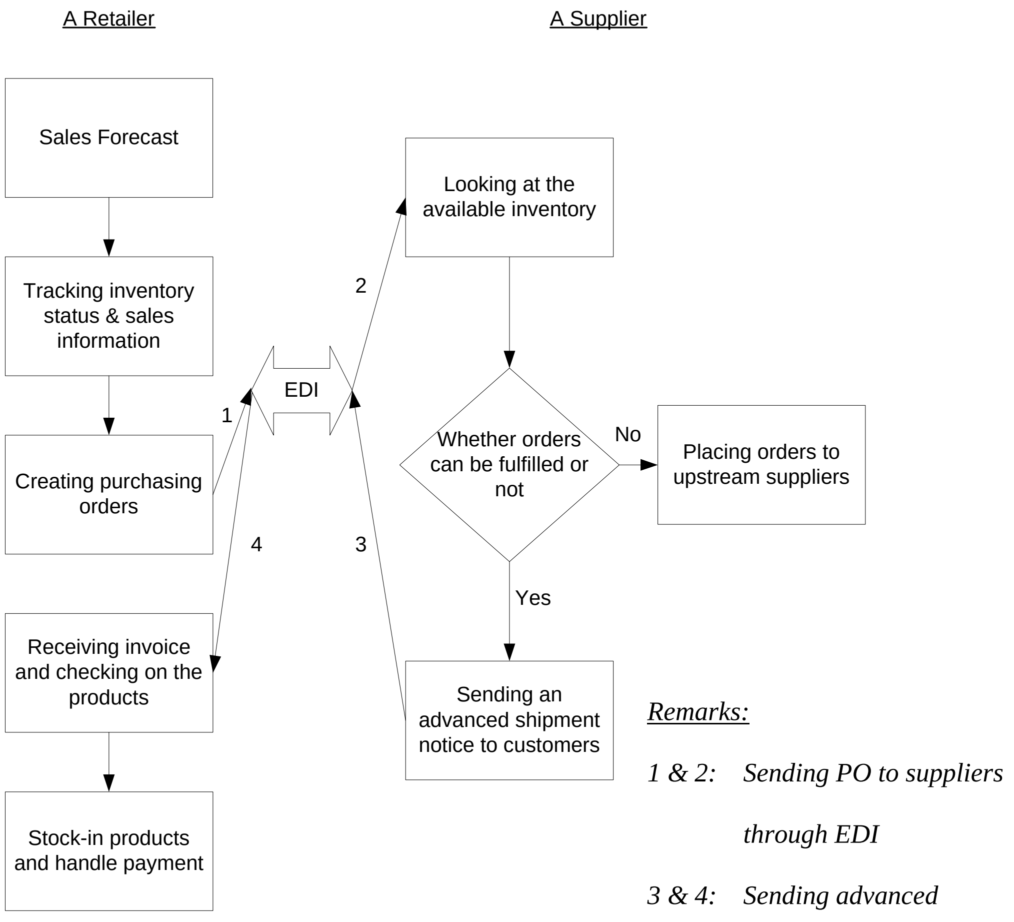


Figure 3.1 - A schematic diagram of conventional approach

3.1.1 Challenges of the Conventional Approach

With the conventional operational approach, supply chain parties such as suppliers, manufacturers, retailers and customers encounter the following problems:

(i) Inaccurate sales forecast;

(ii) Keeping a large amount of inventory;

(iii) Insufficient trust between parties;

(iv) Long lead time for product replenishment.

In some cases, customer demands fluctuate frequently. It is difficult for retailers to provide accurate sales forecasts. High inventory levels are required to be kept in both suppliers and retailers, which is due to the lack of trust and invisibility of information between suppliers and retailers. Besides, manufacturers may face problems such as long lead time for production and complex replenishment processes with their upstream suppliers. When sudden demands are requested by customers, the bullwhip effect occurs in a whole supply chain. There would be insufficient finished products provided by manufacturers to cope with the problem of fluctuating demands. As a result, the customer service level and customer loyalty decrease. For high product availability and fast customer response, a VMI strategy is becoming widely adopted in SCM.

3.1.2 Problems on Existing Systems

To bring out the significance of the KBSP, some enterprise systems are briefly introduced and problems with the existing systems are then discussed. In a commercial market, use of information technology to manage business operations is a critical step used to enable the business to succeed. Taking SAP as an example, it is an enterprise system to enhance the supply chain management. It is combined with enterprise resource planning (ERP) and customer relationship management (CRM) so that it improves on sales and marketing operations, inventory management and manufacturing.

Oracle is another example of the enterprise system. Except for basic functions in enterprise applications, it functions as a server and a database. It provides a high degree of customization to meet customers’ needs. These two systems are powerful tools for collecting data and doing the analysis in SCM. However, a large amount of capital investment, for the provision of hardware, software and maintenance, is needed for full implementation. Due to the limitation of costs, SMEs will implement the systems only with great effort.

One solution is to understand basic requirements of each process in order to select required components for building the systems. Most commercial systems are designed to extend the levels of customization. This provides SMEs with many features of the systems for selection. It is effective for users who intend to reduce costs on the system implementation. However, a large extent of customization leads to a wide variety of systems in a supply chain. Integration of existing systems becomes an issue that the SMEs usually have to face.

In most cases, there is no platform for all staff to access and share their experience, or working guidance. Except for managerial staff, staff who work in the same department can only check the information. When customers ask for any information such as status of orders, a number of procedures are needed. If there is lack of communication among departments, a long time is needed to find the required information. It is inefficient in operations and may cause a reduction in customer satisfaction.

Besides, some existing systems can be developed by the organization itself. For instance, a non-licensed platform for the operational systems, such as Linux or an open source language such as Java language, is commonly used. This can reduce costs and customize the features for their needs. If staff leave the organization, it is difficult for others to follow up the technical problems on such systems. Once the system is upgraded to provide more applications, much more effort is needed.

In an enterprise, use of commercial systems is effective to improve workflows in each department. This helps to solve problems on the management of human resources, production, sales and marketing. Based on needs, at each stage of the processes, the commercial system should be customized. As the design of business workflow in each enterprise is different from that in other enterprises, the functions built into the system are changed. For better collaboration among parties, a common platform should be provided to integrate these functions. The effect of integration is a critical factor for enterprises to consider. Most enterprises are likely to gain benefits, such as an increase in profits and market share, from the integration. For predicting the impact on integration, a simulated platform will be needed.

As mentioned earlier, the KBSP provides a simulated platform for supply chain parties to integrate current operations. The effects of integration are simulated. During the processes, the implicit knowledge is transferred to explicit knowledge in the form of guidance that is retained in the KBSP for reuse.

3.1.3 The Importance of the Knowledge-based Simulation Platform

To cope with the challenges in the conventional approach, the VMI strategy is proposed for SMEs to manage supply chain operations. The KBSP would bring a critical concept in SCM and it would play an essential role in simulating supply chain operations. A comparison between the conventional approach and the KBSP is shown in Figure 3.2. In the KBSP, special features, such as knowledge acquisition and diffusion, are added to retain knowledge within an organization to provide high quality customer services. The advantages of the KBSP are as follows:

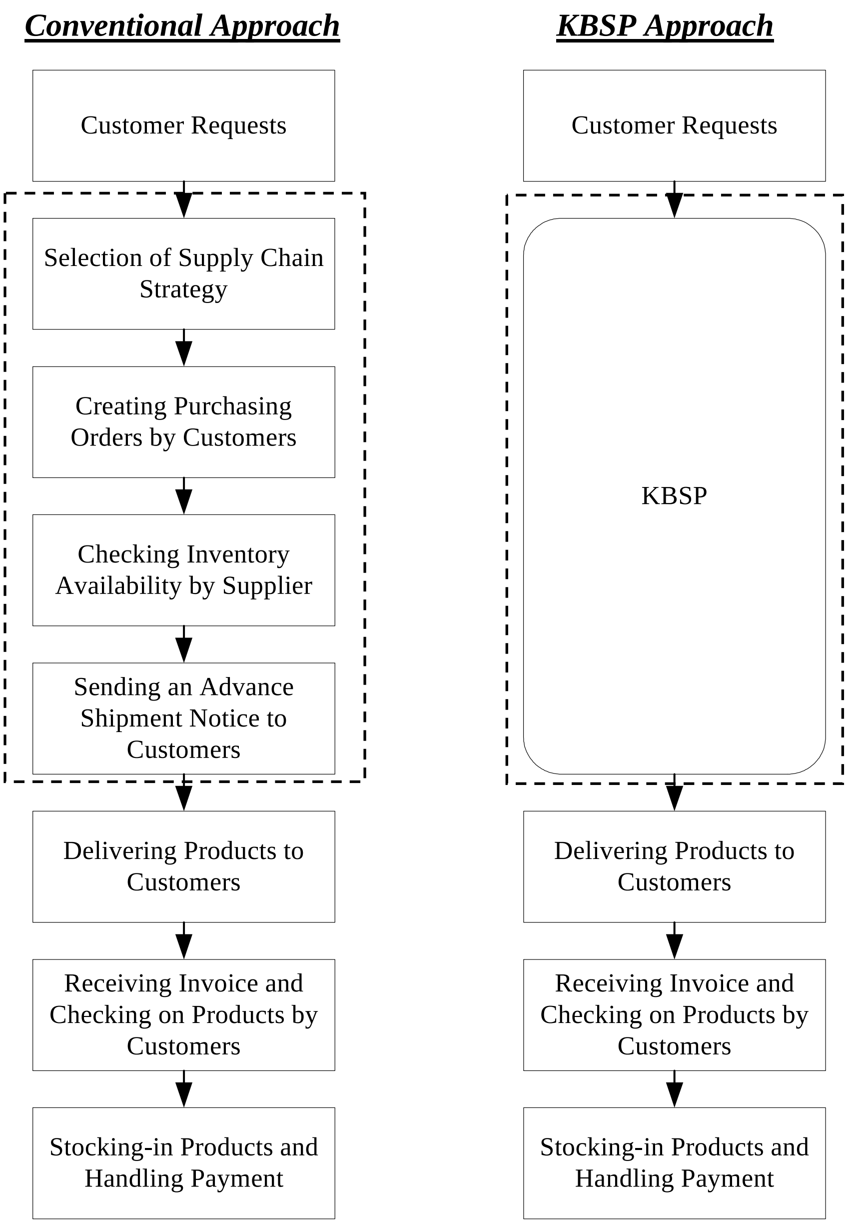


Figure 3.2 - A comparison between the conventional and the KBSP approaches

(i) Acquisition and Diffusion of Knowledge

An artificial intelligence (AI) technique, such as CBR, is embedded into the KBSP. This is to facilitate knowledge capture and sharing among staff. It provides training and suggestions to new staff, and helps in reducing time and expenses on training. Traditionally, when an experienced member of staff leaves the organization, his know-how may be lost. However, with KBSP, the knowledge can be retained and retrieved. Such knowledge is diffused to new staff once they learn from the existing practice via the platform.

(ii) Real-Time Response

Since records are updated automatically in the KBSP, customers, retailers and suppliers are able to give a real-time response to their business partners. In the KBSP, a user plays a role of the supplier, collaborates with the retailers, and monitors the inventory and process status. This is an efficient and effective way to mimic real situations for supply chain processes.

3.2 Working Principle of KBSP

The KBSP provides a web-based platform for supply chain parties to simulate their daily operations. It is built for customers, retailers and suppliers so they can manage their daily transactions, including order placement and order tracking, inventory management as well as product replenishment. These parties can collaborate with each other in a real-time mode. Through an interactive user interface, data are transferred and stored in the back-end applications. In the KBSP, sales demand is analysed by a spectrum analysis module. The generated pattern is an index for comparison with past cases in the case-based reasoning model. Recommendations are then suggested for use in the simulation model. Parameters and strategy are pre-set into the simulation model. Historical data are input to generate and evaluate simulation results of the inventory, fill rate and stock-out rate.

The concept of the CBR model is embedded in the KBSP, which is able to facilitate knowledge production, acquisition, sharing and diffusion within organizations (Davenport & Prusak, 1998; Preece et al., 2001). With the use of

KBSP, tacit knowledge, such as the techniques of managing the inventory for retailers and the skills of problem solving, is efficiently changed to explicit knowledge, such as a set of guidelines or documents. The most valuable feature of KBSP is to capture the knowledge of well-experienced staff and make such knowledge available to new staff. This is an effective and efficient way to train new staff to handle complex supply chain processes. As shown in Figure 3.3, the KBSP is composed of four components: the simulation model, the spectrum analysis module (Mulgrew & Grant, 1999), the dynamic forecasting model (Cheung, 2003) and the CBR model (Aamodt & Plaza, 1994).

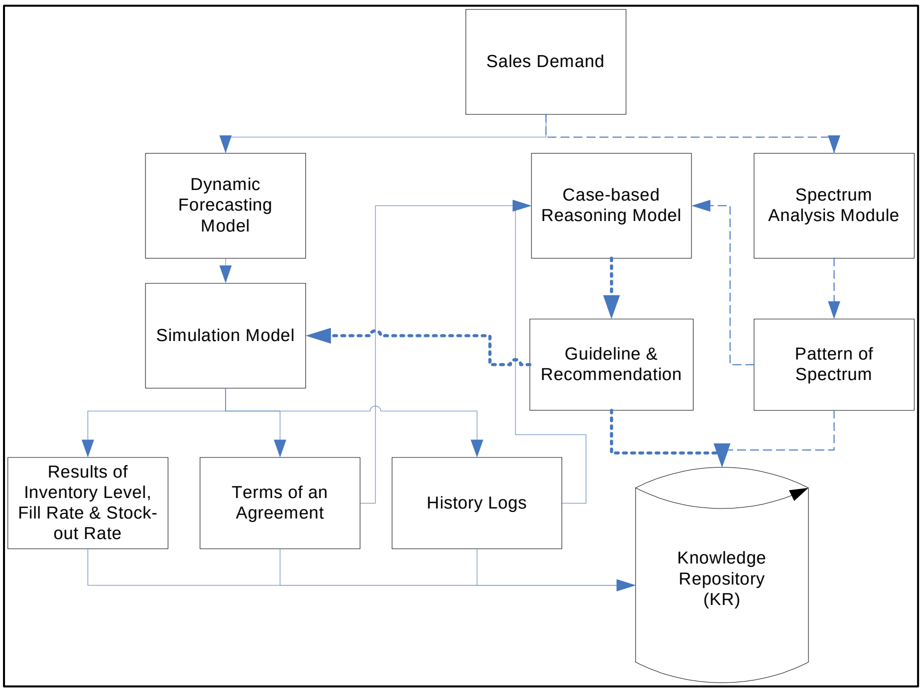


Figure 3.3 - Architecture of a knowledge-based simulation system (KBSP)

The working principle of the KBSP is related to how knowledge is captured in the processes of the simulation. Knowledge includes the transaction processes of product delivery and replenishment that are logged in the KR. Each event, which happens in the supply chain simulation, represents how the staff make decisions and how they plan for dealing with orders in future. The log saved in the KR is valuable knowledge that reflects the operations.

Knowledge retrieval is a process to extract a relevant case from a number of past cases. In the previous section, transformation from customer demands to a series of spectra has been discussed. Each set of spectra reflects the ordering pattern of a customer. In a process of case retrieval, a player compares the spectrum of a new case with that of past cases. A similar case is chosen by the player and the relevant information of the case is then extracted. A spectrum is used as an index to retrieve a similar case from the KR. The information consists of the parameters of a blanket agreement of a VMI strategy and records of simulated processes. These are useful for new staff to learn from past cases. A schematic diagram of the CBR model is shown in Figure 3.4.

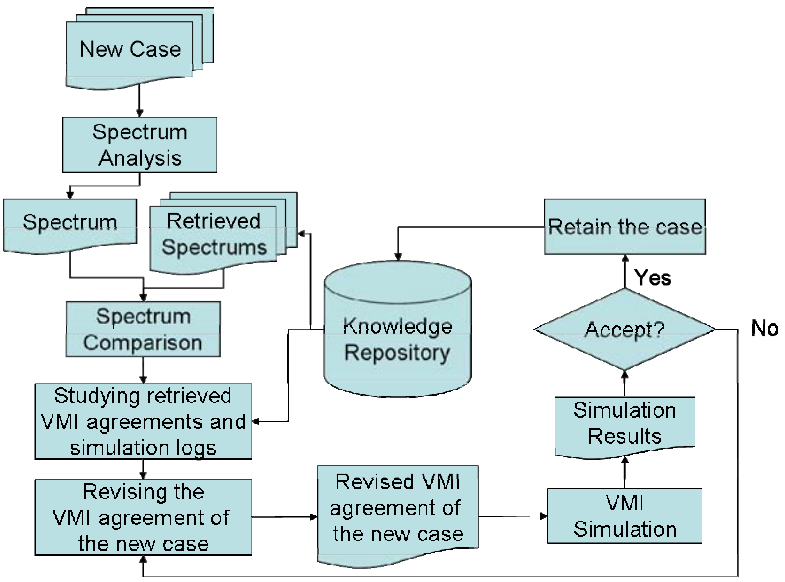


Figure 3.4 - A schematic diagram of the CBR model

In a cycle of the CBR model, case reuse and case revision include the terms of an agreement, such as batch size and lead time. The retrieved values are revised if necessary. The staff initiates a new simulation based on a set of parameters and logs.

This shows how tacit knowledge can be transferred to explicit knowledge, which is represented in logs. A snapshot of the logs is shown in Figure 3.5.

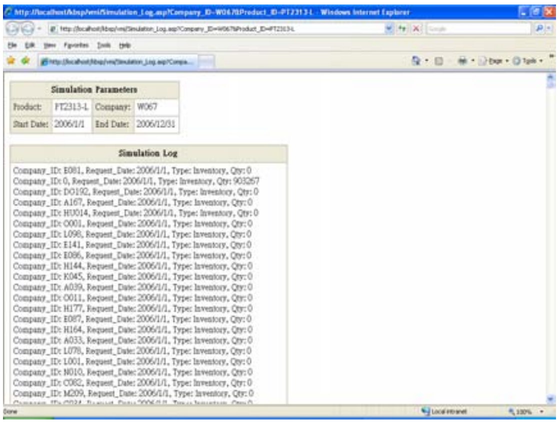


Figure 3.5 - A snapshot of history logs

With reference to the practice in the past case, new staff are able to mimic a process of handling transactions. If the results obtained from the simulation are better, a new case with the terms of an agreement and records is identified as a successful case and retained in the KR. A description of the KBSP is summarized in Table 3.1.

Table 3.1 - A description of the KBSP

KBSP	Description
Knowledge defined in the simulation	Transaction logs and parameters of a VMI agreement
Index for searching	A spectrum generated by the spectrum analysis model is an index for searching
Case Retrieval	The new case is compared with the spectrum of a past case; a similar case is retrieved.
Case Reuse	The terms of the agreement and the method of doing transactions are reused.
Case Revise	The parameters are revised to suit the current situation.
Case Retain	A successful case with a new set of parameters is retained in the KR for future reuse.

In addition to the CBR model, a simulation model is also incorporated in the KBSP. The simulation model is designed in three stages. This assists suppliers and retailers to select appropriate strategies, such as the VMI strategy or the push system, to redesign their supply chain operations. In the first stage, the supplier uses the KBSP to find optimal parameters such as batch size, safety stock and reorder level. These values will be used in the second stage.

In the second stage, retailers and a supplier make transactions through the platform. The dynamic forecasting model is designed for a supplier to predict and plan the retailer's needs. The concept of forecasting is commonly used in ordering and replenishment process of VMI strategies. Compared with traditional forecasting models, such as moving average or seasonal forecasting, the adaptive time-series forecasting model (Cheung et al., 2003) is adopted, based on an autoregressive (AR) time-series model (Box & Jenkins, 1996). In this process, the orders data are obtained and transferred to a pattern generated by a spectrum analysis module. Based on the generated pattern, a supplier can analyse current transactions effectively. It is efficient in managing orders and provides a fast response to fulfill the retailer's requirements.

The CBR model is built in the KBSP, which is used to provide recommendations or guidance for staff to manage fluctuating demand and complex operations. It is also used to facilitate the acquisition of tacit and explicit knowledge in supply chain operations (Aamodt & Plaza, 1994; Kolodner, 1992; Simoudis, 1992). At the end of the cycle, successful cases are retained in the KR for future use.

3.3 Simulation Model

A simulation model for the KBSP is divided into two parts which are feasibility of a change from a push strategy to a VMI strategy and negotiation of the terms of reference for a VMI agreement with a retailer at a web-based platform. For a change from a push strategy to a VMI strategy, a (s, Q) policy and a (s, S) policy are used to traditional push systems and a VMI strategy, respectively. In the simulation mode, outcomes of fill rate and the stock-out rate under the VMI strategy are better than that of the push strategy. The feasibility to change from the traditional approach to the VMI strategy will be evaluated. An optimization model is used to find an optimal value of batch size which acts as a starting point for negotiation of the VMI agreement in the web-based platform. For part 2 of the simulation model, a supplier, based on the optimal value of batch size, starts a process of negotiation with a retailer under the VMI strategy.

3.3.1 Mathematical model of Simulation

In the simulation of a push strategy, retailers place orders to a supplier. The products are delivered based on the expected delivery date. When inventories of products at the supplier drop to a certain level of safety stock, the supplier makes an order to its upstream supplier for product replenishment. According to Hax & Candea (1984), the inventory strategy of (s, Q) policy is used in the model. In a (s, Q) policy, when inventory drops to the safety stock (s), products will be replenished at a certain quantity (Q). Assumptions made in the scenario are:

(i) Demand is discrete;

(ii) Backorders are not allowed;

(iii) Partial delivery is not considered;

(iv) Quantity discount is not available.

The inventory kept at time interval i is calculated by Equation (3.1). Once the inventory level is lower than the level of safety stock s in the period $i-T$, replenishment quantities Q_i are ordered in the period i.

$$I_{t+1} = I_0 - \sum_{p=0}^{q} \sum_{i=0}^{t} D_{p,i} + \sum_{i=0}^{t} Q_i \ , \ \forall p \in N, \forall i \in N \qquad (3.1)$$

where I_0 is an initial value of inventory level, I_{t+1} is the inventory which is kept in period $t+1$, $D_{p,i}$ is the demand of a customer p in period i, q is total number of customers, Q_i is replenished quantities in period i.

With a VMI strategy, a supplier has the responsibility to decide or suggest when to replenish products to a retailer. The supplier helps the retailer to conduct the forecast. In the traditional approach, the retailer places an order when an inventory level drops nearly to the level of the safety stock. With the VMI strategy, the responsibility of replenishment is shifted to the supplier. The aim of the retailer is to keep stock at the lowest level. Once the inventory level is decreased to a certain point, the supplier will make a decision on replenishment. In most cases, ordering processes are omitted. Products are directly sent to the retailer after the replenishment quantities are confirmed. Thus, a supplier keeps a large amount of inventory to meet the requirements from retailers.

The scenario of a VMI strategy is for the retailer and the supplier to negotiate a blanket agreement. A period is normally set which lasts for one year or two years. A dynamic forecasting model proposed by Cheung (2003) is added as a component to forecast fluctuations in demand. Users can select which strategy they would like to use. They must compare the outcomes and make a decision on their long-term agreement. The inventory strategy in the model, with the VMI agreement, is (s, S) policy. When the inventory drops to the safety stock (s), products will be replenished at the reorder level (S). Assumptions made in the scenario are:

(i) Demand is discrete;

(ii) Backorders are allowed;

(iii) Partial delivery is not considered;

(iv) Quantity discount is not available.

In the following part, the optimization model used in the VMI strategy is described. In practical cases, costs invested in reengineering processes are difficult to estimate for both parties. In the simulation, the inventory level, stock-out rate and fill rate are used to decide whether the strategy is successfully achieved or not. Starting with the simulation, a supplier selects a retailer and uses one-year historical data as a training set.

The measurement of inventory level is related to how many products are delivered to the retailers per batch and when replenishment processes take place. The user inputs an upper limit and a lower limit of batch size. When fine-tuning one of the parameters such as batch size and sales demand, other parameters such as safety stock and reorder level are fixed. In the same way, a number of combinations are generated by an iterative method. When the optimal outcome is obtained, the process of fine-tuning is terminated. The objective is to minimize the inventory level of the supplier and of the retailer. The way to do this is shown in Equation (3.2). According to Hax & Candea (1984), when inventory in period $i-T$ reaches the level of safety stock, replenishment quantities Q_i in period i are received.

$$I_{t+1} = I_o - \sum_{p=1}^{q} \sum_{i=1}^{t} D_{p,i} - \sum_{u=1}^{v} \sum_{i=1}^{t} B_{u,i} + \sum_{i=1}^{t} Q_i \ , \ \forall u \in N, \forall p \in N, \forall i \in N \qquad (3.2)$$

where I_o is an initial value of inventory level at the beginning of a simulation, I_{t+1} is an inventory level in period $t+1$, $D_{p,i}$ is a demand of a retailer p in period i, q is total number of retailers who do not use the VMI strategy, a batch size $B_{u,i}$ is delivered to a retailer u, v is total number of retailers using the VMI strategy and Q_i are quantities which are replenished from an upstream supplier in period i.

From Equation (3.2), a value of Q_i is calculated by Equation (3.3).

$$Q_i = \begin{cases} ROL - I_{i-T} & if \ I_{i-T} \le SS \\ 0 & if \ I_{i-T} > SS \end{cases} \ , \ \forall i \in N \qquad (3.3)$$

where ROL is a reorder level, I_{i-T} is an inventory level in period $i-T$. T is the lead time of replenishment from the upstream supplier. SS is the level of safety stock, which is kept by the supplier.

According to Choi, Dai & Song (2004), the fill rate is equal to the proportion of the demand that is to be fulfilled immediately. The fill rate and stock-out rate are calculated by Equation (3.4) and Equation (3.5).

$$FR = 1 - \frac{E(U)}{E(D)} \qquad (3.4)$$

where FR is a fraction of satisfied demand from an available inventory, $E(U)$ is expected unsatisfied demand and $E(D)$ is expected demand.

$$SR = \frac{1}{T} \sum_{t=1}^{T} X_t \ , \ \forall t \in N \qquad (3.5)$$

where SR is the fraction of having stock out between two successive replenishment deliveries, X_t is the probability of having stock out in period t, as shown in Equation (3.6).

$$X = \begin{cases} 1 & Y > 0 \\ 0 & Y \leq 0 \end{cases} \tag{3.6}$$

where Y are the quantities requested by retailers but not available in period T.

During the simulation, the optimization model is used to refine the agreement term on batch size. The optimal value of the batch size is a suggestion as a term of the VMI agreement. A supplier is based on the value, which acts as a starting point, for negotiation of terms with a retailer. In practice, it is difficult to predict outcomes for the adoption of a new strategy. Results of the simulation model reflect what the situation will be after the implementation of VMI. This helps both the supplier and the retailer to determine whether the strategy should be redesigned.

3.3.2 Design of Simulation Model with One Player

As mentioned in a previous section, three parties are joined in a simulation. A one-player design mode is introduced first, as shown in Figure 3.6. Customer information such as sales demand, inventory and forecast data are preset in the system. A user plays the role of a supplier and selects a strategy of the push system or of the VMI. If the (s, S) policy is selected in the traditional push system, parameters of safety stock, lead time and reorder level are considered.

With a VMI strategy, replenishment of product is decided by the contract terms between a retailer and a supplier. The (s, S) policy is used for an upstream replenishment cycle from a supplier. Compared to the simulation of the push system, more parameters are considered. These are batch size and in-transit lead time from a supplier to a retailer. The parameters of safety stock, reorder level and lead time of replenishment are considered from an upstream supplier to a downstream supplier. The objectives of VMI simulation are to minimize inventory level, stock-out rate, lead time and to maximize fill rate. Two scenarios are considered in the simulation of VMI strategy. They are the feasibility of VMI strategy and addition of a retailer for implementation of VMI strategy.

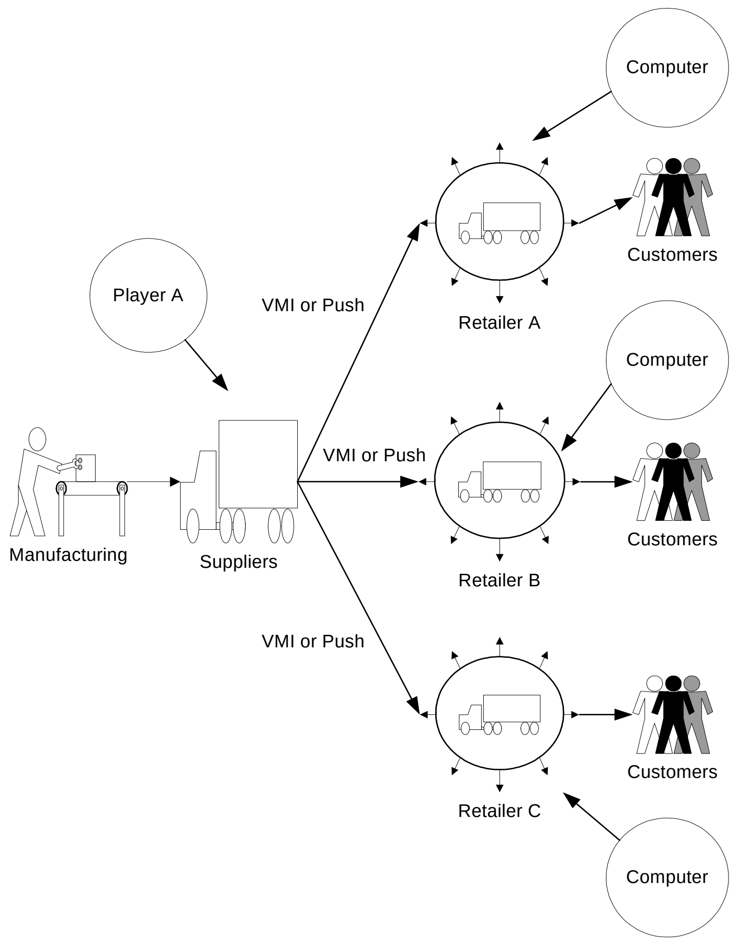


Figure 3.6 - One-player design mode of the simulation

3.3.2.1 Feasibility of VMI Strategy

Both the retailer and the supplier change from a conventionally strategic mode to a VMI strategy so that cost is one of the main considerations. Achievement of a VMI strategy is carried out under a long-term agreement. Before signing an agreement, both parties investigate the current situation of all parties and evaluate the feasibility of success. Due to a large amount of investment to reengineer operational processes, both parties predict what effects will be brought out and how to overcome problems. If a decision is not made in a proper way, it may lead to losses for both parties. Thus, it is vital to evaluate the feasibility of the VMI strategy.

As mentioned previously, historical data over a one-year period are selected for the training process. In the application phase, historical data in the following year are used to simulate the actual demand in the future. The user does not know the value until after a certain period. The dynamic forecast model is used in the VMI strategy. It provides a way of predicting sales demand for the retailers. When the simulation starts, the user changes batch size, safety stock and reorder level in order to handle the fluctuations in demand. Until the simulation stops, crossing a period of 52 weeks, the results generated by a retailer and a supplier are measured separately.

With the VMI strategy, besides the consideration of costs, the relationships among parties are also important. One of the factors related to the costs is amount of inventory kept. The relationship between retailers and a supplier is measured by service level and product availability. According to Chopra & Meindl (2004), the service level is related to stock-out rate. The product availability is a level of available products provided to retailers. It affects the product fill rate given to retailers. On the other side, a supplier changes the reorder level and safety stock to replenish the products from its upstream supplier. The outcomes which are the inventory level, fill rate and stock-out rate are generated at the end of the simulation. The fill rate and the stock-out rate are the percentage of orders delivered on time and the percentage of products that are out of stock, respectively.

3.3.2.2 Addition of Retailers

In the previous section, the feasibility of VMI strategy was tested. The addition of retailers is to test how many retailers a supplier can handle in the VMI strategy. Suppose the number of products supplied from an upstream supplier is fixed at a certain amount, and the parameters of the original retailers are fixed for use in either the VMI strategy or the push system.

By using the same methodology, a supplier changes the parameters of batch size, safety stock and reorder level, to manage the transactions. After running the simulated processes for over one year, the outcomes will be shown. If the outcomes are better than before, the supplier considers the addition of a retailer with the use of the VMI strategy. The steps are repeated and the results are then checked. Backorders are not allowed at the end of the VMI simulation. The occurrence of the backorders implies that the supplier is not capable of fulfilling the new sales demand. The suggestion for adding a customer will be rejected.

3.3.3 Design of Simulation Model with Two Players

In a two-player design mode, only one fixed party, an upstream supplier, is provided, as shown in Figure 3.7. The supplier is required to select one of the retailers to simulate the VMI strategy and set the remaining retailers for use in the push system.

In this case, a retailer and a supplier negotiate terms of a blanket agreement. After confirming the terms, a simulation starts. A period of simulation is the same as in the one-player design mode. A supplier is responsible for providing a forecast and replenishing the sales demands of retailers. Through a simulation process, the two parties attempt to find out the most suitable parameters to be applied to their current situations, including lead time, batch size, safety stock level and reorder level, in order to keep a low inventory level, shorten lead time, decrease stock-out rate and maximize the fill rate.

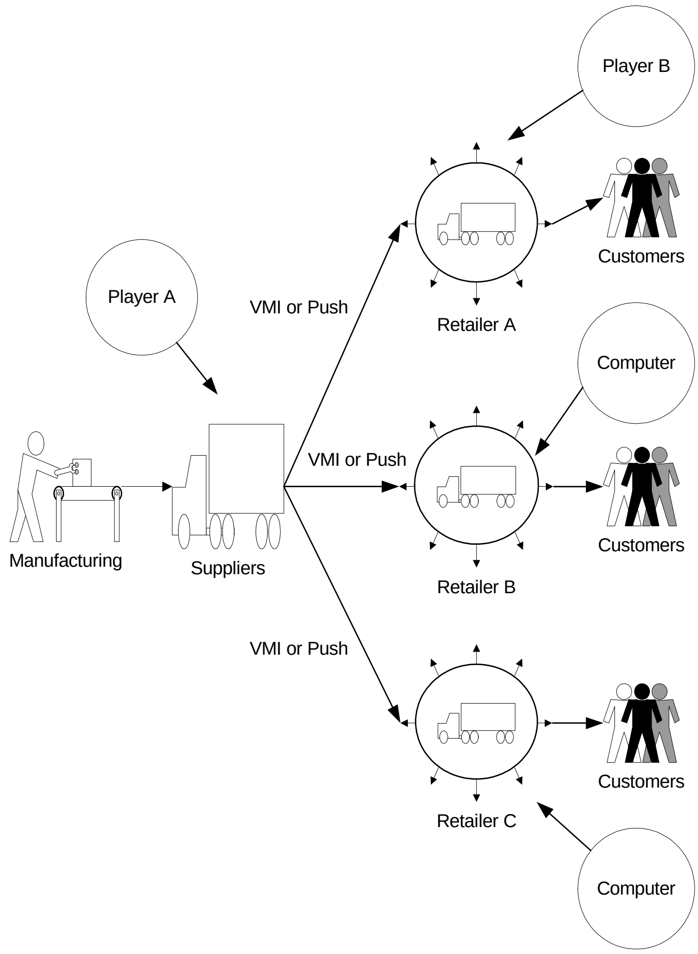


Figure 3.7 - Two-player design mode of the simulation model

3.3.4 Design of a Simulation Model with Multi-players

In the multi-players mode, a number of retailers and a supplier are allowed to participate in the KBSP as shown in Figure 3.8. For the first step of the simulation, retailers select strategies of either a push system or a VMI strategy. The supplier suggests the initial set of terms in a VMI agreement. As introduced earlier, the batch size is optimized in the simulation model that is reused as a suggested parameter in the agreement. Except for the use of a set of optimal parameters, the supplier compares the demand pattern, which is generated in the spectrum analysis module, with the pattern of a past case for the retrieval of terms as a reference in the negotiation. According to their current practices, the retailers negotiate the terms with the supplier. They bargain the terms with the supplier for a low level of inventory, a low stock-out rate and a high fill rate until all retailers and the supplier agree on the terms of reference for the VMI agreement. These terms are used in the simulation.

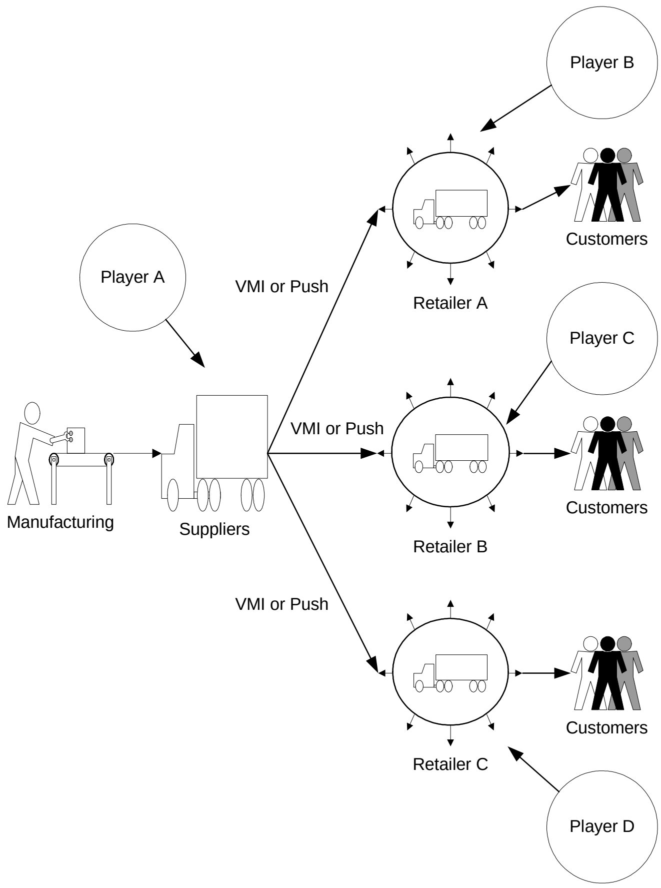


Figure 3.8 - Multi-player design mode of KBSP

In the KBSP, the retailers provide their daily transactions to the supplier. The supplier forecasts the number of products to its retailers on time. At the end of the simulation, inventory levels of retailers, fill rate and stock-out rate are reported. The outcomes of the simulation are valuable information that helps them in making a better decision on the issue of redesigning a strategy. Besides, the supplier and the retailers learn from the processes of simulation as they may find some unexpected problems in the processes of fulfilment of requirements, product distribution and product replenishment. The implicit knowledge is transferred to explicit knowledge such as recommendations and simulation logs that are stored as a new case in the KR for further reuse.

In the simulation, the objective function is set to find the minimum inventory level. In the study, constraints such as expected fill rate and stock-out rate are set, based on the requirements of a retailer. For the safety stock parameters, reorder level and batch size, an upper limit, a lower limit and a value of an increment are pre-defined. In the process of optimization, all retailers' demands are aggregated at a certain period. For changing parameters each time, a new inventory level is generated. A local optimal value is a point at the lowest inventory level. Occasionally, a number of sets of parameters used in the optimization come up with the same value of the minimum inventory level. A second criterion for the selection of an optimal solution is to find a set of parameters, which is close to the expected fill rate and the stock-out rate. If more than one set of parameters fulfill the requirements of optimization, the first set of parameters is selected as an optimal solution.

Compared with the beer game, multi-retailers are allowed to do transactions with their supplier in real time. In practice, working with more parties in a supply chain increases complexity in operations. Some unexpected problems are not tackled by direct interaction between a retailer and its supplier. They are related to new requests of other retailers and actions of a supplier taken for them. Human intervention in supply chain operations is not easily modeled by a set of rules. Such a simulation creates an environment for different parties to mimic a scenario in which a VMI strategy is implemented. The predicted outcomes help them to decide whether a new strategy is feasible or not.

3.4 Spectrum Analysis Module

Discrete Fourier Transform (DFT) is a mechanism which can be used to decompose a periodic signal to simple periodic functions, in the form of sine and cosine functions (Harris, 1978). Fast Fourier Transform (FFT) is a method to efficiently compute DFT using a time series of discrete data samples and facilitate power spectrum analysis (Cochran et al. 1967). In practice, most customer demand patterns are periodic and data are most likely to be time dependent. Thus, it is suitable to use the FFT algorithm to perform a power spectrum analysis in which the ordering data are transformed into the frequency domain. A pattern of the spectrum is generated and it is effective in extracting characteristic features of customer demand.

Customer demand is denoted by $z(k)$ with $k = 0,1,2,..., N-1$, where N is the total number of samples which represents the total number of customer orders. The power spectrum of customer demand is defined by Equation (3.7).

$$Z(v_n) = \sum_{k=0}^{N-1} z(k) \exp(\frac{-2\pi jkv_n}{N}) \qquad (3.7)$$

where n is an integer, v_n is a frequency component of the customer order, and N is the total number of samples.

According to the Sampling Theorem (Proakis & Manolakis, 1996), the sample rate v_{sample} must be at least twice that of the highest non-zero frequency component $v_{\max}$ contained in customer demand, expressed as:

$$v_{sample} \geq 2v_{\max} \tag{3.8}$$

The power spectral density (PSD) is directly determined from DFT. The periodgram $|Z(v_n)|^2$ is obtained by transforming the real data. These yield N transform points corresponding to N real data points. In order to minimize the distortion of the true spectrum attributable to Gibb's phenomenon, the spectral window corresponding to the Hanning lag window is used to obtain the PSD (Mulgrew & Grant, 1999). The Hanning window is selected as it has its spectral intensity concentrated at its main lobe in the frequency domain, and is shown in Equation (3.9) as

$$PSD(v_o) = 0.25[Z(V_p)]^2 + 0.50[Z(V_o)]^2 + 0.25[Z(V_s)]^2 \tag{3.9}$$

where $PSD(v_o)$ is the power spectral density at a particular frequency v_o, v_p and v_s are the preceding and the succeeding frequencies for v_o, v_p and v_s respectively.

3.5 Dynamic Forecasting Model

Based on the Autoregressive (AR) time-series model (Box & Jenkins 1976), Cheung (2003) proposed a dynamic forecasting method which makes use of a modified least mean square (MLMS) algorithm to predict customer demand in a VMI strategy. The input of the dynamic forecasting model is sales demand, which is provided by retailers. The prediction error of the AR time-series model is calculated by the determination of the difference between the actual and the estimated values. An estimated value is calculated by Equation (3.10) as follows:

$$\hat{y}(k) = N(K) + \sum_{i=1}^{n} a_i(k)y(k-i) \tag{3.10}$$

where $N(k)$ is white noise, $y(k)$ is sampled data at k^{th} instant of time and $a(k)$ are time-series coefficients.

Equation (3.11) can be expressed as:

$$\hat{y}(k) = N(K) + \mathbf{a}^T(k)\mathbf{y}(k-i) \qquad (3.11)$$

where $\mathbf{a}(k)$ is defined as a n^{th} order time-series coefficient vector defined in Equation (3.12) and $\mathbf{y}(k)$ is defined as a vector of current n and $n-1$ past data in Equation (3.13).

$$\mathbf{a}(k) = [a_1(k), a_2(k), ..., a_n(k)]^T \qquad (3.12)$$

$$\mathbf{y}(k) = [y_1(k), y(k-1), ..., y(k-n)]^T \qquad (3.13)$$

The prediction error $e(k)$ is defined as

$$e(k) = y(k) - \hat{y}(k) \qquad (3.14)$$

where $e(k)$ is the difference between sampled data $y(k)$ and the AR model estimated value $\hat{y}(k)$.

Based on the MLMS algorithm, the $\mathbf{a}(k)$ of AR time-series model is modified to $\mathbf{a}(k+1)$ and $e(k)$ is the prediction error at the k^{th} instant of time in Equations (3.15) and (3.16).

$$\mathbf{a}(k+1) = \mathbf{a}(k) + \beta e(k)\mathbf{y}(k) \qquad (3.15)$$

$$e(k) = y(k) - \mathbf{y}^T(k-a)\mathbf{a}(k) \qquad (3.16)$$

where β is the adaptation gain which determines the step size of change of $\mathbf{a}$ at each adjustment, and $\mathbf{y}(k)$ is defined as a vector of current n and $n-1$ past data.

A schematic diagram of the dynamic forecasting model is shown in Figure 3.9. If the abnormal errors are greater than the acceptable level, the forecasting process is stopped and the model needs to be re-calibrated. The dynamic forecasting model is re-calibrated by updating the adaptation gain (β), and vectors of **a**(k) and **y**(k) to the optimum values. The optimization is obtained by searching for the minimum value of the sum of the squared errors (SSE).

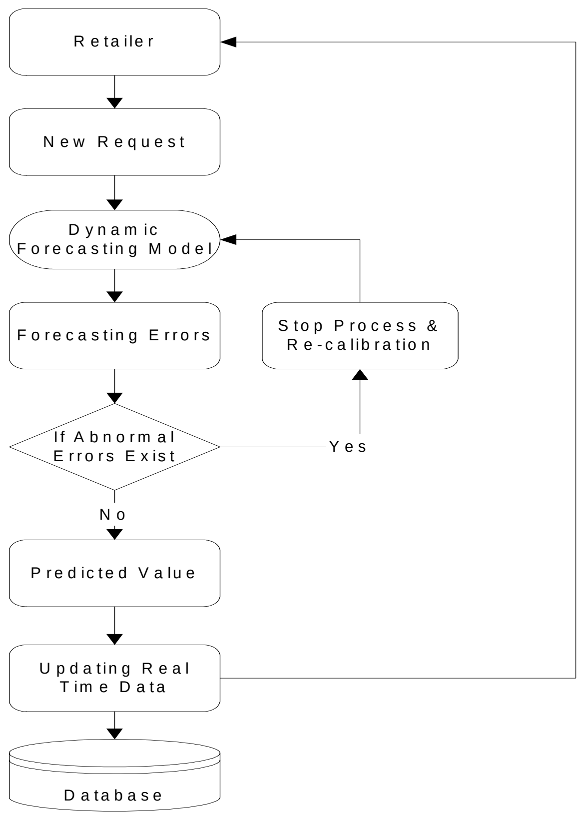


Figure 3.9 - A schematic diagram of the dynamic forecasting model

3.6 Knowledge-based Systems

Knowledge-based Systems (KBS) transfer information into knowledge, which is retained within the organization. For example, the demand trend indicates the number of sales orders changed during a particular period. With such information, an experienced member of staff would know why demand has suddenly increased and how to replenish products to meet the new demand. However, it is difficult to arrange sharing of the knowledge of experienced staff with new or less experienced staff. The KBS provides an important means for providing suggestions and guidance for new staff so they will understand the problems and learn how to solve them. So, even if the experts are not available in the company, daily operations are still able to continue.

In the KBSP, the CBR model is used to diffuse knowledge to each staff member so that a learning environment is provided for enhancing competitiveness. The CBR model consists of case retrieval, reuse, revision and retention. Case retrieval is a primary process that is done by using features of new cases as references for similarity matching. By extracting solutions from some retrieved cases, they are reused for working in a new situation. The proposed solution is sometimes obtained by a process of case revision, which is used to compare and contrast the new situation with previous cases. Once the solutions are adapted to fit the new case, the revised solutions are retained as new knowledge.

The KR serves as a database to store past successful cases, such as transaction records, inventory level, lead time, information on customers and suppliers. In each previous case, case indices are used to retrieve and reuse cases effectively. Such indices include product model, inventory level and safety stock. When an order is placed by a customer, previous relevant cases are retrieved from the KR.

The power spectrum patterns generated in the simulation model are extracted and used for comparison with those generated from a new demand pattern. As data are transferred by the power spectrum analysis into visible information, staff members are able to classify and analyse any similarities efficiently. Once the similarity analysis is completed, the relevant parameter settings are obtained and reused as references. The suggested solutions are then provided, revised, and retained in the KR. The retained cases become new knowledge for future reuse.

Players collaborate with each other and attempt to mimic the operations of real cases through the simulation platform. During the simulation process, implicit knowledge such as their thinking, problem solving skills and experience is transferred to explicit knowledge in the form of words such as historical logs, remarks and reasons, which are stored in the KR of the KBSP. Relevant cases, including explicit knowledge, are extracted and reused in the KBSP. Thus, the knowledge transfer mechanism starts with the retrieval of implicit knowledge by the players and then the knowledge is transferred to explicit knowledge by inputting it via the front-end interface of the simulation platform. The result is retained in the KR for reuse in the future.

3.7 Summary

In this chapter, the traditional approach of supply chain operations is first compared with the KBSP. It found that the problems of the conventional approach are inaccurate sales forecasts, keeping high inventory, insufficient trust between parties and long lead time for product replenishment. In contrast, the KBSP reduces unpredictable risks, gains experience in a simulated environment, acquires and diffuses knowledge as well as getting real time responses from other parties.

The simulation module is divided into three scenarios: one-player design mode, two-player design mode and multi-player design mode. In the one-player design mode, a feasibility scenario of the VMI strategy and that of the addition of customers are described. This addresses the limitations of traditional simulation games, such as the beer game, in supply chain management. A number of algorithms are used to build the KBSP, including DFT and FFT Algorithms (for analysis of an ordering pattern), MLMS algorithm (in the dynamic forecasting model) and the CBR model.

Chapter 4 Implementation of the KBSP

In this chapter, the implementation of a prototype is described. The prototype of the KBSP is based on the design in chapter 3. Two main applications are provided in the prototype: (1) simulation and (2) knowledge generation and retention.

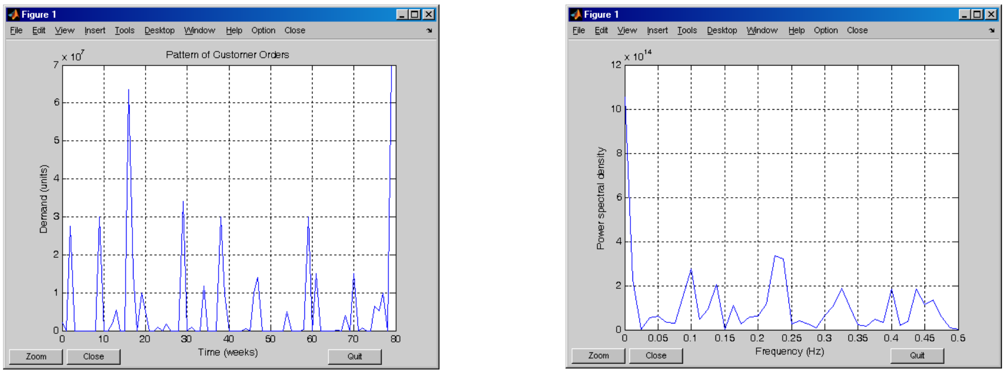
4.1 Simulation of the KBSP

In this section, several components are built into the KBSP. Spectrum analysis of the ordering pattern is firstly introduced. It is used to analyse demand patterns. By comparing similar patterns, knowledge of relevant cases is retrieved and stored for future use. Following this topic, the use of forecasting in a VMI strategy, simulation of ordering and replenishment processes and simulation of VMI strategy are further described.

4.1.1 Spectrum Analysis on Ordering Pattern

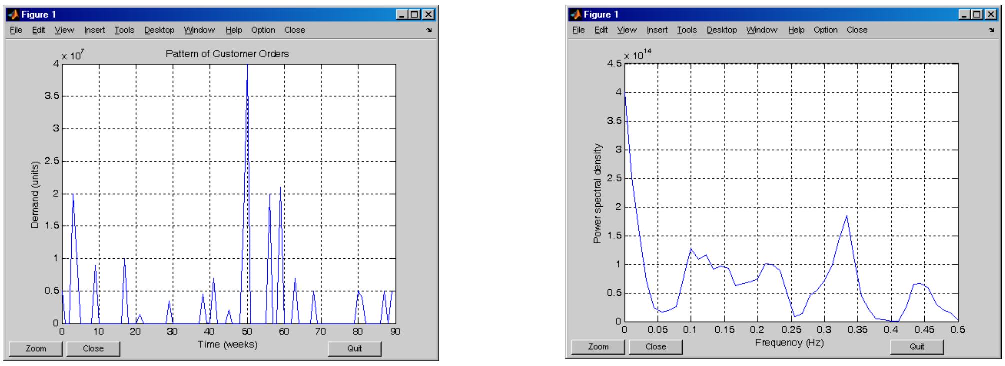
In supply chain operations, customers have their own ordering patterns. Some customers prefer to order a larger or smaller quantity of the product than they need. This results in frequent changes in demand and makes it difficult for retailers to forecast and plan delivery schedules. The data are collected, analysed, and demand patterns are generated by using a large number of transactions. However, it does not mean that the customer behaviour cannot be captured, even if they have such demand curves. However, such a demand pattern usually changes over a period of time, and the comparison and classification of its pattern is not easily made by a supplier.

In section 3.4, the methodology of the spectrum analysis module has been discussed. In this section, results of spectrum analysis show how a pattern is generated for a given demand. Two sets of results have been obtained by spectrum analysis. As shown in Figure 4.1(a) and Figure 4.2(a), the first set of results compares two different customers who order the same type of products for the same period. The second set of results compares ordering patterns for the same customers who order the same type of products at different times, as shown in Figure 4.3(a), Figure 4.4(a) and Figure 4.5(a).



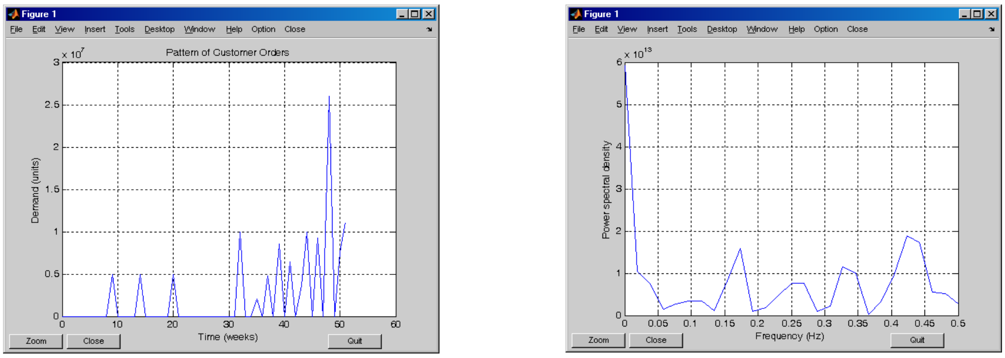
(a) **(b)**

Figure 4.1 - (a) Ordering pattern of customer A with product A and (b) the result of its spectrum analysis



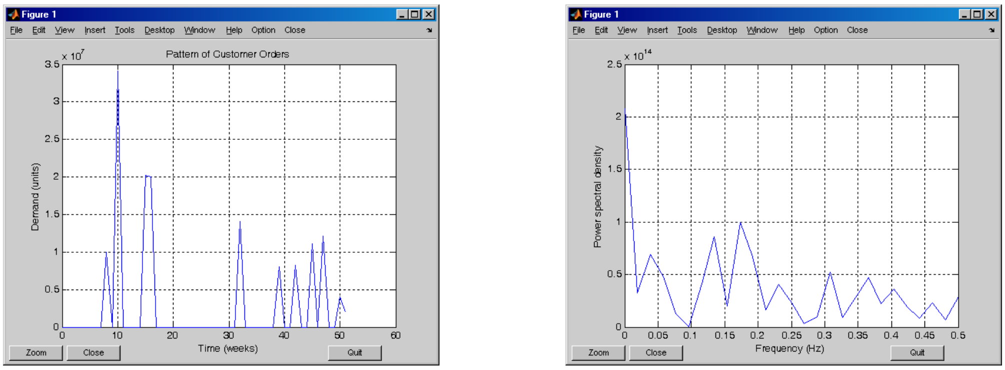
(a) **(b)**

Figure 4.2 - (a) Ordering pattern of customer B with product A and (b) the result of its spectrum analysis



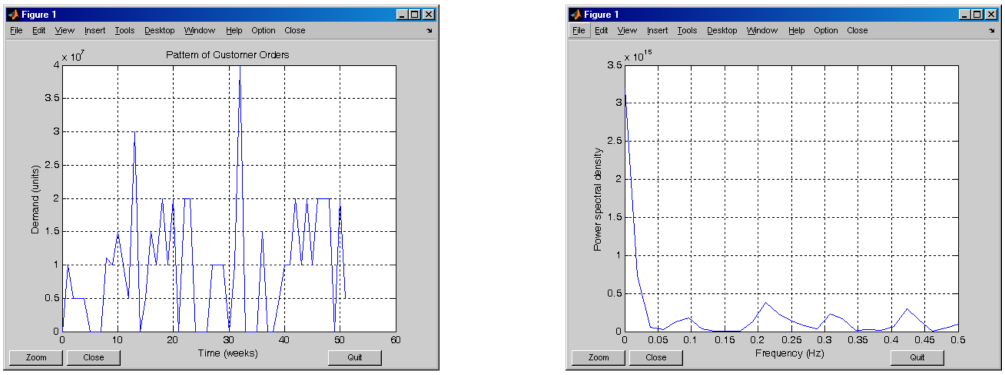
(a) **(b)**

Figure 4.3 - (a) Customer demand pattern of new case and (b) the result after spectrum analysis



(a) **(b)**

Figure 4.4 - (a) Customer demand pattern of retrieval case 1 (one of the successful cases) and (b) the result after spectrum analysis



(a) **(b)**

Figure 4.5 - (a) Customer demand pattern of retrieval case 2 (another successful case) and (b) the result after spectrum analysis

According to the results, after spectrum analysis, each peak of the spectrum represents the frequency of customer demand. For example, in Figure 4.1(b), the peak is located at a frequency of 0.1 Hz which is the frequency of customer demand (x-axis). This means the customer places a first order and then a second order is placed after a period of 10 weeks, and is calculated from Equation (4.1),

$$T = \frac{1}{f} \tag{4.1}$$

where T is at period for each order and f is frequency of customer order.

If orders are repeated more frequently, the peak is sharper. The power spectrum density, that is the peak level, is directly proportional to the order frequency. Thus, the ordering behaviour of each customer is shown in each pattern of the spectrum.

The results show that different ordering patterns in Figure 4.1(a) and Figure 4.2(a) are transformed into similar patterns in the spectrum analysis in Figure 4.1(b) and Figure 4.2(b). It allows the supplier to understand the customer ordering pattern. The use of spectrum analysis is to transform data to a spectrum which is collected for further reuse in the KR.

For a new customer request, a new demand pattern, as shown in Figure 4.3(a), is generated in which the characteristics are not easily extracted and are compared with other cases. Based on the results of spectrum analysis as shown in Figure 4.3(b), Figure 4.4(b) and Figure 4.5(b), the spectrum pattern of Figure 4.3(b) is relatively similar to that of Figure 4.4(b) when they are compared with Figure 4.5(b) by manual comparison. It indicates that the pattern of customer orders in the new case is relatively similar to that in case 1, which is one of the successful cases retained in

KR, as shown in Figure 4.4(b). Thus, the spectrum pattern of the relevant case 1 is extracted and transferred to the KBS for further reuse.

4.1.2 Forecasting in a VMI Strategy

The dynamic forecasting model, mentioned in section 3.4, is built in the KBSP for simulating a forecasting process in VMI. In this model, the adaptive time-series algorithm is used to adapt the predicted value to the actual value at each period. In a VMI strategy, a supplier needs to manage the retailer's inventory and to replenish products on time. A retailer sends sales information to a supplier daily or weekly. A supplier uses the information to provide a forecast and to predict demand. If the forecast is accurate, a supplier is able to manage the inventory well. Once the supplier receives the actual demand from a retailer, a dynamic forecast model is used to predict the next period of demand. The main function is for a supplier to plan his stock well and prevent a stock-out situation from occurring.

4.1.3 Simulation of VMI Strategy

A retailer collects sales information and inventory status in real-time. The retailer sends the information to the supplier daily or weekly. The supplier receives it through the Internet, and is responsible for providing a sales forecast and for replenishing the retailer's stock of products. The VMI program is simulated from the stock check by a supplier, who arranges for replenishment and for product delivery.

The simulation of the VMI strategy is divided into two stages. The first is setting an agreement between the supplier and the retailer. Initial terms of an agreement are generated from optimization in the simulation model, as discussed in section 3.3. The optimal values serve as references for a supplier to start up the negotiation in the VMI simulation. The VMI agreement is shown in Figure 4.6. In the process, both parties negotiate the terms of the agreement, such as the annual ordering size, batch size and lead time. When the terms are confirmed, the simulation is initiated at the second stage.

In Figure 4.7, the process of simulation in a VMI strategy is shown. The supplier frequently checks demand and forecast data of the retailer. After a period of time, actual demand is shown in the history log area. The supplier delivers products to the customers based on the actual demand of customers and their expected delivery date. If the inventory drops to the safety stock level, the supplier makes a decision on replenishment from its upstream supplier. In the simulation, safety stock and reorder level are provided as suggestions. A final decision of how to do the replenishment is based on the experience of the supplier. After running transactions for one year, the results of the simulation, which are inventory level, fill rate and stock-out rate, are provided. Some of the potential benefits realized from the implementation of the KBSP are:

(i) providing a simulation environment for staff to gain experience for handling uncertainty;

(ii) predicting values obtained from various strategies;

(iii) reducing cost and risk for re-designing supply chain processes.

In some cases, experienced staff are able to keep inventory at a low level based on their personal knowledge. For instance, when to replenish products, how much will be ordered and which retailers will be first served depend on the experience and skill of the operators in handling fluctuating demand. The reasons for adjusting the parameters and for making decisions on issues of ordering and replenishment depend on personal know-how and this is valuable to an organization. The staff obtain explicit knowledge through interaction in the simulation.

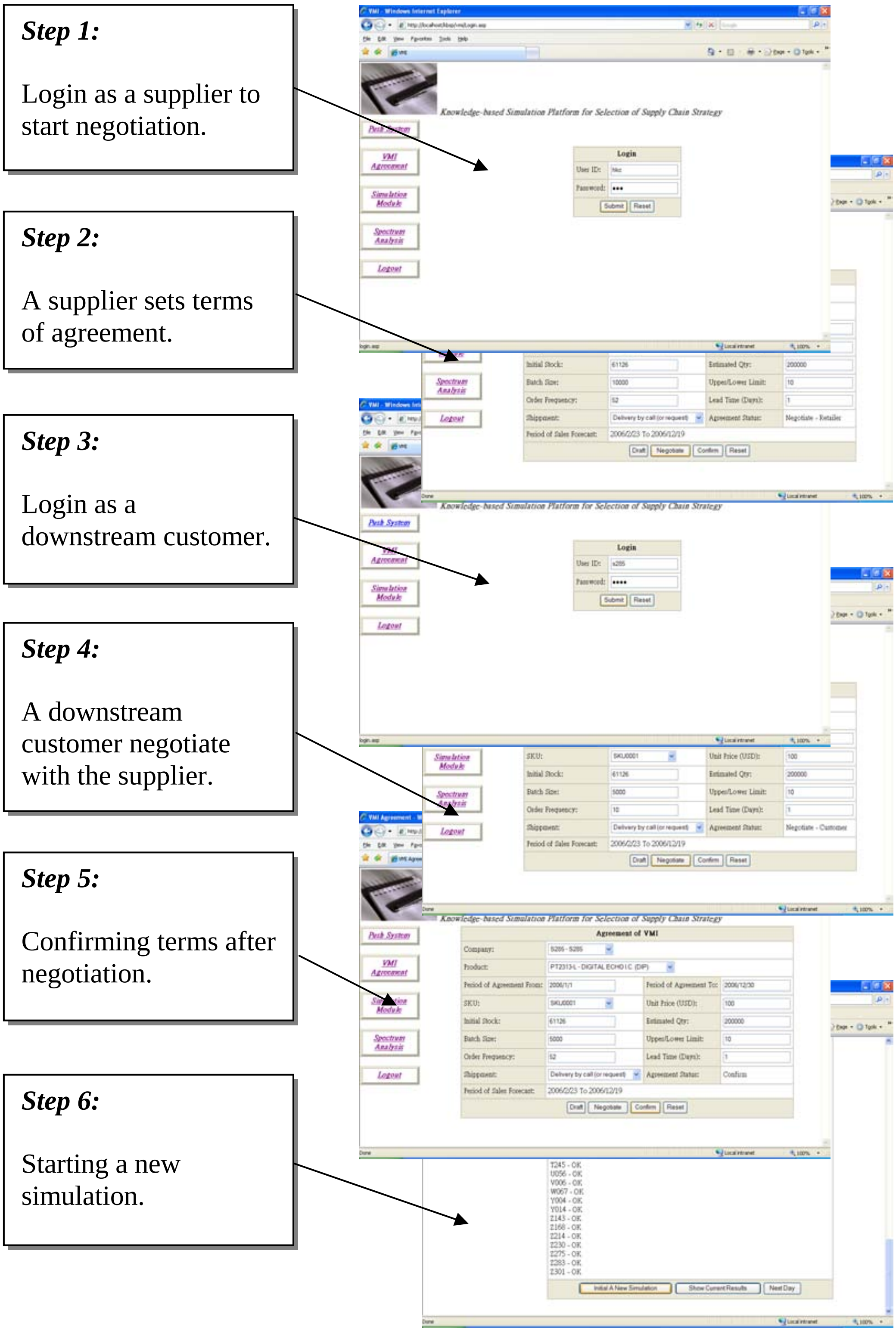


Figure 4.6 - Negotiation of terms of a VMI agreement

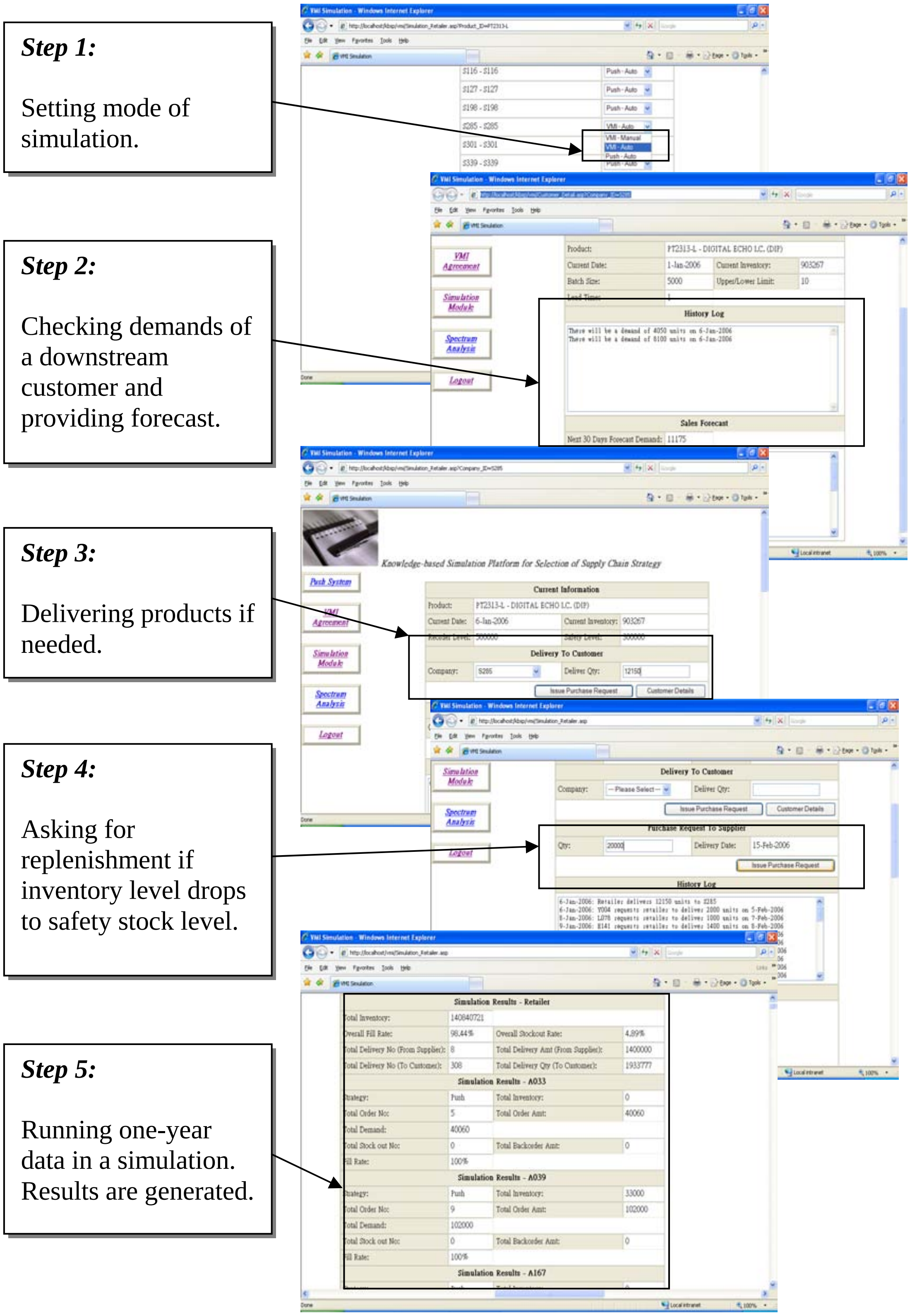


Figure 4.7 - A process of simulation with a VMI strategy

4.2 Knowledge Retention in the KBSP

A learning environment is built for staff to share, diffuse, and acquire knowledge. The KBS is developed, as shown in Figure 4.8. Through the collaborative simulation system, tacit knowledge is converted to explicit knowledge. Staff are able to have a better understanding of the ordering behaviour of their customers and thus effectively collaborate with their downstream customers and upstream suppliers.

As it is confusing to try and analyse dynamic demand patterns and difficult to make decisions, the relevant spectrum patterns are extracted. Comparing new cases with previous relevant cases, a similarity analysis can be undertaken. The recommendation is obtained from remarks provided by well-experienced staff, and are given to the staff for adoption. These remarks are experiences or skills for solving problems of fluctuating demand. Implicit knowledge, such as experience and skill, is transferred to explicit knowledge, such as remarks. If the case is reused frequently, it will be validated as new knowledge and retained in the KR for future reuse.

Once the KBSP is implemented, the company is expected to foresee which parts are frequently ordered during a certain period, what quantities of each kind of goods are required by the customer and when the orders are confirmed. This is an effective way to plan how to distribute limited quantities of products among several customers. At operational levels, staff are able to gain working experience in a short period of time from the simulation game, for example, on how to manage urgent orders. More practice is provided for them before they face real events, so they can gain more experience before they have to deal with real transactions.

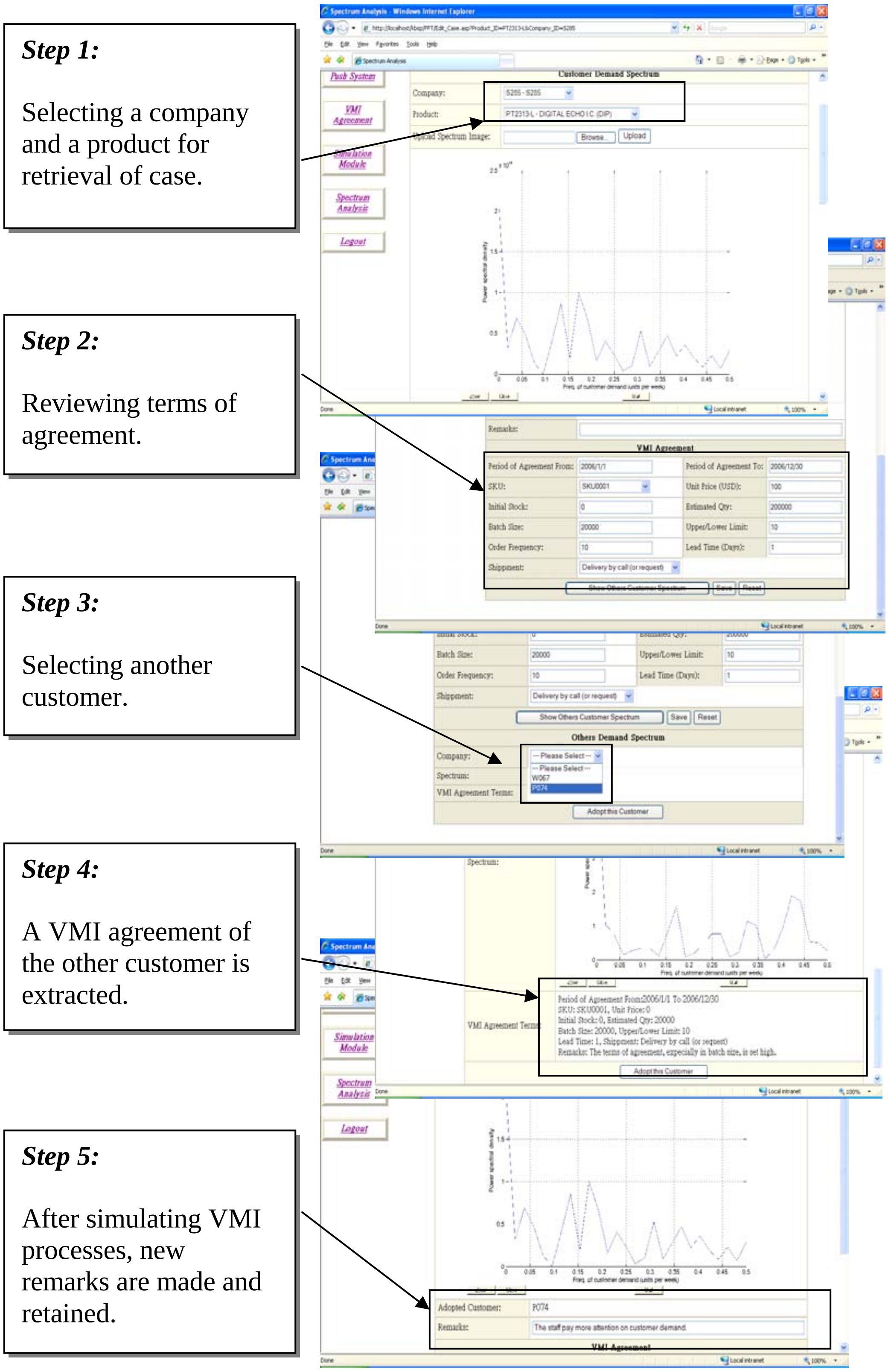


Figure 4.8 - Implementation of KBS

4.3 Summary

The goal of KBSP is to predict the outcomes of various supply chain strategies, such as the push system and the VMI strategy in the simulation mode, so as to reduce traps or risks during real events. Thus, the implementation focuses on several functions: spectrum analysis of customer behaviour, forecasts in the VMI strategy and knowledge retention.

A simulation starts from the traditional ordering and replenishment processes. Retailers and suppliers are involved in the simulation so they learn how to manage transactions. As the supplier is responsible for providing forecasts for their retailers under a VMI strategy, a dynamic forecast system is provided by the supplier. By collecting customer orders from transactions, customer behaviour can be analysed and transformed to spectrum patterns, which are effective for making comparisons. The knowledge used in the retrieved case is extracted to provide possible solutions and recommendations corresponding to the new cases. With a VMI strategy, many supply chain operations, such as scheduling and planning, forecasting and inventory control, are shifted to a supplier.

Chapter 5 Case Study and Discussion

5.1 Company Background

This chapter contains an account of the trial implementation of the KBSP in a selected reference site, the Angus Electronics Company Ltd. It is a trading company headquartered in the Hong Kong SAR, China. Angus Electronics was established in 1990 and produces mainly audio products, security products, ultrasonic sensors and home appliances. Most of their major vendors are well established Taiwan Consumer IC Design Houses or Manufacturers. While 60% of its market is in Hong Kong, 30% is in the China Market and the rest is overseas.

5.2 Challenges to Current Practices

One of Angus' current challenges is the uncertainty of demand for its items. It is difficult to provide an accurate forecast for sudden demand change based on urgent orders from customers. High inventory is kept to safeguard against such uncertainty. In a typical company, such as Angus, more than half of all I.C. chips used in the company products are for audio or video products, as shown in Figure 5.1. The demand for I.C. chips increases if new models of audio or video product are put onto the market. A long lead time, from four to six weeks, is needed for product replenishment from its supplier. This reduces customer responsiveness and customer satisfaction.

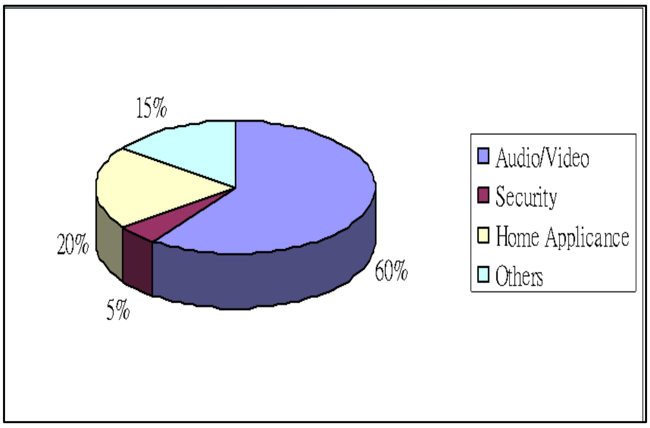


Figure 5.1 - Distribution of sales in Angus Company

5.3 Results of Simulation

Because of the challenges involved, a VMI strategy is recommended for Angus Electronics Company Ltd. When redesigning the supply chain strategy, the return on the investment and the risks involved are the main causes for concern in the company. The KBSP is proposed for the parties so they can mimic and predict the impacts of their strategy on the supply chain. The evaluation of the performance of the KBSP is based on three indicators: inventory level, fill rate and stock-out rate. Items selected for the trial run in the prototype are those that are frequently ordered by customers. A set of optimal parameters and results are simulated from three scenarios, which are the use of the push system for all retailers; one retailer adopting a VMI strategy; and two retailers adopting such a strategy. The results of each scenario are discussed in the following sections.

In the case study, a trial implementation in Angus was completed and the processes are shown in Figure 5.2. Before setting up an agreement, the supplier extracted the customer's demand from a database. These demands were transformed to a series of spectra by use of a spectrum analysis model. Compared with similar patterns of spectra, the supplier retrieved the most similar case. The parameters of the retrieved case were reused in the new case. As customer demands in each case are different, adapted parameters of the agreement were revised. It was suggested that the revised value is used in the new agreement. The simulation was re-run using the revised parameters.

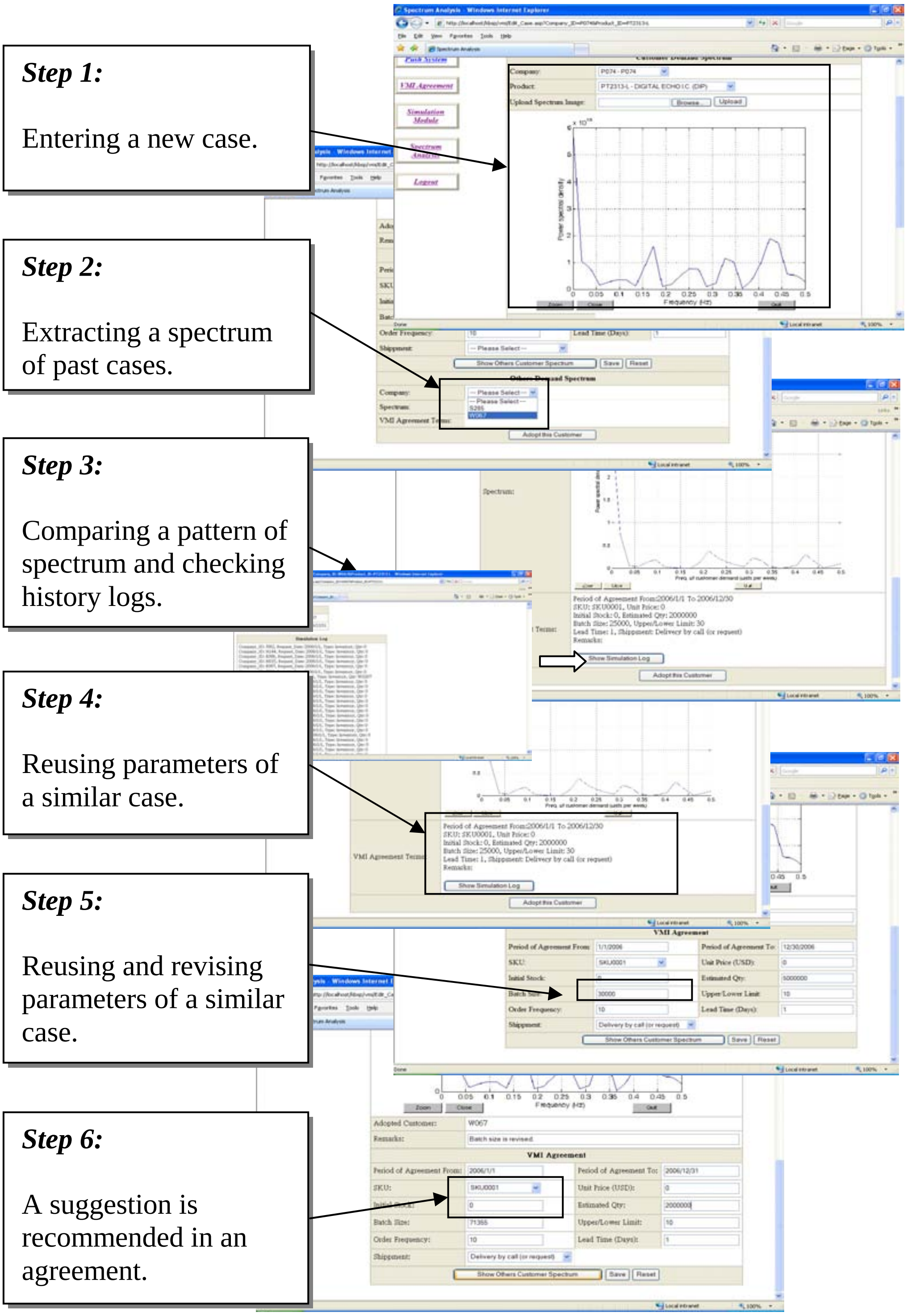


Figure 5.2 - Results of trial implementation in Angus

During the new simulation, the transaction events were logged. This was a process of knowledge capture. The player took action to respond to each event, which became new knowledge. Compared with the results of a traditional approach, this case is better. The set of parameters and the log record of the successful case were retained in the KR. The processes of knowledge capture and knowledge retention are shown in Figure 5.3.

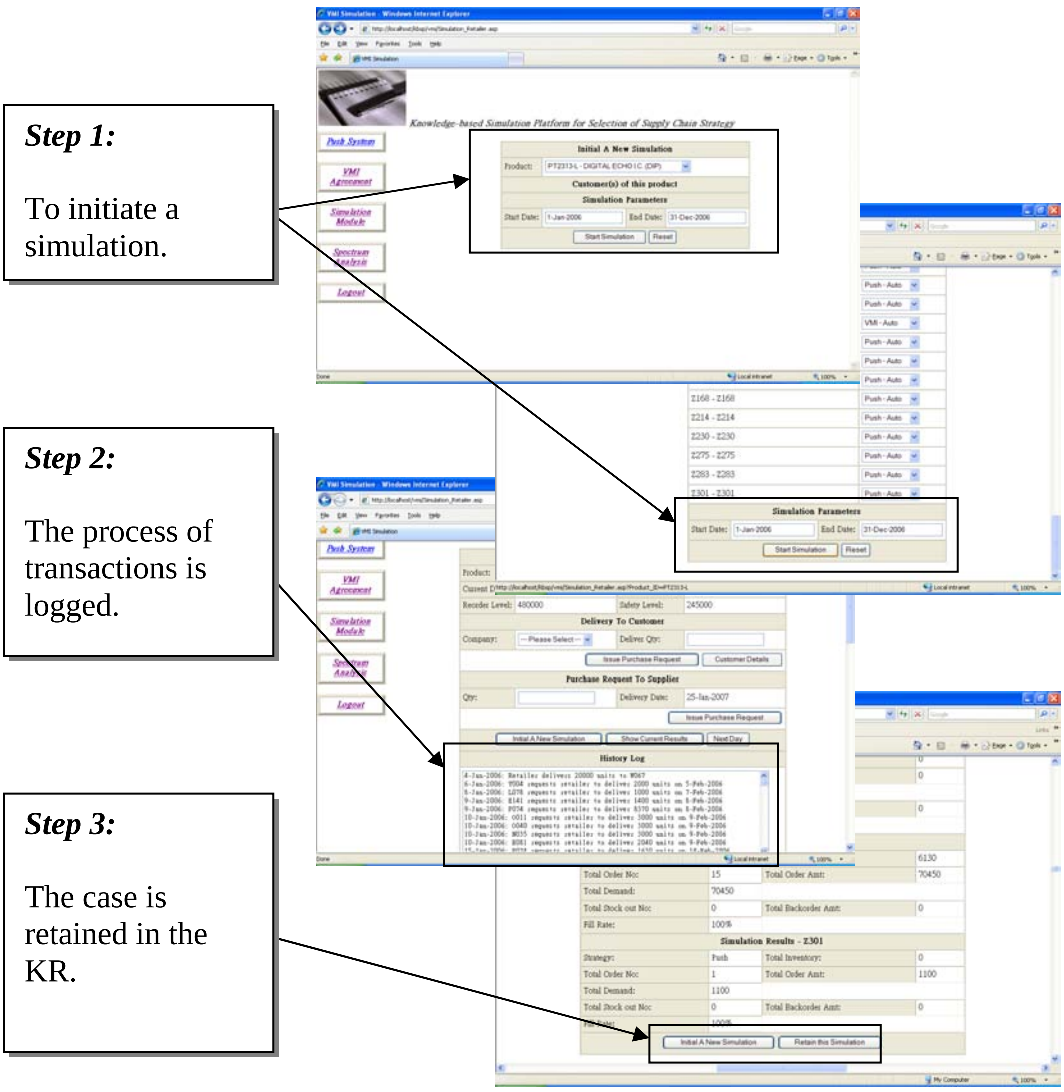


Figure 5.3 - Processes of knowledge capture and knowledge retention

5.3.1 Use of Push System for All Retailers

The scenario consists of retailers only, using the push system in which all the system's processes are simulated. The period of simulation lasts for one year. In such a scenario, the lead time used in the process of replenishment for retailers is 28 days. The results show that the fill rate and the stock-out rate of retailer W067 are 92% and 5.41% respectively. These results are used as a baseline of performance measurement for the supplier to compare its results after changing to a VMI strategy.

5.3.2 One Retailer with a VMI strategy

In the scenario of one retailer with a VMI strategy, the retailer decides to optimize the terms of reference for the VMI agreement. After setting an upper limit and a lower limit for each parameter, the optimization process is initiated and the optimal results are generated at the end of the simulation. Compared with the result of the push system scenario, the lead time is reduced from 28 days to 1 day. The result shows that the inventory level of the supplier is slightly increased by 3.01%. However, a 100% fill rate and no stock-rate for retailer W067 are achieved. It can be inferred from this that it is feasible for the retailer to change the strategy from a push system strategy to a VMI strategy, as better results are indicated in the KBSP simulation.

5.3.3 Two Retailers with a VMI strategy

Two retailers were selected to run the simulation with a VMI strategy. In the scenario, the fill rate and stock-out rate of retailer W067 are still kept at 100% and 0% respectively. Compared with the result of one retailer with a VMI strategy, the fill rate of retailer P074 is increased from 93% to 100%. The stock-out rate is reduced from 6.67% to 0%. This shows that the addition of a retailer with the VMI strategy would bring more benefits to the supplier and the retailer.

5.3.4 Results of a VMI Strategy in an Electrical Company

In the scenario of using the VMI strategy in an electrical company, a set of results has been generated, as shown in Figure 5.4 and Figure 5.5. The purpose is to measure how the staff handle replenishment processes with the company's various retailers, under the same circumstances.

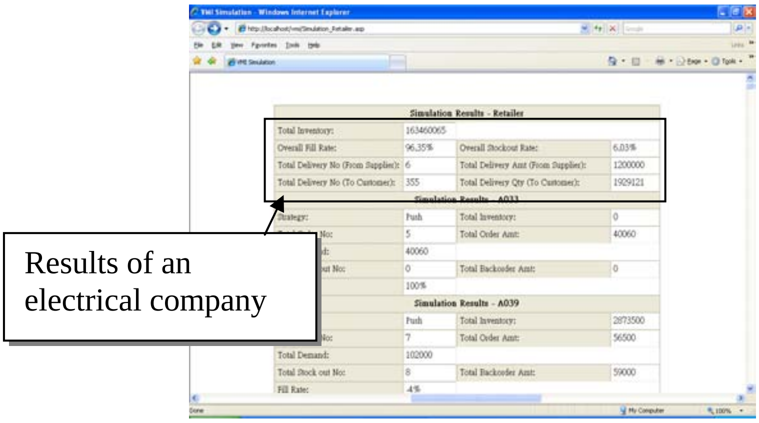


Figure 5.4 - A result of simulation in Angus

The results reflect the direct impact of the VMI strategy in a real situation. A trial implementation was undertaken via a web-based platform. The result of fill rate and stock-out rate counted by retailer W067 are 98% and 2.78% respectively. After a series of processes, the outcome of the simulation reflects a real situation and provides a chance for the company to consider whether the new strategy should be adopted or not.

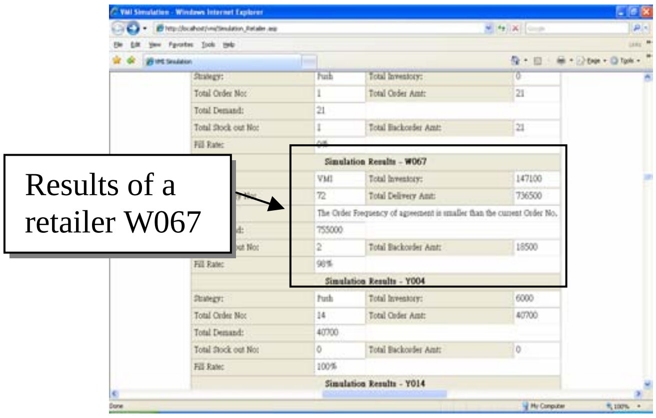


Figure 5.5 - The result of a simulation for one of the retailers

5.3.5 Verification of Optimization in the KBSP

As discussed in chapter 3, the process of optimization is run within a boundary for each parameter, such as safety stock and reorder level. Its objective is to find a set of parameters with the lowest inventory level for a retailer, the lowest stock-out rate and the highest fill rate. Acceptable levels of stock-out rate and fill rate are set by a supplier. The process for optimization, starting from searching for the local optimal value, is to find a set of parameters to fulfil the requirements of the lowest inventory level, the lowest stock-out rate and the highest fill rate. If more than one solution results, the first set of values is selected as the local optimal value. The process of optimization is verified by a case study in a trading company. The optimal result generated in the KBSP for the product PT 2313-L Digital ECHO I.C. (Dip) for retailer W067, is 100% fill rate and 0% stock-out rate. Compared with these, the values obtained from a trial run in the company are 98% fill rate and 2.78% stock-out rate. As the results from the case are close to that of the KBSP, the parameters are verified as the local optimal solution under a case-based optimization environment.

5.4 Discussion

The KBSP prototype is evaluated through a series of functionality tests and satisfactory results are obtained. The functionalities of the KBSP are divided into five areas for discussion: (1) the web-based simulation platform, (2) the knowledge acquisition, (3) the analysis of customer ordering behaviour for replenishment, (4) difficulties in a trial run and (5) limitations of the KBSP. Details follow:

5.4.1 Web-based Simulation Platform

Instead of traditional simulation tools, the KBSP has been built in a web-based environment, so it is more convenient for suppliers, retailers and customers to simulate supply chain operations. In the platform, customers are able to place orders anytime and anywhere through the Internet, even if they are located overseas. The supplier needs to monitor inventory level frequently, asking for replenishment when the inventory is lower than the safety stock level. An electronic delivery notice can be sent by suppliers through the KBSP instead of using traditional channels, such as fax or mail, When running a series of supply chain operations, problems faced by each party can be identified.

5.4.2 Knowledge Acquisition

The KBSP acts as a knowledge-based system for the assimilation of knowledge in supply chain operations. Knowledge is generated and retained in the organization. For example, suggestions on how to manage customer demand and when to take action on replenishment are entered into the system. This is similar to a knowledge transfer process which converts implicit knowledge to explicit knowledge.

The significance of the KBSP is to acquire the knowledge that is embedded in the minds of well-experienced staff, and to retain it systematically. Remarks provided in similar cases provide very valuable information to help others to cope with new problems.

5.4.3 Analysis of Ordering Patterns

Customers have their own ordering behaviour. Some like to order many more products than they actually need; others prefer to buy products in quantities that just suit their immediate requirements. Through analysis of the spectrum, patterns are generated. From the results, staff are able to know the customer's behaviour and thus they are able to manage new transactions efficiently. When there is a new customer, the new spectrum pattern is compared with existing ones. Similar order handling techniques are reused with new cases for reduction in the level of risk.

5.4.4 Difficulties in a Trial Run

During the trial run of the KBSP in a selected reference site, one of the difficulties faced at the beginning was to extract relevant data from a great number of transactions. In some cases, duplicate records existing in the system are retrieved. It takes a long time to understand the problems of each special record. This reduces the efficiency of the data collection process.

Besides, the reliability of the KBSP greatly relies on human intervention in manipulating the transactions rather than sets of rules built into the simulation. It is assumed that well-experienced staff, who are knowledgeable on inventory management, are invited to test the system. However, not many knowledgeable workers are involved in the trial run. They usually give suggestions to their colleagues on how to do it well rather than doing it by themselves. This affects the reliability of the outcomes predicted at the end of the simulation.

5.4.5 Limitations of the KBSP

Adoption of a new supply chain strategy, from design planning and implementation, involves complicated processes. Defining a small scope for the redesigning strategy makes it more effective for SMEs to implement. In the present study, the scope is defined as the simulation of supply chain operations from a push strategy to a VMI strategy. This will result in an estimation of outcomes and knowledge capture, that assist SMEs in making decisions on redesigning strategies. In order to reduce the complexity in a supply chain network, the simulation design uses one supplier and multiple retailers. However, some limitations exist in the KBSP. This will be discussed under three headings: (1) the structure of supply chains, (2) types of supply chain strategies, (3) the mode of supply chain operations.

(i) The Structure of Supply Chains

In practice, a supply chain network is formed by a number of supply chains. A retailer in a supply chain not only collaborates with its upstream suppliers and downstream customers, but also cooperates with a retailer which functions in another supply chain that forms a complex and a widely diverse supply chain network. In the present study, the structure of a supply chain is limited to two tiers, involving a supplier and multiple retailers. Even though the proposed KBSP cannot simulate all operations in a supply chain, it provides a good starting point for two parties to consider and adopt a new supply chain strategy.

(ii) Types of Supply Chain Strategies

In the study, a scenario is defined in the simulation, which allows SMEs to change supply chain strategies from a traditional push strategy to a VMI strategy. Other traditional strategies such as a pull strategy and a pricing strategy are not included. According to Simchi-Levi et al. (2003), some supply chain contracts, such as buy-back contracts, revenue-sharing contracts, quantity-flexibility contracts and sales rebate contracts, are used in supply chain management. However, these are not considered in the KBSP.

In the simulated replenishment process of the KBSP, products are delivered as a whole. That means partial delivery is not allowed. For bulk purchase, discount is not allowed, and is one of assumptions made in the optimization. However, both a supplier and a retailer in their real situations negotiate unit price of products and sales order in a web-based simulation platform. Terms, which are batch size and its upper and lower limits, unit price, size of an order, lead time and a mode of product delivery, are decided at the end of simulation.

(iii) The Mode of Supply Chain Operations

In the KBSP, the processes of simulation mimic a business mode of a trading company with a VMI strategy. In such kind of companies, operational activities range from processes of product replenishment, product repackage to product delivery. Compared with processes conducted in a manufacturing company, operations in the trading company are less complex that the production scheduling and assembly are not included. This is one of limitations that the mode of supply chain operations in the KBSP is to simulate processes in a trading company only.

5.5 Summary

In this study, the KBSP has had a trial run in the reference company. Compared with the traditional approach, the results, using the VMI strategy, for three indicators (inventory level, stock-out rate and fill rate), is significantly improved. With this approach, the responsibility for replenishment is shifted from a retailer to his supplier. It is beneficial for the company to make the operation of the supply chain processes as smooth as possible and reduce the lead time significantly. Once the strategy is implemented successfully, the potential benefits gained from the organization include reduction in inventory level, lead time and costs, which in turn leads to increase in profits and fill rate.

The KBSP provides a web-based simulation platform for supply chain parties to mimic their supply chain operations anytime and anywhere via the Internet. The costs involved in redesigning the supply chain strategy from a traditional one to VMI are minimized. In addition, one of the main benefits is to retain new knowledge in the system for future reuse. Through the simulation processes, the knowledge possessed by highly experienced staff is retained in the KR. Techniques for handling customer requirements are the kinds of knowledge obtained from the KBSP. These are critical factors for building good relationships in the supply chain.

Chapter 6 Conclusions

In today's rapidly changing business environment, industry is operating in an environment that is characterized by intense global competition. This calls for strategic management of supply chain activities. It will be beneficial to an enterprise if there is a predictive analytical tool, which provides them with the knowledge of how to formulate a good supply chain strategy, with appropriate guidance for establishing such strategies in order to gain a competitive edge in the global market. Thus, the KBSP plays an essential role in bringing effective changes into an organization which will change its supply chain strategy from a traditional approach to a VMI strategy. Based on supply chain strategies, a prototype of the KBSP is proposed.

The KBSP has four components: (1) the simulation model, (2) the spectrum analysis module, (3) the dynamic forecast model, and (4) the CBR model. The simulation model is used to mimic the effect of downstream and upstream operations on the effectiveness of the supply chain. Thus, an appropriate supply chain strategy can be derived from the simulation results. A dynamic forecast model can be used by suppliers to predict customer demands under the VMI. The CBR model is used to capture tacit and explicit knowledge during the processes in a simulation. It also provides a web-based simulation environment where multiple supply chain parties can join, share and collaborate in order to gain benefits for all parties involved in the supply chain.

The simulation model is divided into three design modes, which are: (1) one-player mode, (2) two-player mode, and (3) multi-player mode. Angus Electronics Company Ltd. is selected as a reference site for a trial run. The results of fill rate and stock-out rate have been generated by staff in Angus, and counted by retailer W067, are 98% and 2.78%. Lead time is reduced from 28 days to 1 day. When the strategy is implemented successfully, benefits gained by the company include reduction in unfilled rate and stock-out rate, shorter lead time and a lower inventory level, which lead to cost saving and increase in profits.

With the successful implementation of the KBSP, it is possible to predict the outcomes of a real situation. As the supply chain processes are managed in a simulated environment, the risks can be evaluated and reduced before implementation. Building the infrastructure to manage supply chain operations on the web will ultimately give a way to a more mature approach to enhance the competitiveness of an enterprise. It is clear that the KBSP will play an important role in supply chain management in future.

Chapter 7 Suggestions for Future Research

In this research study, the KBSP has been built and a trial run was conducted at a reference site. Some areas have not been covered in this research work, and three suggestions are recommended for further work:

7.1 Implementation on Manufacturing Sites

The simulation platform is designed for multi-players and focuses only on three levels of supply chain parties, namely the customer, retailer and supplier. It is suitable to fit into the business mode of a trading company. It is recommended that the scope is expanded and future research is undertaken in the manufacturing industry area. The additional considerations for the manufacturing industry include material resources planning, production scheduling and the structure of bills of materials (BOM).

In manufacturing firms, products, such as semiconductor equipment, are composed of thousands of components. When one of the critical components is out of stock, the production will break down, and the lead time will increase. When the KBSP is implemented in manufacturing sites, the effect of the whole supply chain will be simulated. The risks, such as demand uncertainties, would be identified and reduced. Besides, it is proposed that the KBSP is linked with enterprise resources planning (ERP) systems to provide more realistic environment for different parties to conduct simulation.

It is suggested that the enhanced KBSP will be trial implemented in a manufacturer. One of the current challenges is the uncertainty of demand for the parts. Customization of products is provided to customers in order to increase customer satisfaction and loyalty. Even if two customers have made orders for the same model of semiconductor equipment, the required parts may be different due to customization. It is difficult to provide an accurate forecast of the demand based on sales orders from customers. A high level of inventory would need to be kept to cope with such uncertainty. This is another challenge for the inventory management of some critical products.

The lead time is very long for the replenishment of some parts which are provided by very few vendors. If the demand for the parts increases suddenly and they are out of stock, this leads to delay in the production processes and the lead time becomes very long. The result is that the production schedule may not be followed and the customer requirement will not be met at the end. When the implementation of the KBSP is successful, benefits can be realized in the elimination of errors due to the lack of stock, and in an increase in the efficiency of the production process.

7.2 Customization of Other Supply Chain Strategy

In the current stage, only two supply chain strategies are considered in the simulation platform. They are the push system and the VMI. In practice, other supply chain strategies, such as a pricing strategy and an outsourcing strategy, are used in some companies. In fact, business processes in the supply chain are moving so quickly that more advanced supply chain strategies may soon evolve which have not been covered in this simulation platform. Based on the theory and know-how established in the present study, the KBSP should be further developed to include new supply chain strategies so as to increase the level of customization.

7.3 Integration with Other Commercially Available ERP Systems

In a supply chain, ERP systems are widely used to handle financial issues, human resources, purchasing and manufacturing. Each SME may have an individual ERP system to process its daily operations. To provide good communication and fast responsiveness between two parties, one suggestion is to enhance the capability of the KBSP by integrating it with an ERP system. This system allows the records to be transmitted to other parties in a fast way. After integrating the KBSP with the ERP system, more enterprises can be linked and a supply chain network can then be formed, and the effects of a networked supply chain will be simulated. Thus, it would be valuable to have further research and development on the integration of the KBSP and ERP systems.

7.4 Increasing the Variety of the Scenarios in the KBSP

In the present study, the terms of a VMI agreement are limited to batch size, lead time, ordering size, unit price and the mode of product delivery. Even though the agreement has not fully fulfilled various scenarios in practice, it provides a basic VMI agreement for two supply chain parties, who are interested in a joint venture, to negotiate terms at the preliminary stage of formulating a supply chain strategy.

For new products coming to the market, high demand for products leads to high price. In case of product shortage at the beginning stage, the price will increase to a higher level. The price of out-of-date products will decline. As the price of a product varies at different stages, different strategies on the pricing issue are adopted. In common practice for the VMI, retailers buy a bulk of products at each transaction so that they seek a quantity discount and partial delivery of products. It would be better if more scenarios are built into the platform for simulating various situations of supply chain operations.

7.5 Increasing Number of tiers in the Supply Chain

A supply chain network is vertically formed by a number of supply chains and is horizontally integrated with multiple supply chain parties at each tier. In the KBSP, only two-tier supply chain parties, which are retailers and a supplier, are involved. To make the simulation more realistic, there is a need for adding more tiers of suppliers in the platform. In this scenario, the supply of products from the supplier to the retailers is limited and it is highly dependent on how third-tier suppliers to distribute products. It is suggested that more supply chain strategies are added in the KBSP for simulating various scenarios of product replenishment from the third-tier suppliers.

References

Aamodt, A., & Plaza, E. (1994). Case-based reasoning: foundational issues. Methodological variations, and system approaches. *AI Communications, 7*(1), 39-59.

Ahn, H.J., Lee, H., & Part, S.J. (2003). A flexible agent system for change adaptation in supply chains. *Expert Systems with Applications, 25*(4), 603-618.

Allen, B.P. (1994). Case-based reasoning: Business applications. *Knowledge Engineering Systems, 37*(3), 40-42.

Anderson Jr, E.G., & Morrice, D.J. (2000). A simulation game for teaching services-oriented supply chain management: does information sharing help managers with service capacity decision?. *Production and Operations Management, 9*(1), 40-54.

Andraski, J.C. (1994). Foundations for successful continuous replenishment programs. *The International Journal of Logistics Management, 5*(1), 1-8.

Archibald, G., Karabakal, N., & Karlsson P. (1999). Supply chain vs supply chain: Using simulation to compete beyond the four walls. *Proceedings of the 31st conference on Winter simulation, USA, 2*, 1207-1212.

Arnold, J.R. Tony, & Chapman, Stephen N. (2004). *Introduction to Materials Management* (5th ed.). Upper Saddle River, N.J.: Pearson/Prentice Hall.

Bagchi, P.K., & Skjoett-Larsen, T. (2005). Supply Chain Integration: A European Survey. *The International Journal of Logistics Management, 16*(2), 275-294.

Barratt, M.A., & Oliveira, A. (2001). Exploring the experiences of collaborative planning initiatives. *International Journal of Logistics Management*, *31*(4), 266-289.

Bevilacqua, M., & Petroni, A. (2007). From Traditional Purchasing to Supplier Management: A Fuzzy Logic-based Approach to Supplier Selection. *International Journal of Logistics: Research and Applications, 5*(3), 235-255.

Bezdek, J. (1981). *Pattern Recognition with Fuzzy Objective Function Algorithms.* New York: Plenum Press.

Box, E. P., & Jenkins, M. (1976). *Time Series Analysis: Forecasting and Control.* San Francisco: Holden-Day.

Bui, T.N., & Moon, B.R. (1996). Genetic algorithm and graph partitioning. *IEEE Transactions on Computers, 45*(7), 841-855.

Cachon, G., & Fisher, M. (1997). Campbell soup's continuous replenishment program: Evaluation and enhanced inventory decision rules. *Production and Operations Management, 6*(3), 266-276.

Cares, J. (2006). Battle of the Networks. *Harvard Business Review, 84*(2), 40-41.

Carpenter, G.A., & Grossberg, S. (1988). The art of adaptive pattern recognition by a self-organizing neural. *IEEE Computer, 21*(3), 77-88.

Cetinkaya, S., & Lee, C.Y. (2000). Stock replenishment and shipment scheduling for vendor-managed inventory systems. *Management Science, 46*(2), 217-232.

Chang, Y.S., & Lee, J.K. (2004). Case-based modification for optimization agents: AGENT-OPT. *Decision Support Systems, 36*(4), 355-370.

Chatfield, D.C., Kim, J.G., Harrison, T.P., & Hayya, J.C. (2004). The bullwhip effect-impact of stochastic lead time, information quality, and information sharing: A simulation study. *Production & Operations Management, 13*(4), 340-353.

Chen, G., & Pham, T.T. (2001). *Introduction to Fuzzy Sets, Fuzzy Logic, and Fuzzy Control Systems.* Boca Raton, Florida: CRC Press.

Chen, L., & Gao, J. (2004). Knowledge acquisition system based-on multi-agent technology in ERP implementation assistant. *Proceedings of the 2004 International Conference on Intelligent Mechatronics and Automation, China,* 152-156.

Chern, C.-C., & Hsieh, J.-S. (2007). A heuristic algorithm for master planning that satisfies multiple objectives. *Computers & Operations Research, 34*(11), 3491-3513.

Cheung, C. F., Lee, W. B., Wang, W. M., Chu, K. F., & To, S. (2003). A multi-perspective knowledge-based system for customer service management. *Expert Systems with Applications, 24*(4), 455-468.

Choi, K.S., Dai, J.G., & Song, J.S. (2004). On measuring supplier performance under vendor-managed inventory programs in capacitated supply chains. *Manufacturing & Service Operations Management, 6*(1), 53-72.

Chopra, S., & Meindl, P. (2004). *Supply chain management strategy planning, and operation.* Upper Saddle River, N.J.: Pearson/Prentice Hall.

Cochran, W.T., Cooley, J., Favin, D., Helms, H., Kaenel, R., Lang, W., et al. (1967). What is the fast fourier transform. *IEEE Transactions on Audio and Electroacoustics, 15*(2), 45-55.

Cottrill, K. (1997). Reforging the supply chain. *The Journal of Business Strategy, 18*(6), 35-39.

Cox, E. (1992). Fuzzy fundamentals. *IEEE Spectrum, 29*(10), 58–61.

Cutting Decelle, A.F., Young, B.I., Das, B.P., Case, K., Rahimifard, S., Anumba, C.J., et al. (2007). A review of approaches to supply chain communications: From manufacturing to construction. *ITcon, 12*, 73-102.

Daniel, J.S.R., & Rajendran, C., (2005). A simulation-based genetic algorithm for inventory optimization in a serial supply chain. *International Transactions in Operational Research, 12*(1), 101-127.

Davenport, T.H., De Long, D.W., & Beers, M.C., (1998). Successful knowledge management projects. *Sloan Management Review, 39*(2), 43-57.

Davenport, T.H., & Prusak, L., (1998). *Working knowledge: How organizations manage what they know.* Boston, Mass: Harvard Business School Press.

Desai, P.S., Koenigsberg, O., & Purohit, D. (2007). The role of production lead time and demand uncertainty in marketing durable goods. *Management Science, 53*(1), 150-158.

Dhar, V., & Stein, R. (1997). *Seven methods for transforming corporate data into business intelligence.* Upper Saddle River, N.J.: Prentice Hall.

Ding, H., Benyoucef, L., & Xie, X. (2005). A simulation optimization methodology for supplier selection problem. *International Journal of Computer Integrated Manufacturing, 18*(2/3), 210-224.

Donk, D.P. (2001). Make to stock or make to order: The decoupling point in the food processing industries. *International Journal of Production Economics, 69*(3), 297-306.

Dong, Y., & Xu, K. (2002). A supply chain model of vendor managed inventory. *Transportation Research, 38*(2), 75-95.

Duan, Y., Edwards, J.S., & Robins, P.C. (1998). Experiences with exgame: An expert system for playing a competitive business game. *International Journal of Intelligent System in Accounting, Finance & Management, 7*(1), 1-19.

Durkin, J. (1994). *Expert systems: Design and development.* New York: Maxwell Macmillan International.

Dutta, S. (1997). Strategies for implementing knowledge-based systems. *IEEE Transactions on Engineering Management, 44*(1), 79-90.

Epstein J.M., & Axtell R. (1996). *Growing Artificial Societies - Social Science from the Bottom Up.* Cambridge: MIT Press.

Erraguntla, M., Benjamin, P.C., & Mayer, R.J. (1994). An architecture of a knowledge-based simulation engine. *Proceedings of the 26th conference on Winter simulation, USA,* 673-680.

Fox, M.S., Husain, N., McRoberts, M., & Reddy, Y.V. (1988). Knowledge based simulation: An artificial intelligence approach to system modeling and automating the simulation life cycle (Technical Report CMU-RI-TR-88-5). Pittsburgh PA: Carneie Mellon University, The Robotics Institute.

Frantti, T., & Mähönen, P. (2001). Fuzzy logic-based forecasting model. *Engineering Applications of Artificial Intelligence, 14*(2), 189-201.

Frohlich, M.T., & Westbrook, R. (2001). Arcs of integration: An international study of supply chain strategies. *Journal of Operations Management, 19*(2), 185–200.

Fung, S.H., Cheung, C.F., Lee, W.B., & Kwok, S.K. (2005). A virtual warehouse system for production logistics. *Production Planning & Control, 16*(6), 597-607.

Geary, S. Disney, S.M., & Towill, D.R. (2006). On bullwhip in supply chains - Historical review, present practice and expected future impact. *International Journal of Production Economics, 101*(1), 2-18.

Goldenberg, B.J. (2002). *CRM Automation.* Upper Saddle River, N.J.: Prentice Hall.

Golfarelli, M., Rizzi, S., & Cella, I. (2004). Beyond data warehousing: What's next in business intelligence?. *Proceedings of the 7th ACM International Workshop on Data Warehousing and OLAP, USA*, 1-6.

Gonzalez, L.R., Garcia, M.L., & Centeno, M.A. (1996) On-line knowledge-based simulation for FMS: A state of the art survey. *Proceedings of the 28th conference on Winter simulation, USA*, 1057-1061.

Grout, J.R. (1996). A model of incentive contracts for just-in-time delivery. *European Journal of Operational Research, 96*(1), 139 - 147.

Gunasekaran, A., & Ngai, E.W.T. (2007). Knowledge management in 21st century manufacturing. *International Journal of Production Research, 45*(11), 2391-2418.

Gunasekaran, A., Patel, C., & McGaughey, Ronald E. (2004). A framework for supply chain performance measurement. *International Journal of Production Economics, 87*(3), 333-347.

Hagan, M.T. (2002). *Neural network design.* Boulder, Colo.: University of Colorado at Boulder.

Harrell, C.R. & Hicks, D.A. (1998). Simulation software component architecture for simulation-based enterprise applications. *Proceedings of the 30th conference on Winter simulation, USA*, 1717-1721.

Harris, F. (1978). On the use of windows for harmonic analysis with the discrete fourier transform. *Proceedings of the IEEE, 66*(1), 51-83.

Harris, J.K., Swatman, P.M.C, & Kurnia, S. (1999). Efficient consumer response (ECR): A survey of the Australian grocery industry. *Supply Chain Management, 4*(1), 35-42.

Hax, A.C. & Candea, D. (1984). *Production and Inventory Management*. Englewood Cliffs: Prentice-Hall.

Hecht-Nielsen, R. (1989). Theory of the backpropagation neural network. *International Joint Conference on Neural Networks, USA, 1,* 593-605.

Hedgebeth, D. (2007). Data-driven decision making for the enterprise: an overview of business intelligence applications. *The Journal of Information and Knowledge Management Systems, 37*(4), 414-420.

Hendriks, P.H.J., & Vriens, D.J. (1999). Knowledge-based systems and knowledge management: friends or foes?. *Information & Management, 35*(2), 113-125.

Hieber, R. & Hartel, I. (2003). Impacts of SCM order strategies evaluated by simulation-based 'Beer Game' approach: The model, concept, and initial experiences. *Production Planning & Control, 14*(2), 122-134.

Hieta, S. (1998). Supply chain simulation with LOGSIM-simulator. *Proceedings of the 30th conference on Winter simulation, USA,* 323–326.

Hill, T., O'Connor, M., & William, R. (1996). Neural network models for time series forecasts. *Management Science, 42*(7), 1082-1092.

Holland, W., & Sodhi, M. (2004). Quantifying the effect of batch size and order errors on the bullwhip effect using simulation. *International Journal of Logistics, 7*(3), 251-261.

Holmstrom J. (1998a). Business process innovation in the supply chain: a case study of implementing VMI. *European Journal of Purchasing & Supply, 4*(2/3), 127-131.

Holmstrom J. (1998b). Implementing vendor-managed inventory the efficient way: A case study of partnership in the supply chain. *Production and Inventory Management Journal, 39*(3), 1-5.

Holmstrom, J., Framling, K., Kaipia, R., & Saranen, J. (2002). Collaborative planning forecasting and replenishment: New solutions needed for mass collaboration. *Supply Chain Management: An International Journal, 7*(3), 136-145.

Hsieh, C.C., & Wu, C.H. (2008). Capacity allocation, ordering, and pricing decisions in a supply chain with demand and supply uncertainties. *European Journal of Operational Research, 184*(2), 2008.

Hsu, R.L., Abdel-Mottaleb, M., & Hain, A.K. (2002). Face detection in color images. *IEEE Transactions on Pattern Analysis and Machine Intelligence, 24*(5), 696-706.

Hua, Z., Li, S., & Liang, L. (2006). Impact of demand uncertainty on supply chain cooperation of single-period products. *International Journal of Production Economics, 100*(2), 268-284.

Jung, J.Y., Blau, G., & Pekny, J.F. (2004). A simulation based optimization approach to supply chain management under demand uncertainty. *Computers and Chemical Engineering, 28*(10), 2087-2106.

Kandel, A (1992). *Fuzzy expert systems.* Boca Raton, Florida: CRC Press.

Kim, J., & Rogers, K.J. (2005). An object-oriented approach for building a flexible supply chain model. *International Journal of Physical Distribution & Logistics Management, 35*(7), 481-502.

Kim, T.M. (1985). Just-in-time manufacturing system: A periodic pull system. *International Journal of Production Research, 23*(3), 553-562.

Kleijnen, J.P.C. (2005). Supply chain simulation tools and techniques: A survey. *International Journal of Simulation and Process Modelling, 1*(1/2), 82-89.

Kolodner, J.L. (1992). An introduction to case-based reasoning. *Artificial Intelligence Review, 6*(1), 3-34.

Kolodner, J.L. (1983). Maintaining organization in a dynamic long-term memory. *Cognitive Science, 7*(4), 243-280.

Kolodner, J.L. (1993). Case-based reasoning. Morgan Kaufmann Publishers.

Kopacsi, S., & Kovacs, G.L. (1993). A knowledge based simulation system for flexible manufacturing. *IEEE International Conference on Robotics and Automation, 1,* 858-863.

Kotzab, H., Skjoldager, N. & Vinum, T. (2003). The development and empirical validation of an e-based supply chain strategy optimization model. Industrial *Management & Data Systems*, *103*(5), 347-360.

Kubota, F., Sato S., & Nakano, M. (1999) Enterprise modeling and simulation platform integrating manufacturing system design and supply chain. *IEEE International Conference on Systems, Man, and Cybernetics, 4,* 511-515.

Kuo, R.J., & Chen, J.A. (2004). A decision support system for order selection in electronic commerce based on fuzzy neural network supported by real-coded genetic algorithm. *Expert Systems with Applications*, *26*(2), 141-154.

Kwon, O. Im, G.P, & Lee, K.C. (2007). MACE-SCM: A multi-agent and case-based reasoning collaboration mechanism for supply chain management under supply and demand uncertainties. *Expert Systems with Applications*, *33*(3), 690-705.

Lambert, D.M., Cooper, M.C., & Pagh, J.D. (1998). Supply chain management: Implementation issues and research opportunities. *The International Journal of Logistics Management, 9*(2), 1-20.

Landry, S., & Duguay, C. (1997). Integrating MRP, Kanban and barcoding systems to achieve JIT procurement. *Production and Inventory Management Journal, 38*(1), 8-13.

Lau, H.C.W., Choy, K.L., Lau, P.K.H., Tsui, W.T. & Choy, L.C. (2004). An intelligent logistics support system for enhancing the airfreight forwarding business. *Expert Systems, 21*(5), 253-268.

Lee, L.H., Padmanabhan, V., & Whang, S. (1997). The bullwhip effect in supply chains. *Sloan Management Review, 38*(3), 93-102.

Li, L. (1992). The role of inventory in delivery-time competition. *Management Science, 48*(2), 182-197.

Liao, S.H. (2005). Expert system methodologies and applications - a decade review from 1995 to 2004. *Expert Systems with Applications, 28*(1), 93-103.

Lin, C.T., & Lee, C.S.G. (1991). Neural-network-based fuzzy logic control and decision system. *IEEE Transactions on computers, 40*(12), 1320-1336.

Lowson, B., King, R., & Hunter, A. (1999). *Quick response: Managing the supply chain to meet consumer demand.* New York, N.Y.: John Wiley & Sons.

Marshall, B., McDonald, D., Chen, H., & Chung, W. (2004). EBizPort: Collecting and analyzing business intelligence information. *Journal of the American Society for Information Science and Technology, 55*(10), 873-891.

Mehrabi, M.G., Ulsoy, A.G., & Koren, Y. (2000). Reconfigurable manufacturing systems: key to future manufacturing. *Journal of Intelligent Manufacturing, 11*(4), 403-419.

Mikhailov, L. (2000). A fuzzy programming method for deriving priorities in the analytic hierarchy process. *Journal of the Operational Research Society, 51*(3), 341-349.

Mishra, B.K. & Raghunathan, S. (2004). Retailer- vs. vendor-managed inventory and brand competition. *Management Science, 50*(4), 445-457.

Mitra, R.S., & Basu, A. (1997). Knowledge representation in mickey: An expert system for designing microprocessor-based systems. *IEEE Transactions on Systems, Man, and Cybernetics, 27*(4), 467-479.

Mockler, R.J., Dologite D.G., & Gartenfeld M.E. (2000). Talk with the experts: Learning management decision-making using CAI. *Cybernetics and Systems: An International Journal, 31*(4), 431–464.

Moinzadeh, K. (2002). A multi-echelon inventory system with information exchange. *Management Science, 48*(3), 414-426.

Morash, E.A. (2001). Supply chain strategies, capabilities. *Transportation Journal, 41*(1), 37-54.

Mulgrew, P., & Grant, J.T. (1999). *Digital signal processing: Concepts and applications.* Macmillan Press.

Nachiappan, S.P., & Jawahar, N. (2007). A genetic algorithm for optimal operating parameters of VMI system in a two-echelon supply chain. *European Journal of Operational Research, 182*(3), 1433-1452.

Nonaka, I., & Takeuchi, H. (1995). *The knowledge-creating company: How Japanese companies create the dynamics of innovation.* Oxford University Press.

O'Keefe, R.M., & Preece, A.D. (1996). The development, validation and implementation of knowledge-based systems. *European Journal of Operational Research, 92*(3), 458-473.

O'Leary, D.E. (1998). Using al in knowledge management: Knowledge based and ontologies. *IEEE Intelligent Systems, 13*(3), 34-39.

Olhager, J. & Ostlund, B. (1990). An integrated push-pull manufacturing strategy. *European Journal of Operational Research, 45*(2/3), 135-142.

Osuna, E., Freund, R., & Girosi, F. (1997). Training support vector machines: an application to face detection. *Proceedings of the 1997 Conference on Computer Vision and Pattern Recognition, USA,* 130-136.

Persson, F., & Olhager, J. (2002). Performance simulation of supply chain designs. *International Journal of Production Economics, 77*(3), 231-245.

Preece, A., Flett, A., Sleeman, D., Curry, D., Meany, N., & Perry, P. (2001). Better knowledge management through knowledge engineering. *IEEE Intelligent Systems*, *16*(1), 36-43.

Preis, M.W. (2003). The impact of interpersonal satisfaction on repurchase decisions. *Journal of Supply Chain Management*, *39*(3), 30-38.

Proakis, J.G., & Manolakis, D.G. (1996). *Digital signal processing: principles, algorithm, and applications* (3rd ed.). Upper Saddle River, NJ: Prentice Hall.

Rajagopalan, S. (2002). *Make to order or make to stock: Model and application. Management Science*, *48*(2), 241-256.

Riedmiller, M., & Braun, H. (1993). A direct adaptive method for faster backpropagation learning: the RPROP algorithm. *Proceedings of the IEEE International Conference on Neural Networks*, *1*, 586-591.

Rowley, H.A., Baluja, S., & Kanade, T. (1998). Neural network-based face detection. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, *20*(1), 23-37.

Russell S.J., & Norvig P. (2nd ed.) (2003). *Artificial intelligence: A modern approach.* Englewood Cliffs: Prentice-Hall.

Saaty, T.L. (1990). How to make a decision: The analytic hierarchy process. *European Journal of Operational Research, 48*(1), 9-26.

Sakakibara, S., Flynn, B.B., Schroeder, R.G., & William, T.M. (1997). The impact of just-in-time manufacturing and its infrastructure on manufacturing performance. *Management Science, 43*(9), 1246-1257.

Schank, R.C. (1982). *Dynamic memory: A theory of reminding and learning in computers and people.* Cambridge University Press.

Seifert, D. (2003). *Collaborative planning, forecasting, and replenishment: How to create a supply chain strategy*. New York: AMACOM.

Shore, B., & Venkatachalam, A.R. (2003). Evaluating the information sharing capabilities of supply chain partners: A fuzzy logic model. *International Journal of Physical Distribution & Logistics Management, 33*(9), 804-824.

Siegel, C.F. (2000). Introducing marketing students to business intelligence using project-based learning on the World Wide Web. *Journal of Marketing Education, 22*(2), 90-98.

Simchi-Levi, D., Kaminsky, P., & Simchi-Levi, E. (2003). *Designing and managing the supply chain: Concepts, strategies, and case studies* (2nd ed.). McGraw-Hill.

Simoudis, E. (1992). Using case-based reasoning for customer technical support. *IEEE Expert, 7*(5), 7-13.

Soman, A.C., Donk, D.P.V., & Gaalman, G. (2004). Combined make-to-order and make-to-stock in a food production system. *International Journal of Production Economics, 90*(2), 223-235.

Srinivasan, K., & Jayaraman, S. (1997). Integration of simulation with enterprise models. *Proceedings of the 1997 Winter Simulation Conference*, 1352-1356.

Stank, T.P., Keller, S. B. & Daugherty, P.J. (2001). Supply chain collaboration and logistical service performance. *Journal of Business Logistics, 22*(1), 29-48.

Svensson, G. (2002). Efficient consumer response - its origin and evolution in the history of marketing. *Management Decision, 40*(5/6), 508-519.

Syarif, A., Yun, Y., & Gen, M. (2002). Study on multi-stage logistic chain network: a spanning tree-based genetic algorithm approach. *Computers & Industrial Engineering, 43*(1/2), 299-314.

Terna, P. (2001). Creating artificial worlds: A note on sugars cape and two comments. *Journal of Artificial Societies and Social Simulation, 4*(2). Retrieved from 31 January, 2008 from the World Wide Web: http://www.soc.surrey.ac.uk/JASSS/4/2/9.html.

Toni, A.F.D, & Zamolo, E. (2005). From a traditional replenishment system to vendor-managed inventory: A case study from the household electrical appliances sector. *International Journal of Production Economics, 96*(1), 63-79.

Van der Zee, D.J., & Van der Vorst, J.G.A.J. (2005). A modelling framework for supply chain simulation: opportunities for improved decision making. *Decision Sciences, 36*(1), 65-95.

Vergin, R.C., & Barr, K. (1999). Building competitiveness in grocery supply through continuous replenishment planning: Insights from the field. *Industrial Marketing Management, 28*(2), 145-153.

VICS, (1999). Roadmap to CPFR: The case study. *Voluntary Interindustry Commerce Standards (VICS) Association.*

Volberda, H.W., & Rutges, A. (1999). Fatsys: A knowledge-based system for managing strategic change. *Decision Support Systems, 26*(2), 99-123.

Umeda, S., & Jones, A. (1998). An integration test-bed system for supply chain Management. *Proceedings of the 30th conference on Winter simulation, USA,* 1377–1385.

Usernik, J., & Bogataj, M. (2005). A fuzzy set approach for a location-inventory model. *Transportation Planning and Technology, 28*(6), 447-464.

Waller, M., Johnson M.E., & Davis, T. (1999). Vendor-managed inventory in the retailer supply chain. *Journal of Business Logistics, 20*(1), 183-203.

Wang, D., Yung, K.L., & Ip, W.H. (2001). A heuristic genetic algorithm for subcontractor selection in a global manufacturing environment. *IEEE Transactions on Systems, Man and Cybernetics-Part C: Applications and Reviews, 31*(2), 189-198.

Wang, L., Chu, J., & Wu, J. (2007). Selection of optimum maintenance strategies based on a fuzzy analytic hierarchy process. *International Journal of Production Economics, 107*(1), 151-163.

Wee, H.W., & Yang, P.C. (2004). The optimal and heuristic solutions of a distribution network. *European Journal of Operational Research, 158*(3), 626-632.

Wiig, K.M. (1997). Knowledge management: Where did it come from and where will it go?. *Expert Systems with Applications, 13*(1), 1-14.

Wiig, K.M. (2004). *People-focused knowledge management: How effective decision making leads to corporate success.* Burlington: Elsevier Inc.

Wilke, W., Vollrath, I., Althoff, K.D., & Bergmann, R. (1997). A framework for learning adaptation knowledge based on knowledge light approaches. *Proceedings of the Fifth German Workshop on Case-based Reasoning, GWCBR’97.*

Williams, E.J., & Gunal, A. (2003). Supply chain simulation and analysis with SIMFLEX. *Proceedings of the 2003 winter simulation conference,* 231-237.

Xie, Y., Petrovic, D., & Burnham, K. (2006). A heuristic procedure for the two-level control of serial supply chains under fuzzy customer demand. *International Journal of Production Economics, 102*(1), 37-50.

Yang C.F., & Pang, M.B. (2005). Game research into production material logistics center scope determining based on supply chain management. *Proceedings of International Conference on Services Systems and Services Management*, *1,* 651- 653.

Zadeh, L.A. (1975). Fuzzy logic and approximate reasoning. *Synthese, 30*(3/4), 407-428.

Appendix – Source Code of the KBSP

Agreement_VMI.asp

```
<html>
<head><title>VMI Agreement</title>
</ head >
<body topmargin="0" leftmargin="0">
<!-- #include file="top.inc" -->
<table border="0" cellpadding="0" cellspacing="0" style="border-collapse: collapse; border-width: 0" width="100%">
  <tr>
    <td width="17%" style="border-style: none; border-width: medium" valign="top"><!-- #include file="left.inc" --></td>
    <td width="83%" style="border-style: none; border-width: medium" valign="top" align="center"><!-- #include
file="Agreement_VMI.inc" --></td>
  </tr>
</table>
</body>
</html>
```

Agreement_VMI.inc

```
<!-- #include file="Redirect.inc" -->
<!--#include file="Database.inc"-->
<%
 Dim Con, SQL, RS
 Dim Products(), Product_ID()
 Dim Companys(), Company_ID()
 Dim SKU, Period_Start, Period_End, Unit_Price, Initial_Stock, Estimated_Qty, Upper_Lower_Limit, Batch_Size,
Order_Frequency, Lead_Time, Shippment, Status
 Dim Sales_Forecast
 Dim i

 Set Con = CreateObject("ADODB.Connection")
 Set RS = CreateObject("ADODB.Recordset")
 Con.Open DBCon

 If Request("Status") <> "" And Request("Product_ID") <> "" And Request("Company_ID") <> "" Then
  SQL = "SELECT * FROM [VMI] WHERE [Product_ID] = '" & Request("Product_ID") & "' And [Company_ID] = '" &
Request("Company_ID") & "'"
  RS.Open SQL, Con, 1, 3
  If RS.EOF Then RS.AddNew
  RS("Product_ID") = Request("Product_ID")
  RS("Company_ID") = Request("Company_ID")
  RS("SKU") = Request("SKU")
  RS("Period_Start") = Request("Period_Start")
  RS("Period_End") = Request("Period_End")
  RS("Unit_Price") = Request("Unit_Price")
  RS("Initial_Stock") = Request("Initial_Stock")
  RS("Estimated_Qty") = Request("Estimated_Qty")
  RS("Upper/Lower_Limit") = Request("Upper/Lower_Limit")
  RS("Batch_Size") = Request("Batch_Size")
  RS("Order_Frequency") = Request("Order_Frequency")
  RS("Lead_Time") = Request("Lead_Time")
  RS("Shippment") = Request("Shippment")
  RS("Status") = Request("Status")
  RS.Update
  RS.Close
 End If

 SQL = "SELECT [Product_ID], [Product_Name] FROM [Products]"
 RS.Open SQL, Con, 1, 3
 Redim Products(RS.RecordCount - 1), Product_ID(RS.RecordCount - 1)
 i = 0
 Do Until RS.EOF
  Products(i) = RS("Product_ID") & " - " & RS("Product_Name")
  Product_ID(i) = RS("Product_ID")
  i = i + 1
  RS.MoveNext
 Loop
 RS.Close
```

```
 SQL = "SELECT [Company_ID], [Company_Name] FROM [Companys]"
 RS.Open SQL, Con, 1, 3
 Redim Companys(RS.RecordCount - 1), Company_ID(RS.RecordCount - 1)
 i = 0
 Do Until RS.EOF
  Companys(i) = RS("Company_ID") & " - " & RS("Company_Name")
  Company_ID(i) = RS("Company_ID")
  i = i + 1
  RS.MoveNext
 Loop
 RS.Close

 Sales_Forecast = "Nil"
 If Request("Product_ID") <> "" And Request("Company_ID") <> "" Then
  SQL = "SELECT * FROM [VMI] WHERE [Product_ID] = '" & Request("Product_ID") & "' And [Company_ID] = '" &
Request("Company_ID") & "'"
  RS.Open SQL, Con, 1, 3
  If RS.EOF = False Then
   SKU = RS("SKU")
   Period_Start = RS("Period_Start")
   Period_End = RS("Period_End")
   Unit_Price = RS("Unit_Price")
   Initial_Stock = RS("Initial_Stock")
   Estimated_Qty = RS("Estimated_Qty")
   Upper_Lower_Limit = RS("Upper/Lower_Limit")
   Batch_Size = RS("Batch_Size")
   Order_Frequency = RS("Order_Frequency")
   Lead_Time = RS("Lead_Time")
   Shippment = RS("Shippment")
   Status = RS("Status")
  End If
  RS.Close

  SQL = "SELECT [Date] FROM [Sales Forecast] WHERE [Product_ID] = '" & Request("Product_ID") & "' And
[Company_ID] = '" & Request("Company_ID") & "' ORDER BY [Date]"
  RS.Open SQL, Con, 1, 3
  If RS.EOF = False Then
   Sales_Forecast = RS("Date") & " To "
   RS.MoveLast
   Sales_Forecast = Sales_Forecast & RS("Date")
  End If
  RS.Close
 End If

 Con.Close
 Set RS = Nothing
 Set Con = Nothing
%>
<form name="Frm" method="post" action="">
<table border="1" cellspacing="0" cellpadding="5">
   <tr>
     <td colspan="4" bgcolor="#ECE9D8" align="center"><strong>Agreement of VMI</strong></td>
   </tr>
   <tr>
     <td bgcolor="#ECE9D8">Company:</td>
     <td colspan="3">
<%If Session("Company_ID") = "0" Then%>
              <select name="Company_ID"
onChange="window.location.href='?Product_ID=<%=Request("Product_ID")%>&Company_ID=' +
this[this.selectedIndex].value;">
        <option>--- Please Select ---</option>
<%For i = 0 To Ubound(Company_ID)
 Response.Write("<option value=""" & Company_ID(i) & """")
 If Request("Company_ID") = Company_ID(i) Then Response.Write(" selected")
 Response.Write(">" & Companys(i) & "</option>")
Next%>
     </select>
<%Else%>
<%For i = 0 To Ubound(Company_ID)
 If Session("Company_ID") = Company_ID(i) Then
  Response.Write(Companys(i) & "<input type=""hidden"" name=""Company_ID"" value=""" & Company_ID(i) & """>")
  Exit For
 End If
Next%>
<%End If%>
```

```
                </td>
    </tr>
    <tr>
      <td bgcolor="#ECE9D8">Product:</td>
      <td colspan="3">
<%If Session("Company_ID") = "0" Then%>
                <select name="Product_ID"
onChange="window.location.href='?Company_ID=<%=Request("Company_ID")%>&Product_ID=' +
this[this.selectedIndex].value;">
<%Else%>
                <select name="Product_ID"
onChange="window.location.href='?Company_ID=<%=Session("Company_ID")%>&Product_ID=' +
this[this.selectedIndex].value;">
<%End If%>
        <option>--- Please Select ---</option>
<%For i = 0 To Ubound(Product_ID)
 Response.Write("<option value=""" & Product_ID(i) & """")
 If Request("Product_ID") = Product_ID(i) Then  Response.Write(" selected")
 Response.Write(">" & Products(i) & "</option>")
Next%>
      </select>
                </td>
    </tr>
    <tr>
      <td bgcolor="#ECE9D8">Period of Agreement From:</td>
      <td><input type="text" name="Period_Start" value="<%=Period_Start%>"></td>
      <td bgcolor="#ECE9D8">Period of Agreement To:</td>
      <td><input type="text" name="Period_End" value="<%=Period_End%>"></td>
    </tr>
    <tr>
      <td bgcolor="#ECE9D8">SKU:</td>
      <td>
        <select name="SKU">
          <option>--- Please Select ---</option>
          <option<%If SKU = "SKU0001" Then%> selected<%End If%>>SKU0001</option>
          <option<%If SKU = "SKU0002" Then%> selected<%End If%>>SKU0002</option>
        </select>
      </td>
      <td bgcolor="#ECE9D8">Unit Price (USD):</td>
      <td><input type="text" name="Unit_Price" value="<%=Unit_Price%>"></td>
    </tr>
    <tr>
      <td bgcolor="#ECE9D8">Initial Stock:</td>
      <td><input type="text" name="Initial_Stock" value="<%=Initial_Stock%>"></td>
      <td bgcolor="#ECE9D8">Estimated Qty:</td>
      <td><input type="text" name="Estimated_Qty" value="<%=Estimated_Qty%>"></td>
    </tr>
    <tr>
      <td bgcolor="#ECE9D8">Batch Size:</td>
      <td><input type="text" name="Batch_Size" value="<%=Batch_Size%>"></td>
      <td bgcolor="#ECE9D8">Upper/Lower Limit:</td>
      <td><input type="text" name="Upper/Lower_Limit" value="<%=Upper_Lower_Limit%>"></td>
    </tr>
    <tr>
      <td bgcolor="#ECE9D8">Order Frequency:</td>
      <td><input type="text" name="Order_Frequency" value="<%=Order_Frequency%>"></td>
      <td bgcolor="#ECE9D8">Lead Time (Days):</td>
      <td><input type="text" name="Lead_Time" value="<%=Lead_Time%>"></td>
    </tr>
    <tr>
      <td bgcolor="#ECE9D8">Shippment:</td>
      <td>
        <select name="Shippment">
          <option>--- Please Select ---</option>
          <option<%If Shippment = "Delivery by safety stock level" Then%> selected<%End If%>>Delivery by safety stock
level</option>
          <option<%If Shippment = "Delivery by call (or request)" Then%> selected<%End If%>>Delivery by call (or
request)</option>
        </select>
      </td>
      <td bgcolor="#ECE9D8">Agreement Status:</td>
      <td><%=Status%>    <input type="hidden" name="Status" value="<%=Status%>"></td>
    </tr>
    <tr>
      <td bgcolor="#ECE9D8">Period of Sales Forecast:</td>
```

```
    <td colspan="3"><%=Sales_Forecast%></td>
   </tr>
   <tr>
    <td colspan="4" bgcolor="#ECE9D8" align="center">
      <input type="button" value="Draft" onClick="FrmSend('Draft')">
<%If Session("Company_ID") = "0" Then%>
      <input type="button" value="Negotiate" onClick="FrmSend('Negotiate - Retailer')">
<%Else%>
      <input type="button" value="Negotiate" onClick="FrmSend('Negotiate - Customer')">
<%End If%>
      <input type="button" value="Confirm" onClick="FrmSend('Confirm')">
      <input type="reset" name="Reset" value="Reset">
    </div></td>
   </tr>
</table>
</form>

<script language="vbscript">
Sub FrmSend(InputVal)
 Frm.Status.Value = InputVal
 Frm.Submit
End Sub
</script>
```

ConvertDate.inc

```
<%
Function ConvertDate(InputDate)
 Dim MyCMonth
 If IsDate(InputDate) = False Then Exit Function
 MyCMonth = Array("Jan", "Feb", "Mar", "Apr", "May", "Jun", "Jul", "Aug", "Sep", "Oct", "Nov", "Dec")
 ConvertDate = Day(InputDate) & "-" & MyCMonth(Month(InputDate) - 1) & "-" & Year(InputDate)
End Function
%>
```

Customer_Detail.asp

```
<html>
<head><title>VMI Simulation</ title>
</ head >
<body topmargin="0" leftmargin="0">
<!-- #include file="top.inc" -->
<table border="0" cellpadding="0" cellspacing="0" style="border-collapse: collapse; border-width: 0" width="100%">
  <tr>
   <td width="17%" style="border-style: none; border-width: medium" valign="top"><!-- #include file="left.inc" --></td>
   <td width="83%" style="border-style: none; border-width: medium" valign="top" align="center"><!-- #include
file="Customer_Detail.inc" --></td>
  </tr>
</table>
</body>
</html>
```

Customer_Detail.inc

```
<!-- #include file="Redirect.inc" -->
<!--#include file="ConvertDate.inc"-->
<!--#include file="Database.inc"-->
<%
Call LoadPage

Sub LoadPage()
 Dim Con, SQL, RS
 Dim Product_ID, Start_Date, End_Date, Curr_Date, Company_ID, Company_Name, Company_IDs, Customer_Control
 Dim Curr_Inventory, Log, SF
 Dim Product_Name, Lead_Time, Upper_Lower_Limit, Batch_Size
 Dim i

 Set Con = CreateObject("ADODB.Connection")
 Set RS = CreateObject("ADODB.Recordset")
 Con.Open DBCon
```

```
'Get Company Name
Company_ID = Request("Company_ID")
SQL = "SELECT * FROM Companys WHERE Company_ID = '" & Company_ID & "'"
RS.Open SQL, Con, 1, 3
Company_Name = RS("Company_Name")
RS.Close

'Get Simulation
SQL = "SELECT * FROM Simulation"
RS.Open SQL, Con, 1, 3
Product_ID = RS("Product_ID")
Start_Date = RS("Start_Date")
End_Date = RS("End_Date")
Curr_Date = RS("Curr_Date")
Company_IDs = Split(RS("Customer_IDs"), ";")
For i = 0 To Ubound(Company_IDs)
 If Company_IDs(i) = Company_ID Then
  Customer_Control = Mid(RS("Customer_Control"), i + 1, 1)
  Exit For
 End If
Next
RS.Close

'Get Curr Inventory
SQL = "SELECT * FROM Simulation_Details WHERE Company_ID = '" & Company_ID & "' AND [Type] = 'Inventory'
AND [Request_Date] = #" & ConvertDate(Curr_Date) & "#"
RS.Open SQL, Con, 1, 3
Curr_Inventory = RS("Detail")
RS.Close

'Get Product Name
SQL = "SELECT * FROM [Products] WHERE [Product_ID] = '" & Product_ID & "'"
RS.Open SQL, Con, 1, 3
Product_Name = RS("Product_Name")
RS.Close

'Get VMI
If Customer_Control = 2 Then
 SQL = "SELECT * FROM [Push] WHERE Company_ID = '" & Company_ID & "' AND [Product_ID] = '" & Product_ID &
"'"
 RS.Open SQL, Con, 1, 3
 Lead_Time = RS("Lead_Time")
 RS.Close
Else
 SQL = "SELECT * FROM [VMI] WHERE Company_ID = '" & Company_ID & "' AND [Product_ID] = '" & Product_ID &
"'"
 RS.Open SQL, Con, 1, 3
 Lead_Time = RS("Lead_Time")
 Upper_Lower_Limit = RS("Upper/Lower_Limit")
 Batch_Size = RS("Batch_Size")
 RS.Close
End If

'Log
SQL = "SELECT * FROM Simulation_Details WHERE Company_ID = '" & Company_ID & "' AND ([Type] = 'PR' OR
[Type] = 'Delivery' OR ([Type] = 'Demand' AND [Request_Date] >= #" & ConvertDate(Curr_Date) & "# AND [Request_Date]
<= #" & ConvertDate(Curr_Date + 7) & "#)) ORDER BY [Request_Date]"
RS.Open SQL, Con, 1, 3
Do Until RS.EOF
 Select Case RS("Type")
  Case "PR"
   Log = Log & ConvertDate(RS("Request_Date")) & ": Requests supplier to deliver " & RS("Detail") & " units on " &
ConvertDate(RS("Deliver_Date")) & VbCrLf
  Case "Delivery"
   Log = Log & ConvertDate(RS("Request_Date")) & ": Supplier delivers " & RS("Detail") & " units" & VbCrLf
  Case "Demand"
   Log = Log & "There will be a demand of " & RS("Detail") & " units on " & ConvertDate(RS("Request_Date")) & VbCrLf
 End Select
 RS.MoveNext
Loop
RS.Close

'Sales Forecast
```

```
 SQL = "SELECT * FROM [Sales Forecast] WHERE Company_ID = '" & Company_ID & "' AND Product_ID = '" &
Product_ID & "' AND [Date] >= #" & Start_Date & "# AND [Date] <= #" & End_Date & "# ORDER BY [Date]"
 RS.Open SQL, Con, 1, 3
 Do Until RS.EOF
  SF = SF & ConvertDate(RS("Date")) & ": " & RS("Qty") & VbCrLf
  RS.MoveNext
 Loop
 RS.Close
%>
<br>
<form name="Frm" method="post" action="">
  <table border="1" cellspacing="0" cellpadding="5">
    <tr>
      <th colspan="4" bgcolor="#ECE9D8" align="center"><%=Company_Name%> Information</th>
    </tr>
    <tr>
      <td bgcolor="#ECE9D8">Product:</td><td colspan="3"><%=Product_ID%> - <%=Product_Name%></td>
    </tr>
    <tr>
      <td bgcolor="#ECE9D8">Current Date:</td><td><%=ConvertDate(Curr_Date)%></td>
      <td bgcolor="#ECE9D8">Current Inventory:</td><td><%=Curr_Inventory%></td>
    </tr>
<%If Customer_Control = 2 Then%>
    <tr>
      <td bgcolor="#ECE9D8">Lead Time:</td><td><%=Lead_Time%></td>
    </tr>
<%Else%>
    <tr>
      <td bgcolor="#ECE9D8">Batch Size:</td><td><%=Batch_Size%></td>
      <td bgcolor="#ECE9D8">Upper/Lower Limit:</td><td><%=Upper_Lower_Limit%></td>
    </tr>
    <tr>
      <td bgcolor="#ECE9D8">Lead Time:</td><td><%=Lead_Time%></td>
    </tr>
<%End If%>
    <tr>
      <th colspan="4" bgcolor="#ECE9D8" align="center">History Log</th>
    </tr>
    <tr>
      <td colspan="4"><textarea rows="10" cols="80"><%=Log%></textarea></td>
    </tr>
    <tr>
      <th colspan="4" bgcolor="#ECE9D8" align="center">Sales Forecast</th>
    </tr>
<%If Customer_Control <> 2 Then%>
    <tr>
      <td bgcolor="#ECE9D8">This month forecast demand:</td>
      <td><%=GetForecast(Con, Curr_Date, Company_ID, Product_ID)%></td>
    </tr>
<%End If%>
    <tr>
      <td colspan="4"><textarea rows="10" cols="80"><%=SF%></textarea></td>
    </tr>
  </table>
</form>
<%
 Con.Close
 Set RS = Nothing
 Set Con = Nothing
End Sub

Function GetForecast(Con, Curr_Date, Company_ID, Product_ID)
 Dim RS, SQL
 Dim Y(), Coef, Forecast
 Dim i
 Set RS = CreateObject("ADODB.Recordset")
 SQL = "SELECT * FROM [Forecast Parameters]"
 RS.Open SQL, Con, 1, 3
 GetForecast = Int(RS("Next_Forecast"))
 RS.Close
 Set RS = Nothing
End Function
%>
```

Database.inc

```
<%
Dim DBCon
DBCon = "Driver={Microsoft Access Driver (*.mdb)};DBQ=" & Server.MapPath("VMI.mdb") & ";UID=;PWD="
%>
```

Edit_Case.asp

```
<%If Session("Company_ID") <> "0" Then Response.Redirect("index.asp")%>
<html>
<head><title>Spectrum Analysis</ title>
</ head >

<body topmargin="0" leftmargin="0">
<!-- #include file="top.inc" -->
<table border="0" cellpadding="0" cellspacing="0" style="border-collapse: collapse; border-width: 0" width="100%">
  <tr>
    <td width="17%" style="border-style: none; border-width: medium" valign="top"><!-- #include file="left.inc" --></td>
    <td width="83%" style="border-style: none; border-width: medium" valign="top" align="center"><!-- #include
file="Edit_Case.inc" --></td>
  </tr>
</table>
</body>
</html>
```

Edit_Case.inc

```
<!-- #include file="Redirect.inc" -->
<!--#include file="Database.inc"-->
<%
 Dim Con, SQL, RS
 Dim Products(), Product_ID()
 Dim Companys(), Company_ID()
 Dim SKU, Period_Start, Period_End, Unit_Price, Initial_Stock, Estimated_Qty, Upper_Lower_Limit, Batch_Size,
Order_Frequency, Lead_Time, Shippment, Remarks, Pic_Link, Adopted_Customer
 Dim S_Pic_Link(), S_Company_ID(), S_Company_Name(), S_Text()
 Dim i

 Set Con = CreateObject("ADODB.Connection")
 Set RS = CreateObject("ADODB.Recordset")
 Con.Open DBCon

 If Request("Status") = "Save" Then
  SQL = "SELECT * FROM [VMI] WHERE [Product_ID] = '" & Request("Product_ID") & "' And [Company_ID] = '" &
Request("Company_ID") & "'"
  RS.Open SQL, Con, 1, 3
  If RS.EOF Then RS.AddNew
  RS("Product_ID") = Request("Product_ID")
  RS("Company_ID") = Request("Company_ID")
  RS("SKU") = Request("SKU")
  RS("Period_Start") = Request("Period_Start")
  RS("Period_End") = Request("Period_End")
  RS("Unit_Price") = Request("Unit_Price")
  RS("Initial_Stock") = Request("Initial_Stock")
  RS("Estimated_Qty") = Request("Estimated_Qty")
  RS("Upper/Lower_Limit") = Request("Upper/Lower_Limit")
  RS("Batch_Size") = Request("Batch_Size")
  RS("Order_Frequency") = Request("Order_Frequency")
  RS("Lead_Time") = Request("Lead_Time")
  RS("Shippment") = Request("Shippment")
  RS("Remarks") = Request("Remarks")
  RS("Pic_Link") = Request("Pic_Link")
  RS("Adopted_Customer") = Request("Adopted_Customer")
  RS.Update
  RS.Close
 End If

 SQL = "SELECT [Product_ID], [Product_Name] FROM [Products]"
 RS.Open SQL, Con, 1, 3
 Redim Products(RS.RecordCount - 1), Product_ID(RS.RecordCount - 1)
 i = 0
```

```
Do Until RS.EOF
 Products(i) = RS("Product_ID") & " - " & RS("Product_Name")
 Product_ID(i) = RS("Product_ID")
 i = i + 1
 RS.MoveNext
Loop
RS.Close

SQL = "SELECT [Company_ID], [Company_Name] FROM [Companys]"
RS.Open SQL, Con, 1, 3
Redim Companys(RS.RecordCount - 1), Company_ID(RS.RecordCount - 1)
i = 0
Do Until RS.EOF
 Companys(i) = RS("Company_ID") & " - " & RS("Company_Name")
 Company_ID(i) = RS("Company_ID")
 i = i + 1
 RS.MoveNext
Loop
RS.Close

If Request("Product_ID") <> "" And Request("Company_ID") <> "" Then
 SQL = "SELECT * FROM [VMI] WHERE [Product_ID] = '" & Request("Product_ID") & "' AND [Company_ID] = '" &
Request("Company_ID") & "'"
 RS.Open SQL, Con, 1, 3
 If RS.EOF = False Then
  SKU = RS("SKU")
  Period_Start = RS("Period_Start")
  Period_End = RS("Period_End")
  Unit_Price = RS("Unit_Price")
  Initial_Stock = RS("Initial_Stock")
  Estimated_Qty = RS("Estimated_Qty")
  Upper_Lower_Limit = RS("Upper/Lower_Limit")
  Batch_Size = RS("Batch_Size")
  Order_Frequency = RS("Order_Frequency")
  Lead_Time = RS("Lead_Time")
  Shippment = RS("Shippment")
  Remarks = RS("Remarks")
  Pic_Link = RS("Pic_Link")
  Adopted_Customer = RS("Adopted_Customer")
 End If
 RS.Close
End If

If Request("Status") = "Spectrum" Then
 SQL = "SELECT * FROM [VMI] WHERE [Product_ID] = '" & Request("Product_ID") & "' AND [Company_ID] <> '" &
Request("Company_ID") & "' AND [Pic_Link] <> ''"
 RS.Open SQL, Con, 1, 3
 Redim S_Pic_Link(RS.RecordCount - 1), S_Company_ID(RS.RecordCount - 1), S_Company_Name(RS.RecordCount - 1),
S_Text(RS.RecordCount - 1)
 For i = 0 To Ubound(S_Pic_Link)
  S_Pic_Link(i) = RS("Pic_Link")
  S_Company_ID(i) = RS("Company_ID")
  S_Text(i) = "Period of Agreement From:" & RS("Period_Start") & _
            " To " & RS("Period_End") & "<br>" & _
            "SKU: " & RS("SKU") & _
            ", Unit Price: " & RS("Unit_Price") & "<br>" & _
            "Initial Stock: " & RS("Initial_Stock") & _
            ", Estimated Qty: " & RS("Estimated_Qty") & "<br>" & _
            "Batch Size: " & RS("Batch_Size") & _
            ", Upper/Lower Limit: " & RS("Upper/Lower_Limit") & "<br>" & _
            "Lead Time: " & RS("Lead_Time") & _
            ", Shippment: " & RS("Shippment") & "<br>" & _
            "Remarks: " & RS("Remarks") & "<br>"
  RS.MoveNext
 Next
 RS.Close
 For i = 0 To Ubound(S_Pic_Link)
  SQL = "SELECT [Company_Name] FROM [Companys] WHERE [Company_ID] = '" & S_Company_ID(i) & "'"
  RS.Open SQL, Con, 1, 3
  S_Company_Name(i) = RS("Company_Name")
  RS.Close
 Next
End If

Con.Close
```

```
 Set RS = Nothing
 Set Con = Nothing
%>
<table border="1" cellspacing="0" cellpadding="5">
   <tr>
     <td colspan="4" bgcolor="#ECE9D8" align="center"><strong>Customer Demand Spectrum</strong></td>
   </tr>
   <tr>
     <td bgcolor="#ECE9D8">Company:</td>
     <td colspan="3">
                <select name="Company_ID"
onChange="window.location.href='?Product_ID=<%=Request("Product_ID")%>&Company_ID=' +
this[this.selectedIndex].value;">
         <option>--- Please Select ---</option>
<%For i = 0 To Ubound(Company_ID)
 Response.Write("<option value=""" & Company_ID(i) & """")
 If Request("Company_ID") = Company_ID(i) Then Response.Write(" selected")
 Response.Write(">" & Companys(i) & "</option>")
Next%>
     </select>
                </td>
   </tr>
   <tr>
     <td bgcolor="#ECE9D8">Product:</td>
     <td colspan="3">
                <select name="Product_ID"
onChange="window.location.href='?Company_ID=<%=Request("Company_ID")%>&Product_ID=' +
this[this.selectedIndex].value;">
         <option>--- Please Select ---</option>
<%For i = 0 To Ubound(Product_ID)
 Response.Write("<option value=""" & Product_ID(i) & """")
 If Request("Product_ID") = Product_ID(i) Then  Response.Write(" selected")
 Response.Write(">" & Products(i) & "</option>")
Next%>
     </select>
                </td>
   </tr>

   <form enctype="multipart/form-data" method="post" name="Frm_Upload" action="Upload_Process.asp" target="_blank">
   <tr>
     <td bgcolor="#ECE9D8">Upload Spectrum Image:</td>
     <td colspan="3">
       <input type="file" name="ImgLink">
       <input type="submit" name="XXX" value="Upload" onClick="FrmSend('Upload')">
     </td>
   </tr>
   </form>

<form name="Frm" method="post" action="">
   <%If Pic_Link <> "" Then%>
   <tr><td colspan="4" align="center"><img name="Img_Link" src="<%=Pic_Link%>" height="450"
width="600"></td></tr>
   <%End If%>
   <input type="hidden" name="Pic_Link" value="<%=Pic_Link%>">
   <tr>
     <td bgcolor="#ECE9D8">Adopted Customer:</td>
     <td colspan="3"><%=Adopted_Customer%></td>
   <input type="hidden" name="Adopted_Customer" value="<%=Adopted_Customer%>">
   </tr>
   <tr>
     <td bgcolor="#ECE9D8">Remarks:</td>
     <td colspan="3"><input type="text" style="width:100%" value="<%=Remarks%>" name="Remarks"></td>
   </tr>

   <tr>
     <td colspan="4" bgcolor="#ECE9D8" align="center"><strong>VMI Agreement</strong></td>
   </tr>
   <tr>
     <td bgcolor="#ECE9D8">Period of Agreement From:</td>
     <td><input type="text" name="Period_Start" value="<%=Period_Start%>"></td>
     <td bgcolor="#ECE9D8">Period of Agreement To:</td>
     <td><input type="text" name="Period_End" value="<%=Period_End%>"></td>
   </tr>
   <tr>
     <td bgcolor="#ECE9D8">SKU:</td>
```

```
    <td>
      <select name="SKU">
        <option>--- Please Select ---</option>
        <option<%If SKU = "SKU0001" Then%> selected<%End If%>>SKU0001</option>
        <option<%If SKU = "SKU0002" Then%> selected<%End If%>>SKU0002</option>
      </select>
    </td>
    <td bgcolor="#ECE9D8">Unit Price (USD):</td>
    <td><input type="text" name="Unit_Price" value="<%=Unit_Price%>"></td>
  </tr>
  <tr>
    <td bgcolor="#ECE9D8">Initial Stock:</td>
    <td><input type="text" name="Initial_Stock" value="<%=Initial_Stock%>"></td>
    <td bgcolor="#ECE9D8">Estimated Qty:</td>
    <td><input type="text" name="Estimated_Qty" value="<%=Estimated_Qty%>"></td>
  </tr>
  <tr>
    <td bgcolor="#ECE9D8">Batch Size:</td>
    <td><input type="text" name="Batch_Size" value="<%=Batch_Size%>"></td>
    <td bgcolor="#ECE9D8">Upper/Lower Limit:</td>
    <td><input type="text" name="Upper/Lower_Limit" value="<%=Upper_Lower_Limit%>"></td>
  </tr>
  <tr>
    <td bgcolor="#ECE9D8">Order Frequency:</td>
    <td><input type="text" name="Order_Frequency" value="<%=Order_Frequency%>"></td>
    <td bgcolor="#ECE9D8">Lead Time (Days):</td>
    <td><input type="text" name="Lead_Time" value="<%=Lead_Time%>"></td>
  </tr>
  <tr>
    <td bgcolor="#ECE9D8">Shippment:</td>
    <td>
      <select name="Shippment">
        <option>--- Please Select ---</option>
        <option<%If Shippment = "Delivery by safety stock level" Then%> selected<%End If%>>Delivery by safety stock
level</option>
        <option<%If Shippment = "Delivery by call (or request)" Then%> selected<%End If%>>Delivery by call (or
request)</option>
      </select>
    </td>
  </tr>

  <tr>
    <td colspan="4" bgcolor="#ECE9D8" align="center">
      <input type="button" value="Show Others Customer Spectrum" onClick="FrmSend('Spectrum')">
      <input type="button" value="Save" onClick="FrmSend('Save')">
      <input type="reset" name="Reset" value="Reset">
      <input type="hidden" name="Status" value="Save">
      <input type="hidden" value="Company_ID" value="<%=Request("Company_ID")%>">
      <input type="hidden" value="Product_ID" value="<%=Request("Company_ID")%>">
    </td>
  </tr>

<%If Request("Status") = "Spectrum" Then%>
  <tr>
    <td colspan="4" bgcolor="#ECE9D8" align="center"><strong>Others Demand Spectrum</strong></td>
  </tr>
  <tr>
    <td bgcolor="#ECE9D8">Company:</td>
    <td colspan="3">
            <select name="S_Company_ID" onChange="ChangeSpectrum(this.selectedIndex - 1)">
        <option>--- Please Select ---</option>
<%For i = 0 To Ubound(S_Company_ID)
 Response.Write("<option value=""" & S_Company_ID(i) & """")
 Response.Write(">" & S_Company_Name(i) & "</option>")
Next%>
    </select>
            </td>
  </tr>
  <tr>
    <td bgcolor="#ECE9D8">Spectrum:</td>
    <td colspan="3"><div id="Spectrum"></div></td>
  </tr>
  <tr>
    <td bgcolor="#ECE9D8">VMI Agreement Terms:</td>
    <td colspan="3"><div id="VMI"></div></td>
```

```
    </tr>
    <tr>
      <td colspan="4" bgcolor="#ECE9D8" align="center">
        <input type="button" value="Adopt this Customer" onClick="FrmSend('Adopt')">
      </td>
    </tr>

<%End If%>
</table>
</form>

<script language="vbscript">
Sub FrmSend(InputVal)
 Select Case InputVal
  Case "Upload"
   Frm.Pic_Link.Value = "Spectrum\" & Mid(Frm_Upload.ImgLink.Value, InStrRev(Frm_Upload.ImgLink.Value, "\") + 1)
  Case "Clear"
   Frm.Pic_Link.Value = ""
   Frm.Status.Value = "Save"
   Frm.Submit
  Case "Adopt"
   If Frm.S_Company_ID.selectedIndex = 0 Then Exit Sub
   Frm.Adopted_Customer.Value = Frm.S_Company_ID(Frm.S_Company_ID.selectedIndex).Value
   Frm.Status.Value = "Save"
   Frm.Submit
  Case Else
   Frm.Status.Value = InputVal
   Frm.Submit
 End Select
End Sub

<%If Request("Status") = "Spectrum" Then%>
Sub ChangeSpectrum(InputVal)
 If InputVal = -1 Then Exit Sub
 Dim S_Pic_Link(<%=Ubound(S_Pic_Link)%>), S_Company_ID(<%=Ubound(S_Pic_Link)%>),
S_Company_Name(<%=Ubound(S_Pic_Link)%>), S_Text(<%=Ubound(S_Pic_Link)%>)
<%For i = 0 To Ubound(S_Pic_Link)%>
 S_Pic_Link(<%=i%>) = "<%=S_Pic_Link(i)%>"
 S_Company_ID(<%=i%>) = "<%=S_Company_ID(i)%>"
 S_Company_Name(<%=i%>) = "<%=S_Company_Name(i)%>"
 S_Text(<%=i%>) = "<%=S_Text(i)%>"
<%Next%>

 document.all("Spectrum").innerhtml = "<img src=""" & S_Pic_Link(InputVal) & """ height=""450"" width=""600"">"
 document.all("VMI").innerhtml = S_Text(InputVal) & "<br><input type=""button"" value=""Show Simulation Log""
onClick=""OpenNewWin('" & S_Company_ID(InputVal) & "')"">"
End Sub

Sub OpenNewWin(InputVal)
 Call window.open("Simulation_Log.asp?Company_ID=" & InputVal & "&Product_ID=<%=Request("Product_ID")%>",
"_blank")
End Sub
<%End If%>
</script>
```

Gen_Excel.inc

```
<%
Sub GenExcelChart(XAxis(), Series(), SourceData(), Title, XAxisName, YAxisName, PicLink)
 Dim xlapp 'Our Excel App
 Dim wb 'Our Workbook within the Excel App
 Dim ws 'Our Worksheet within the Workbook
 Dim crt 'The chart object
 Dim SourceRange 'The Source Range for the chart object
 Dim i, j, CharArr

 CharArr = Array("A", "B", "C", "D", "E", "F", "G", "H", "I", "J", "K", "L", "M", "N", "O", "P", "Q", "R", "S", "T", "U", "V",
"W", "X", "Y", "Z", "AA", "AB", "AC", "AD", "AE", "AF", "AG", "AH", "AI", "AJ", "AK", "AL", "AM", "AN", "AO", "AP",
"AQ", "AR", "AS", "AT", "AU", "AV", "AW", "AX", "AY", "AZ", "BA", "BB", "BC", "BD", "BE", "BF", "BG", "BH", "BI",
"BJ", "BK", "BL", "BM", "BN", "BO", "BP", "BQ", "BR", "BS", "BT", "BU", "BV", "BW", "BX", "BY", "BZ", "CA", "CB",
"CC", "CD", "CE", "CF", "CG", "CH", "CI", "CJ", "CK", "CL", "CM", "CN", "CO", "CP", "CQ", "CR", "CS", "CT", "CU",
"CV", "CW", "CX", "CY", "CZ", "DA", "DB", "DC", "DD", "DE", "DF", "DG", "DH", "DI", "DJ", "DK", "DL", "DM", "DN",
"DO", "DP", "DQ", "DR", "DS", "DT", "DU", "DV", "DW", "DX", "DY", "DZ", "EA", "EB", "EC", "ED", "EE", "EF", "EG",
"EH", "EI", "EJ", "EK", "EL", "EM", "EN", "EO", "EP", "EQ", "ER", "ES", "ET", "EU", "EV", "EW", "EX", "EY", "EZ",
"FA", "FB", "FC", "FD", "FE", "FF", "FG", "FH", "FI", "FJ", "FK", "FL", "FM", "FN", "FO", "FP", "FQ", "FR", "FS", "FT",
"FU", "FV", "FW", "FX", "FY", "FZ", "GA", "GB", "GC", "GD", "GE", "GF", "GG", "GH", "GI", "GJ", "GK", "GL", "GM",
"GN", "GO", "GP", "GQ", "GR", "GS", "GT", "GU", "GV", "GW", "GX", "GY", "GZ", "HA", "HB", "HC", "HD", "HE",
"HF", "HG", "HH", "HI", "HJ", "HK", "HL", "HM", "HN", "HO", "HP", "HQ", "HR", "HS", "HT", "HU", "HV", "HW", "HX",
"HY", "HZ", "IA", "IB", "IC", "ID", "IE", "IF", "IG", "IH", "II", "IJ", "IK", "IL", "IM", "IN", "IO", "IP", "IQ", "IR", "IS", "IT",
"IU", "IV", "IW", "IX", "IY", "IZ", "JA", "JB", "JC", "JD", "JE", "JF", "JG", "JH", "JI", "JJ", "JK", "JL", "JM", "JN", "JO",
"JP", "JQ", "JR", "JS", "JT", "JU", "JV", "JW", "JX", "JY", "JZ", "KA", "KB", "KC", "KD", "KE", "KF", "KG", "KH", "KI",
"KJ", "KK", "KL", "KM", "KN", "KO", "KP", "KQ", "KR", "KS", "KT", "KU", "KV", "KW", "KX", "KY", "KZ", "LA", "LB",
"LC", "LD", "LE", "LF", "LG", "LH", "LI", "LJ", "LK", "LL", "LM", "LN", "LO", "LP", "LQ", "LR", "LS", "LT", "LU", "LV",
"LW", "LX", "LY", "LZ", "MA", "MB", "MC", "MD", "ME", "MF", "MG", "MH", "MI", "MJ", "MK", "ML", "MM", "MN",
"MO", "MP", "MQ", "MR", "MS", "MT", "MU", "MV", "MW", "MX", "MY", "MZ", "NA", "NB", "NC", "ND", "NE", "NF",
"NG", "NH", "NI", "NJ", "NK", "NL", "NM", "NN", "NO", "NP", "NQ", "NR", "NS", "NT", "NU", "NV", "NW", "NX", "NY",
"NZ", "OA", "OB", "OC", "OD", "OE", "OF", "OG", "OH", "OI", "OJ", "OK", "OL", "OM", "ON", "OO", "OP", "OQ", "OR",
"OS", "OT", "OU", "OV", "OW", "OX", "OY", "OZ", "PA", "PB", "PC", "PD", "PE", "PF", "PG", "PH", "PI", "PJ", "PK",
"PL", "PM", "PN", "PO", "PP", "PQ", "PR", "PS", "PT", "PU", "PV", "PW", "PX", "PY", "PZ", "QA", "QB", "QC", "QD",
"QE", "QF", "QG", "QH", "QI", "QJ", "QK", "QL", "QM", "QN", "QO", "QP", "QQ", "QR", "QS", "QT", "QU", "QV", "QW",
"QX", "QY", "QZ", "RA", "RB", "RC", "RD", "RE", "RF", "RG", "RH", "RI", "RJ", "RK", "RL", "RM", "RN", "RO", "RP",
"RQ", "RR", "RS", "RT", "RU", "RV", "RW", "RX", "RY", "RZ", "SA", "SB", "SC", "SD", "SE", "SF", "SG", "SH", "SI",
"SJ", "SK", "SL", "SM", "SN", "SO", "SP", "SQ", "SR", "SS", "ST", "SU", "SV", "SW", "SX", "SY", "SZ", "TA", "TB", "TC",
"TD", "TE", "TF", "TG", "TH", "TI", "TJ", "TK", "TL", "TM", "TN", "TO", "TP", "TQ", "TR", "TS", "TT", "TU", "TV",
"TW", "TX", "TY", "TZ", "UA", "UB", "UC", "UD", "UE", "UF", "UG", "UH", "UI", "UJ", "UK", "UL", "UM", "UN", "UO",
"UP", "UQ", "UR", "US", "UT", "UU", "UV", "UW", "UX", "UY", "UZ", "VA", "VB", "VC", "VD", "VE", "VF", "VG",
"VH", "VI", "VJ", "VK", "VL", "VM", "VN", "VO", "VP", "VQ", "VR", "VS", "VT", "VU", "VV", "VW", "VX", "VY", "VZ",
"WA", "WB", "WC", "WD", "WE", "WF", "WG", "WH", "WI", "WJ", "WK", "WL", "WM", "WN", "WO", "WP", "WQ",
"WR", "WS", "WT", "WU", "WV", "WW", "WX", "WY", "WZ", "XA", "XB", "XC", "XD", "XE", "XF", "XG", "XH", "XI",
"XJ", "XK", "XL", "XM", "XN", "XO", "XP", "XQ", "XR", "XS", "XT", "XU", "XV", "XW", "XX", "XY", "XZ", "YA",
"YB", "YC", "YD", "YE", "YF", "YG", "YH", "YI", "YJ", "YK", "YL", "YM", "YN", "YO", "YP", "YQ", "YR", "YS", "YT",
"YU", "YV", "YW", "YX", "YY", "YZ", "ZA", "ZB", "ZC", "ZD", "ZE", "ZF", "ZG", "ZH", "ZI", "ZJ", "ZK", "ZL", "ZM",
"ZN", "ZO", "ZP", "ZQ", "ZR", "ZS", "ZT", "ZU", "ZV", "ZW", "ZX", "ZY", "ZZ")

 Const xlWorkSheet = -4167
 Const xlLineMarkers = 65

 Set xlapp = Server.CreateObject("Excel.Application") 'Create an instance of Excel Application
 Set wb = xlapp.Workbooks.Add(xlWorksheet) 'Create a new workbook
 Set ws = wb.Worksheets(1) 'Grab the first worksheet of the new workbook

 ws.Range("A1").Value = Title 'Title of the chart

 For j = 0 To Ubound(Series)
  ws.Range(CharArr(j + 1) & 1).Value = Series(j)
 Next
 For i = 0 To Ubound(XAxis)
  ws.Range("A" & i + 2).Value = XAxis(i)
  For j = 0 To Ubound(Series)
   ws.Range(CharArr(j + 1) & i + 2).Value = SourceData(j, i)
  Next
 Next

 Set SourceRange = ws.Range("A2:" & CharArr(UBound(Series) + 1) & i + 1) 'Set our source range
 Set crt = ws.ChartObjects.Add(0, 0, 500, 450) 'Create a new Chart Object

 'crt.Chart.ChartWizard SourceRange, 4, , 2, 1, 0, 2, Title, XAxisName, YAxisName
 crt.Chart.ChartWizard SourceRange, 4, , 2, 1, 0, True, Title, XAxisName, YAxisName
 crt.Chart.ChartType = xlLineMarkers
 crt.Chart.SeriesCollection(1).Name = "=Sheet1!R1C1"
 crt.Chart.HasTitle = True
```

```
 For j = 0 To Ubound(Series)
  crt.Chart.SeriesCollection(j + 1).Name = "=Sheet1!R1C" & j + 2
 Next

 crt.Chart.Export Server.Mappath(PicLink), "gif" 'Save the chart on web server (gif also ok, but need to pay licence fee)
 wb.Saved = True 'Fool Excel into thinking the Workbook is saved

 xlApp.Visible = True

 Set crt = Nothing
 Set wb = Nothing
 'xlapp.Quit 'Quit Excel to conserve resources
 Set xlapp = Nothing

 'Make sure the page is not cached but is loaded fresh from the web server
 Response.AddHeader "expires","0"
 Response.AddHeader "pragma", "no-cache"
 Response.AddHeader "cache-control","no-cache"
End Sub
%>
```

Index.asp

```
<html>
<head><title>VMI</title>
</head>

<body topmargin="0" leftmargin="0">
<!-- #include file="top.inc" -->

<table border="0" cellpadding="0" cellspacing="0" style="border-collapse: collapse; border-width: 0" width="100%">
  <tr>
    <td width="17%" style="border-style: none; border-width: medium" valign="top"><!-- #include file="left.inc" --></td>
    <td width="83%" style="border-style: none; border-width: medium" valign="top" align="center"><!-- #include
file="index.inc" --></td>
  </tr>
</table>
</body>
</html>
```

Index.inc

```
<!-- #include file="Redirect.inc" -->
<br><br><br><img src="images/map.jpg" width="459" height="358" />
```

Left.inc

```
  <table width="130" border="5" cellspacing="0" cellpadding="5">
    <tr>
      <td width="120"><div align="center"><em><strong><a href="../../fyp/index.asp">Push
System</a></strong></em></div></td>
    </tr>
  </table>
  <br />
  <table width="130" border="5" cellspacing="0" cellpadding="5">
    <tr>
      <td width="120"><div align="center"><em><strong><a href="Agreement_VMI.asp">VMI
Agreement</a></strong></em></div></td>
    </tr>
  </table>
  <br />
  <table width="130" border="5" cellspacing="0" cellpadding="5">
    <tr>
      <td width="120"><div align="center"><em><strong><a href="Simulation.asp">Simulation
Module</a></strong></em></div></td>
    </tr>
  </table>
  <br />
  <table width="130" border="5" cellspacing="0" cellpadding="5">
    <tr>
```

```
    <td width="120"><div align="center"><em><strong><a href="Edit_Case.asp">Spectrum
Analysis</a></strong></em></div></td>
   </tr>
  </table>
  <br />
  <table width="130" border="5" cellspacing="0" cellpadding="5">
   <tr>
    <td width="120"><div align="center"><em><strong><a href="Logout.asp">Logout</a></strong></em></div></td>
   </tr>
  </table>
```

Login.asp

```
<html>
<head><title>VMI</ title>
</ head>

<body topmargin="0" leftmargin="0">
<!-- #include file="top.inc" -->

<table border="0" cellpadding="0" cellspacing="0" style="border-collapse: collapse; border-width: 0" width="100%">
  <tr>
   <td width="17%" style="border-style: none; border-width: medium" valign="top"><!-- #include file="left.inc" --></td>
   <td width="83%" style="border-style: none; border-width: medium" valign="top" align="center"><!-- #include
file="Login.inc" --></td>
  </tr>
</table>

</body>
</html>
```

Login.inc

```
<!--#include file="Database.inc"-->
<%
If Request("User_ID") <> "" Then
 Dim Con, SQL, RS

 Set Con = CreateObject("ADODB.Connection")
 Set RS = CreateObject("ADODB.Recordset")
 Con.Open DBCon

 SQL = "SELECT * FROM [Login] WHERE [User_ID] = '" & Request("User_ID") & "' AND [Password] = '" &
Request("Password") & "'"
 RS.Open SQL, Con, 1, 3

 If RS.EOF = False Then
  Session("Company_ID") = RS("Company_ID")
 Else
  Msg = "Invalid User ID or Password"
  Session("Company_ID") = ""
 End If

 RS.Close
 Con.Close
 Set RS = Nothing
 Set Con = Nothing

 If Session("Company_ID") <> "" Then Response.Redirect("index.asp")
End If
%>
<br><br><br>
<form method="post" action="login.asp" name="Login">
<center><%=Msg%></center>
<table border="1" cellspacing="0" cellpadding="5" align="center">
  <tr>
   <th colspan="2" bgcolor="#ECE9D8" align="center"><strong>Login</strong></th>
  </tr>
  <tr>
   <td bgcolor="#ECE9D8">User ID:</td>
   <td><input type="text" name="User_ID" value="<%=Request("User_ID")%>"></td>
  </tr>
```

```
  <tr>
    <td bgcolor="#ECE9D8">Password:</td>
    <td><input type="password" name="Password"></td>
  </tr>
  <tr>
    <td bgcolor="#ECE9D8" colspan="2" align="center"><input type="submit" value="Submit"> <input type="reset"
value="Reset"></td>
  </tr>
</table>
</form>
```

Logout.asp

```
<%Session.Abandon%>
<html>
<meta http-equiv="refresh" content="5; url=Login.asp">
<head><title>VMI</ title>
</ head>

<body topmargin="0" leftmargin="0">
<!-- #include file="top.inc" -->

<table border="0" cellpadding="0" cellspacing="0" style="border-collapse: collapse; border-width: 0" width="100%">
  <tr>
    <td width="17%" style="border-style: none; border-width: medium" valign="top"><!-- #include file="left.inc" --></td>
    <td width="83%" style="border-style: none; border-width: medium" valign="top" align="center"><br><br><br>Logout. The
page will go back to login page within 5 seconds.</td>
  </tr>
</table>

</body>
</html>
```

Redirect.inc

```
<%If Session("Company_ID") = "" Then Response.Redirect("Login.asp")%>
```

SetInventory.inc

```
<%
Sub SetInventory(Inventory(), Company_Index, StartDate, EndDate)
 Dim i

 For i = StartDate + 1 To EndDate
  Inventory(Company_Index, i) = Inventory(Company_Index, i - 1)
 Next
End Sub

Sub SetRetailerInventory(Inventory(), StartDate, EndDate, Reorder())
 Dim i

 For i = StartDate + 1 To EndDate
  Inventory(0, i) = Inventory(0, i - 1) + Reorder(i)
 Next
End Sub
%>
```

Simulation.asp

```
<%
If Session("Company_ID") = "0" Then
 Response.Redirect("Simulation_Retailer.asp")
Else
 Response.Redirect("Simulation_Customer.asp")
End If
%>
```

Simulation_Customer.asp

```
<html>
<head><title>VMI Simulation</title>
</head>

<body topmargin="0" leftmargin="0">
<!-- #include file="top.inc" -->

<table border="0" cellpadding="0" cellspacing="0" style="border-collapse: collapse; border-width: 0" width="100%">
  <tr>
    <td width="17%" style="border-style: none; border-width: medium" valign="top"><!-- #include file="left.inc" --></td>
    <td width="83%" style="border-style: none; border-width: medium" valign="top" align="center"><!-- #include
file="Simulation_Customer.inc" --></td>
  </tr>
</table>

</body>
</html>
```

Simulation_Customer.inc

```
<!-- #include file="Redirect.inc" -->
<!--#include file="ConvertDate.inc"-->
<!--#include file="GenExcel.inc"-->
<!--#include file="Database.inc"-->
<%
If CheckSimulation = True Then
 If Request("Sim_Status") = "PR_Supplier_Issue" Then
  Call UpdateSupplierIssue
  Call LoadRunSimulation
 ElseIf Request("Sim_Status") = "Next_Day" Then
  Call UpdateComplete
  Call LoadRunSimulation
 ElseIf Request("Sim_Status") = "Show_Result" Then
  Call ShowResult
 Else
  Call LoadRunSimulation
 End If
Else
 Response.Write("<br><br>There is no simulation conducting.")
End If

Function CheckSimulation()
 Dim Con, SQL, RS
 Dim Company_ID

 Set Con = CreateObject("ADODB.Connection")
 Set RS = CreateObject("ADODB.Recordset")
 Con.Open DBCon

 SQL = "SELECT * FROM [Simulation]"
 RS.Open SQL, Con, 1, 3

 If RS.EOF Then
  CheckSimulation = False
 Else
  If InStr(RS("Customer_IDs"), Session("Company_ID")) > 0 Then
   Company_ID = Split(RS("Customer_IDs"), ";")
   For i = 0 To Ubound(Company_ID)
    If Company_ID(i) = Session("Company_ID") Then
     If Mid(RS("Customer_Control"), i + 1, 1) = 0 Then
      CheckSimulation = True
     Else
      CheckSimulation = False
     End If
     Exit For
    End If
   Next
  Else
   CheckSimulation = False
  End If
 End If
 RS.Close

 Con.Close
```

```
 Set RS = Nothing
 Set Con = Nothing
End Function

Sub UpdateComplete()
 Dim Con, SQL, RS
 Dim Company_ID
 Dim i

 Set Con = CreateObject("ADODB.Connection")
 Set RS = CreateObject("ADODB.Recordset")
 Con.Open DBCon

 SQL = "SELECT * FROM Simulation"
 RS.Open SQL, Con, 1, 3
 Company_ID = Split(RS("Customer_IDs"), ";")
 For i = 0 To Ubound(Company_ID)
  If Company_ID(i) = Session("Company_ID") Then
   RS("Customer_Complete") = Mid(RS("Customer_Complete"), 1, i) & 1 & Mid(RS("Customer_Complete"), i + 2,
Len(RS("Customer_Complete")))
   Exit For
  End If
 Next
 RS.Update
 RS.Close

 Con.Close
 Set RS = Nothing
 Set Con = Nothing
End Sub

Sub UpdateSupplierIssue()
 Dim Con, SQL, RS
 Dim Product_ID, Curr_Date, Lead_Time

 Set Con = CreateObject("ADODB.Connection")
 Set RS = CreateObject("ADODB.Recordset")
 Con.Open DBCon

 'Get Product ID
 SQL = "SELECT * FROM Simulation"
 RS.Open SQL, Con, 1, 3
 Product_ID = RS("Product_ID")
 Curr_Date = RS("Curr_Date")
 RS.Close

 'Get Lead Time
 SQL = "SELECT * FROM [VMI] WHERE Company_ID = '" & Session("Company_ID") & "' AND [Product_ID] = '" &
Product_ID & "'"
 RS.Open SQL, Con, 1, 3
 Lead_Time = RS("Lead_Time")
 RS.Close

 SQL = "SELECT * FROM [Simulation_Details]"
 RS.Open SQL, Con, 1, 3
 RS.AddNew
 RS("Company_ID") = Session("Company_ID")
 RS("Request_Date") = ConvertDate(Curr_Date)
 RS("Type") = "PR"
 RS("Detail") = Request("PR_Supplier_Qty")
 RS("Deliver_Date") = ConvertDate(Curr_Date + Lead_Time)
 RS.Update
 RS.Close

 Con.Close
 Set RS = Nothing
 Set Con = Nothing
End Sub

Sub LoadRunSimulation()
 Dim Con, SQL, RS
 Dim Product_ID, End_Date, Curr_Date, Company_ID, Company_Name, Company_IDs, Customer_Complete,
Other_Complete
 Dim Curr_Inventory, Log
 Dim Product_Name, Lead_Time, Upper_Lower_Limit, Batch_Size
```

```
 Dim i

 Set Con = CreateObject("ADODB.Connection")
 Set RS = CreateObject("ADODB.Recordset")
 Con.Open DBCon

 'Get Company Name
 Company_ID = Session("Company_ID")
 SQL = "SELECT * FROM Companys WHERE Company_ID = '" & Company_ID & "'"
 RS.Open SQL, Con, 1, 3
 Company_Name = RS("Company_Name")
 RS.Close

 'Get Simulation
 SQL = "SELECT * FROM Simulation"
 RS.Open SQL, Con, 1, 3
 Product_ID = RS("Product_ID")
 End_Date = RS("End_Date")
 Curr_Date = RS("Curr_Date")
 Company_IDs = Split(RS("Customer_IDs"), ";")
 For i = 0 To Ubound(Company_IDs)
  If Company_IDs(i) = Company_ID Then
   If Mid(RS("Customer_Complete"), i + 1, 1) = 0 Then
    Customer_Complete = False
   Else
    Customer_Complete = True
   End If
   Other_Complete = True
   If InStr(Mid(RS("Customer_Complete"), 1, i), "0") > 0 Then Other_Complete = False
   If InStr(Mid(RS("Customer_Complete"), i + 2, Len(RS("Customer_Complete"))), "0") > 0 Then Other_Complete = False
  End If
 Next
 RS.Close

 'Get Curr Inventory
 SQL = "SELECT * FROM Simulation_Details WHERE Company_ID = '" & Company_ID & "' AND [Type] = 'Inventory'
AND [Request_Date] = #" & ConvertDate(Curr_Date) & "#"
 RS.Open SQL, Con, 1, 3
 Curr_Inventory = RS("Detail")
 RS.Close
 'Get Product Name
 SQL = "SELECT * FROM [Products] WHERE [Product_ID] = '" & Product_ID & "'"
 RS.Open SQL, Con, 1, 3
 Product_Name = RS("Product_Name")
 RS.Close

 'Get VMI
 SQL = "SELECT * FROM [VMI] WHERE Company_ID = '" & Company_ID & "' AND [Product_ID] = '" & Product_ID &
"'"
 RS.Open SQL, Con, 1, 3
 Lead_Time = RS("Lead_Time")
 Upper_Lower_Limit = RS("Upper/Lower_Limit")
 Batch_Size = RS("Batch_Size")
 RS.Close

 'Log
 SQL = "SELECT * FROM Simulation_Details WHERE Company_ID = '" & Company_ID & "' AND ([Type] = 'PR' OR
[Type] = 'Delivery' OR ([Type] = 'Demand' AND [Request_Date] >= #" & ConvertDate(Curr_Date) & "# AND [Request_Date]
<= #" & ConvertDate(Curr_Date + 7) & "#)) ORDER BY [Request_Date]"
 RS.Open SQL, Con, 1, 3
 Do Until RS.EOF
  Select Case RS("Type")
   Case "PR"
    Log = Log & ConvertDate(RS("Request_Date")) & ": Requests supplier to deliver " & RS("Detail") & " units on " &
ConvertDate(RS("Deliver_Date")) & VbCrLf
   Case "Delivery"
    Log = Log & ConvertDate(RS("Request_Date")) & ": Supplier delivers " & RS("Detail") & " units" & VbCrLf
   Case "Demand"
    Log = Log & "There will be a demand of " & RS("Detail") & " units on " & ConvertDate(RS("Request_Date")) & VbCrLf
  End Select
  RS.MoveNext
 Loop
 RS.Close
%>
<br>
```

```
<meta http-equiv="refresh" content="10; url=Simulation.asp">
<form name="Frm" method="post" action="">
 <table border="1" cellspacing="0" cellpadding="5">
  <tr>
   <th colspan="4" bgcolor="#ECE9D8" align="center">Current Information</th>
  </tr>
  <tr>
   <td bgcolor="#ECE9D8">Product:</td><td colspan="3"><%=Product_ID%> - <%=Product_Name%></td>
  </tr>
  <tr>
   <td bgcolor="#ECE9D8">Current Date:</td><td><%=ConvertDate(Curr_Date)%></td>
   <td bgcolor="#ECE9D8">Current Inventory:</td><td><%=Curr_Inventory%></td>
  </tr>
  <tr>
   <th colspan="4" bgcolor="#ECE9D8" align="center">Purchase Request To Supplier</th>
  </tr>
  <tr>
   <td bgcolor="#ECE9D8">Batch_Size:</td><td><%=Batch_Size%></td>
   <td bgcolor="#ECE9D8">Upper/Lower Limit:</td><td><%=Upper_Lower_Limit%></td>
  </tr>
  <tr>
   <td bgcolor="#ECE9D8">Delivery Date:</td><td><%=ConvertDate(Curr_Date + Lead_Time)%></td>
   <td bgcolor="#ECE9D8">Qty:</td><td><input type="text" name="PR_Supplier_Qty"></td>
  </tr>
  <tr>
   <td colspan="4" align="right">
    <input type="button" value="Issue Purchase Request" onClick="FrmSend('PR_Supplier_Issue')">
   </td>
  </tr>
  <tr>
   <th colspan="4" bgcolor="#ECE9D8" align="center">History Log</th>
  </tr>
  <tr>
   <td colspan="4"><textarea rows="10" cols="80"><%=Log%></textarea></td>
  </tr>
  <tr>
   <th colspan="4" bgcolor="#ECE9D8" align="center">Simulation Status</th>
  </tr>
  <tr>
   <td colspan="4">
<%
If Customer_Complete = False Then
 Response.Write(Company_Name & " - Not Complete<br>")
Else
 Response.Write(Company_Name & " - OK<br>")
End If
If Other_Complete = False Then
 Response.Write("Supplier - Not Complete<br>")
Else
 Response.Write("Supplier - OK<br>")
End If
%>
   </td>
  </tr>
  <tr align="center">
   <td colspan="4" bgcolor="#ECE9D8">
    <input type="button" value="Show Current Results" onClick="FrmSend('Show_Result')">
    <input type="button" value="Next Day" onClick="FrmSend('Next_Day')">
    <input type="hidden" name="Sim_Status">
   </td>
  </tr>
 </table>
</form>

<script language="vbscript">
Sub FrmSend(InputVal)
<%If Customer_Complete = True Then%>
 Msgbox "You have completed this day."
 Exit Sub
<%End If%>

 Select Case InputVal
  Case "PR_Supplier_Issue"
   If Frm.PR_Supplier_Qty.Value < <%=Batch_Size * (100 - Upper_Lower_Limit) / 100%> Then
    Msgbox "You cannot order less than the lower limit of batch size."
```

```
    Exit Sub
   End If
  Case "Show_Result"
   Frm.Target = "_blank"
 End Select
 Frm.Sim_Status.Value = InputVal
 Frm.Submit
End Sub
</script>
<%
 Con.Close
 Set RS = Nothing
 Set Con = Nothing
End Sub

Sub ShowResult()
 Dim Con, SQL, RS
 Dim Product_ID, Start_Date, End_Date, Curr_Date, Company_ID, Company_Name
 Dim i
 Dim OrderAmt, BackorderAmt, FR

 Set Con = CreateObject("ADODB.Connection")
 Set RS = CreateObject("ADODB.Recordset")
 Con.Open DBCon

 'Get Company Name
 Company_ID = Session("Company_ID")
 SQL = "SELECT * FROM Companys WHERE Company_ID = '" & Company_ID & "'"
 RS.Open SQL, Con, 1, 3
 Company_Name = RS("Company_Name")
 RS.Close

 SQL = "SELECT * FROM Simulation"
 RS.Open SQL, Con, 1, 3
 Product_ID = RS("Product_ID")
 Start_Date = RS("Start_Date")
 End_Date = RS("End_Date")
 Curr_Date = RS("Curr_Date")
 Company_ID = Session("Company_ID")
 RS.Close
%>
<br>
<form name="Frm" method="post" action="">
  <table border="1" cellspacing="0" cellpadding="5">
    <tr>
      <th colspan="4" bgcolor="#ECE9D8" align="center">Simulation Results - <%=Company_Name%></th>
    </tr>
    <tr>
      <td bgcolor="#ECE9D8">Strategy:</td><td>VMI</td>
      <td bgcolor="#ECE9D8">Total Inventory:</td><td><%=GetCountNo(Con, "SELECT SUM(Detail) FROM
Simulation_Details WHERE Type = 'Inventory' AND Company_ID = '" & Company_ID & "'")%></td>
    </tr>
    <tr>
      <td bgcolor="#ECE9D8">Total Order No:</td><td><%=GetCountNo(Con, "SELECT COUNT(*) FROM
Simulation_Details WHERE (Type = 'PR' OR Type = 'Backorder') AND Company_ID = '" & Company_ID & "'")%></td>
<%OrderAmt = GetCountNo(Con, "SELECT SUM(Detail) FROM Simulation_Details WHERE Type = 'PR' AND
Company_ID = '" & Company_ID & "'")%>
      <td bgcolor="#ECE9D8">Total Order Amt:</td><td><%=OrderAmt%></td>
    </tr>
    <tr>
      <td bgcolor="#ECE9D8">Total Backorder No:</td><td><%=GetCountNo(Con, "SELECT Count(*) FROM
Simulation_Details WHERE Type = 'Backorder' AND Company_ID = '" & Company_ID & "'")%></td>
<%BackorderAmt = GetCountNo(Con, "SELECT SUM(Detail) FROM Simulation_Details WHERE Type = 'Backorder' AND
Company_ID = '" & Company_ID & "'")%>
      <td bgcolor="#ECE9D8">Total Backorder Amt:</td><td><%=BackorderAmt%></td>
    </tr>
    <tr>
<%
If OrderAmt = 0 Then
 FR = 0
Else
 FR = Round(100 - BackorderAmt / OrderAmt * 100)
End If
%>
      <td bgcolor="#ECE9D8">Fill Rate:</td><td><%=FR%>%</td>
```

```
    </tr>
  </table>
</form>

<%
 Dim Inventory()
 Redim Inventory(0, Curr_Date - Start_Date)
 SQL = "SELECT * FROM [Simulation_Details] WHERE [Type] = 'Inventory' AND Company_ID = '" & Company_ID & "'
ORDER BY [Request_Date]"
 RS.Open SQL, Con, 1, 3
 Do Until RS.EOF
  Inventory(0, RS("Request_Date") - Start_Date) = RS("Detail")
  RS.MoveNext
 Loop
 RS.Close
%>

<%
Dim XAxis(), Series(), PicLink
Redim XAxis(Curr_Date - Start_Date), Series(0)
For i = 0 To Ubound(XAxis)
 XAxis(i) = ConvertDate(Start_Date + i)
Next
Series(0) = Company_Name
PicLink = "Excel.gif"
Call GenExcelChart(XAxis, Series, Inventory, "Inventory", "Time", "Qty", PicLink)
%>
<br>
<img src="<%=PicLink%>">

<br>
  <table border="1" cellspacing="0" cellpadding="5">
    <tr>
      <th colspan="2" bgcolor="#ECE9D8" align="center">Inventory</th>
    </tr>
<%
 Response.Write("<tr><th>Date</th>")
 Response.Write("<th>" & Company_Name & "</th>")
 For i = 0 To Ubound(Inventory, 2)
  Response.Write("<tr>")
  Response.Write("<td>" & ConvertDate(Start_Date + i) & "</td>")
  Response.Write("<td>" & Inventory(0, i) & "</td>")
  Response.Write("</tr>")
 Next
%>
</table>
<%
 Con.Close
 Set RS = Nothing
 Set Con = Nothing
End Sub

Function GetCountNo(Con, SQL)
 Set RS = CreateObject("ADODB.Recordset")
 RS.Open SQL, Con, 1, 3
 If IsNull(RS(0)) Then GetCountNo = 0 Else GetCountNo = RS(0)
 RS.Close
 Set RS = Nothing
End Function
%>
```

Simulation_Log.inc

```
<!--#include file="Database.inc"-->
<%
 Dim Con, SQL, RS
 Dim Sim_ID, Company_ID
 Dim i

 Set Con = CreateObject("ADODB.Connection")
 Set RS = CreateObject("ADODB.Recordset")
 Con.Open DBCon

 Sim_ID = 0
```

```
 SQL = "SELECT * FROM [Retain_Simulation] WHERE [Product_ID] = '" & Request("Product_ID") & "' And
INSTR([Customer_IDs], '" & Request("Company_ID") & "') > 0 AND (INSTR([Customer_Control], '0') > 0 OR
INSTR([Customer_Control], '1') > 0) ORDER BY [Simulation_ID] DESC"
 RS.Open SQL, Con, 1, 3
 Do Until RS.EOF
  Company_ID = Split(RS("Customer_IDs"), ";")
  For i = 0 To Ubound(Company_ID)
   If Company_ID(i) = Request("Company_ID") Then
    If Mid(RS("Customer_Control"), i + 1, 1) <> 2 Then 'Push
     Sim_ID = RS("Simulation_ID")
%>
  <table border="1" cellspacing="0" cellpadding="5">
    <tr>
      <td colspan="4" bgcolor="#ECE9D8" align="center"><strong>Simulation Parameters</strong></td>
    </tr>
    <tr>
      <td bgcolor="#ECE9D8">Product:</td>
      <td><%=Request("Product_ID")%></td>
      <td bgcolor="#ECE9D8">Company:</td>
      <td><%=Request("Company_ID")%></td>
    </tr>
    <tr>
      <td bgcolor="#ECE9D8">Start Date:</td>
      <td><%=RS("Start_Date")%></td>
      <td bgcolor="#ECE9D8">End Date:</td>
      <td><%=RS("End_Date")%></td>
    </tr>
  </table>
<%
     Exit Do
    End If
   End If
  Next
  RS.MoveNext
 Loop
 RS.Close

 If Sim_ID = 0 Then
  Response.Write("There is no vmi simulation about this company.")
 Else
  SQL = "SELECT * FROM [Retain_Simulation_Details] WHERE [Simulation_ID] = " & Sim_ID & " ORDER BY
[Request_Date]"
  RS.Open SQL, Con, 1, 3
%>
<br>
  <table border="1" cellspacing="0" cellpadding="5">
    <tr>
      <td colspan="4" bgcolor="#ECE9D8" align="center"><strong>Simulation Log</strong></td>
    </tr>
    <tr>
      <td colspan="4">
<%
  Do Until RS.EOF
%>
Company_ID: <%=RS("Company_ID")%>, Request_Date: <%=RS("Request_Date")%>, Type: <%=RS("Type")%>, Qty:
<%=RS("Detail")%><br>
<%
   RS.MoveNext
  Loop
%>
    </td></tr>
  </table>
<%
  RS.Close
 End If

 Con.Close
 Set RS = Nothing
 Set Con = Nothing
%>
```

Simulation_Retailer.asp

```
<html>
```

```
<head><title>VMI Simulation</ title>
</head>

<body topmargin="0" leftmargin="0">
<!-- #include file="top.inc" -->

<table border="0" cellpadding="0" cellspacing="0" style="border-collapse: collapse; border-width: 0" width="100%">
  <tr>
    <td width="17%" style="border-style: none; border-width: medium" valign="top"><!-- #include file="left.inc" --></td>
    <td width="83%" style="border-style: none; border-width: medium" valign="top" align="center"><!-- #include
file="Simulation_Retailer.inc" --></td>
  </tr>
</table>
</body>
</html>
```

Simulation_Retailer.inc

```
<!-- #include file="Redirect.inc" -->
<!--#include file="ConvertDate.inc"-->
<!--#include file="GenExcel.inc"-->
<!--#include file="Database.inc"-->
<%
If Request("Start_Sim") = "True" Then
 Call StartSimulation
 Call LoadRunSimulation
ElseIf Request("Sim_Status") = "Initial_Sim" Then
 Call ClearSimulation
 Call CheckSimulation
ElseIf Request("Sim_Status") = "PR_Supplier_Issue" Then
 Call UpdateSupplierIssue
 Call LoadRunSimulation
ElseIf Request("Sim_Status") = "Next_Day" Then
 Call UpdateComplete
 Call LoadRunSimulation
ElseIf Request("Sim_Status") = "Show_Result" Then
 Call ShowResult
ElseIf Request("Sim_Status") = "DO_Customer_Issue" Then
 Call UpdateCustomerIssue
 Call LoadRunSimulation
ElseIf Request("Sim_Status") = "Retain" Then
 Call RetainSimulation
 Call ShowResult
Else
 Call CheckSimulation
End If

Sub RetainSimulation
 Dim Con, SQL, RS, RS2
 Dim Sim_ID

 Set Con = CreateObject("ADODB.Connection")
 Set RS = CreateObject("ADODB.Recordset")
 Set RS2 = CreateObject("ADODB.Recordset")
 Con.Open DBCon

 SQL = "SELECT * FROM Simulation"
 RS.Open SQL, Con, 1, 3
 SQL = "SELECT * FROM Retain_Simulation ORDER BY [Simulation_ID] DESC"
 RS2.Open SQL, Con, 1, 3
 If RS2.EOF Then
  Sim_ID = 1
 Else
  Sim_ID = RS2("Simulation_ID") + 1
 End If
 RS2.AddNew
 RS2("Simulation_ID") = Sim_ID
 RS2("Product_ID") = RS("Product_ID")
 RS2("Start_Date") = RS("Start_Date")
 RS2("End_Date") = RS("End_Date")
 RS2("Curr_Date") = RS("Curr_Date")
 RS2("Customer_IDs") = RS("Customer_IDs")
 RS2("Customer_Control") = RS("Customer_Control")
 RS2.Update
```

```
 RS2.Close
 RS.Close

 SQL = "SELECT * FROM Simulation_Details"
 RS.Open SQL, Con, 1, 3
 SQL = "SELECT * FROM Retain_Simulation_Details"
 RS2.Open SQL, Con, 1, 3
 Do Until RS.EOF
  RS2.AddNew
  RS2("Simulation_ID") = Sim_ID
  RS2("Company_ID") = RS("Company_ID")
  RS2("Request_Date") = RS("Request_Date")
  RS2("Type") = RS("Type")
  RS2("Detail") = RS("Detail")
  RS2("Deliver_Date") = RS("Deliver_Date")
  RS2.Update
  RS.MoveNext
 Loop
 RS.Close
 RS2.Close

 Con.Close
 Set RS = Nothing
 Set RS2 = Nothing
 Set Con = Nothing
End Sub

Sub UpdateComplete()
 Dim Con, SQL, RS

 Set Con = CreateObject("ADODB.Connection")
 Set RS = CreateObject("ADODB.Recordset")
 Con.Open DBCon

 SQL = "SELECT * FROM Simulation"
 RS.Open SQL, Con, 1, 3
 RS("Customer_Complete") = Left(RS("Customer_Complete"), Len(RS("Customer_Complete")) - 1) & "1"
 RS.Update
 RS.Close

 Con.Close
 Set RS = Nothing
 Set Con = Nothing
End Sub

Sub UpdateCustomerIssue()
 Dim Con, SQL, RS
 Dim Product_ID, Curr_Date, Company_ID

 Set Con = CreateObject("ADODB.Connection")
 Set RS = CreateObject("ADODB.Recordset")
 Con.Open DBCon

 'Get Product ID
 SQL = "SELECT * FROM Simulation"
 RS.Open SQL, Con, 1, 3
 Product_ID = RS("Product_ID")
 Curr_Date = RS("Curr_Date")
 RS.Close

 Company_ID = Request("Company_ID")

 SQL = "SELECT * FROM [Simulation_Details]"
 RS.Open SQL, Con, 1, 3
 RS.AddNew
 RS("Company_ID") = Company_ID
 RS("Request_Date") = ConvertDate(Curr_Date)
 RS("Type") = "Delivery"
 RS("Detail") = Request("DO_Customer_Qty")
 RS.Update
 RS.Close

 'Update Retailer Inventory
 SQL = "SELECT * FROM [Simulation_Details] WHERE Company_ID = '0' AND Type = 'Inventory' AND Request_Date =
#" & ConvertDate(Curr_Date) & "#"
```

```
 RS.Open SQL, Con, 1, 3
 RS("Detail") = RS("Detail") - Request("DO_Customer_Qty")
 RS.Update
 RS.Close

 'Update Customer Inventory
 SQL = "SELECT * FROM [Simulation_Details] WHERE Company_ID = '" & Company_ID & "' AND Type = 'Inventory'
AND Request_Date = #" & ConvertDate(Curr_Date) & "#"
 RS.Open SQL, Con, 1, 3
 RS("Detail") = RS("Detail") + Request("DO_Customer_Qty")
 RS.Update
 RS.Close

 Con.Close
 Set RS = Nothing
 Set Con = Nothing
End Sub

Sub UpdateSupplierIssue()
 Dim Con, SQL, RS
 Dim Product_ID, Curr_Date, Lead_Time

 Set Con = CreateObject("ADODB.Connection")
 Set RS = CreateObject("ADODB.Recordset")
 Con.Open DBCon

 'Get Product ID
 SQL = "SELECT * FROM Simulation"
 RS.Open SQL, Con, 1, 3
 Product_ID = RS("Product_ID")
 Curr_Date = RS("Curr_Date")
 RS.Close

 'Get Lead Time
 SQL = "SELECT * FROM Products WHERE Product_ID = '" & Product_ID & "'"
 RS.Open SQL, Con, 1, 3
 Lead_Time = RS("Lead_Time")
 RS.Close

 SQL = "SELECT * FROM [Simulation_Details]"
 RS.Open SQL, Con, 1, 3
 RS.AddNew
 RS("Company_ID") = "0"
 RS("Request_Date") = ConvertDate(Curr_Date)
 RS("Type") = "PR"
 RS("Detail") = Request("PR_Supplier_Qty")
 RS("Deliver_Date") = ConvertDate(Curr_Date + Lead_Time)
 RS.Update
 RS.Close

 Con.Close
 Set RS = Nothing
 Set Con = Nothing
End Sub

Sub CheckSimulation()
 Dim Con, SQL, RS

 Set Con = CreateObject("ADODB.Connection")
 Set RS = CreateObject("ADODB.Recordset")
 Con.Open DBCon

 SQL = "SELECT * FROM [Simulation]"
 RS.Open SQL, Con, 1, 3

 If RS.EOF Then
  Call LoadCreateNewSimulation
 Else
  Call LoadRunSimulation
 End If
 RS.Close

 Con.Close
 Set RS = Nothing
 Set Con = Nothing
```

```
End Sub

Sub ClearSimulation()
 Dim Con, SQL, RS
 Dim i, TmpStr

 Set Con = CreateObject("ADODB.Connection")
 Set RS = CreateObject("ADODB.Recordset")
 Con.Open DBCon

 SQL = "DELETE * FROM [Simulation]"
 RS.Open SQL, Con, 1, 3
 SQL = "DELETE * FROM [Simulation_Details]"
 RS.Open SQL, Con, 1, 3

 Con.Close
 Set RS = Nothing
 Set Con = Nothing
End Sub

Sub StartSimulation()
 Dim Con, SQL, RS, RS2
 Dim i, TmpStr, TmpLng
 Dim Company_ID, Customer_Control(), Product_ID, Curr_Date
 Dim Lead_Time, Batch_Size, Demand

 Set Con = CreateObject("ADODB.Connection")
 Set RS = CreateObject("ADODB.Recordset")
 Set RS2 = CreateObject("ADODB.Recordset")
 Con.Open DBCon

 'Ini Simulation Parameters
 SQL = "DELETE * FROM [Simulation]"
 RS.Open SQL, Con, 1, 3
 SQL = "DELETE * FROM [Simulation_Details]"
 RS.Open SQL, Con, 1, 3
 SQL = "SELECT * FROM [Forecast Parameters]"
 RS.Open SQL, Con, 1, 3
 Do Until RS.EOF
  RS("Coef") = RS("Ini_Coef")
  RS("Next_Forecast") = RS("Ini_Next_Forecast")
  RS.Update
  RS.MoveNext
 Loop
 RS.Close

 SQL = "SELECT * FROM [Simulation]"
 RS.Open SQL, Con, 1, 3
 RS.AddNew
 RS("Product_ID") = Request("Product_ID")
 Product_ID = Request("Product_ID")
 RS("Start_Date") = Request("Start_Date")
 RS("End_Date") = Request("End_Date")
 RS("Curr_Date") = Request("Start_Date")
 Curr_Date = CDate(Request("Start_Date"))
 RS("Customer_IDs") = Request("Customer_IDs")
 Company_ID = Split(Request("Customer_IDs"), ";")
 TmpStr = ""
 Redim Customer_Control(Request("Customer_Count"))
 For i = 0 To Request("Customer_Count")
  TmpStr = TmpStr & Request("Company_Str_" & i)
  Customer_Control(i) = Request("Company_Str_" & i)
 Next
 RS("Customer_Control") = TmpStr
 TmpStr = ""
 For i = 0 To Request("Customer_Count")
  If Customer_Control(i) = 0 Then
   TmpStr = TmpStr & "0"
  Else
   TmpStr = TmpStr & "1"
  End If
 Next
 RS("Customer_Complete") = TmpStr & "0"
 RS.Update
 RS.Close
```

```
'Set Retailer Ini Stock
SQL = "SELECT [Inventory] FROM [Products] WHERE [Product_ID] = '" & Request("Product_ID") & "'"
RS.Open SQL, Con, 1, 3
TmpLng = RS(0)
RS.Close
SQL = "SELECT * FROM [Simulation_Details]"
RS.Open SQL, Con, 1, 3
RS.AddNew
RS("Company_ID") = "0"
RS("Request_Date") = Request("Start_Date")
RS("Type") = "Inventory"
RS("Detail") = TmpLng
RS.Update
RS.Close

'Set Customer Ini Stock
For i = 0 To Ubound(Company_ID)
 If Customer_Control(i) = 2 Then
  SQL = "SELECT [Initial_Stock] FROM [Push] WHERE [Company_ID] = '" & Company_ID(i) & "' AND [Product_ID] = '"
& Request("Product_ID") & "'"
 Else
  SQL = "SELECT [Initial_Stock] FROM [VMI] WHERE [Company_ID] = '" & Company_ID(i) & "' AND [Product_ID] = '"
& Request("Product_ID") & "'"
 End If
 RS.Open SQL, Con, 1, 3
 TmpLng = RS(0)
 RS.Close
 SQL = "SELECT * FROM [Simulation_Details]"
 RS.Open SQL, Con, 1, 3
 RS.AddNew
 RS("Company_ID") = Company_ID(i)
 RS("Request_Date") = Request("Start_Date")
 RS("Type") = "Inventory"
 RS("Detail") = TmpLng
 RS.Update
 RS.Close
Next

'Set Customer Demand
For i = 0 To Ubound(Company_ID)
 If Customer_Control(i) = 2 Then 'Push
  SQL = "SELECT [Lead_Time] FROM [Push] WHERE [Company_ID] = '" & Company_ID(i) & "' AND [Product_ID] = '" &
Request("Product_ID") & "'"
 Else
  SQL = "SELECT [Lead_Time] FROM [VMI] WHERE [Company_ID] = '" & Company_ID(i) & "' AND [Product_ID] = '" &
Request("Product_ID") & "'"
 End If
 RS.Open SQL, Con, 1, 3
 Lead_Time = RS(0)
 RS.Close
 SQL = "SELECT * FROM [Sales Demand] WHERE [Company_ID] = '" & Company_ID(i) & "' AND [Product_ID] = '" &
Request("Product_ID") & "' AND [Date] >= #" & CDate(Request("Start_Date")) + Lead_Time & "# AND [Date] <= #" &
Request("End_Date") & "# ORDER BY [Date]"
 RS.Open SQL, Con, 1, 3
 Do Until RS.EOF
  SQL = "SELECT * FROM [Simulation_Details]"
  RS2.Open SQL, Con, 1, 3
  RS2.AddNew
  RS2("Company_ID") = Company_ID(i)
  RS2("Type") = "Demand"
  RS2("Detail") = RS("Qty")
  RS2("Request_Date") = RS("Date")
  RS2.Update
  RS2.Close
  RS.MoveNext
 Loop
 RS.Close
Next

'Auto Customer Request PR
For i = 0 To Ubound(Company_ID)
 If Customer_Control(i) = 2 Then 'Auto
  'Get Lead_Time
  If Customer_Control(i) = 1 Then 'VMI (Useless)
```

```
    SQL = "SELECT * FROM [VMI] WHERE Company_ID = '" & Company_ID(i) & "' AND Product_ID = '" & Product_ID &
"'"
    RS.Open SQL, Con, 1, 3
    Lead_Time = RS("Lead_Time")
    Batch_Size = RS("Batch_Size") * (100 - RS("Upper/Lower_Limit")) / 100
    RS.Close
   Else
    SQL = "SELECT * FROM [Push] WHERE Company_ID = '" & Company_ID(i) & "' AND Product_ID = '" & Product_ID &
"'"
    RS.Open SQL, Con, 1, 3
    Lead_Time = RS("Lead_Time")
    Batch_Size = 0
    RS.Close
   End If

   'Check Demand
   SQL = "SELECT Detail FROM [Simulation_Details] WHERE Company_ID = '" & Company_ID(i) & "' AND Type =
'Demand' AND Request_Date = #" & ConvertDate(Curr_Date + Lead_Time) & "#"
   RS.Open SQL, Con, 1, 3
   Demand = 0
   If RS.EOF = False Then
    Demand = RS("Detail")
   End If
   RS.Close
   If Demand > 0 Then
    SQL = "SELECT Detail FROM [Simulation_Details] WHERE Company_ID = '" & Company_ID(i) & "' AND Type =
'Inventory' AND Request_Date = #" & ConvertDate(Curr_Date) & "#"
    RS.Open SQL, Con, 1, 3
    TmpLng = RS(0)
    RS.Close
    If Demand > TmpLng Then 'Need To Order
     If Batch_Size = 0 Or Batch_Size < Demand - TmpLng Then
      Batch_Size = Demand - TmpLng
     End If
     SQL = "SELECT * FROM [Simulation_Details]"
     RS.Open SQL, Con, 1, 3
     RS.AddNew
     RS("Company_ID") = Company_ID(i)
     RS("Request_Date") = ConvertDate(Curr_Date)
     RS("Type") = "PR"
     RS("Detail") = Batch_Size
     RS("Deliver_Date") = ConvertDate(Curr_Date + Lead_Time)
     RS.Update
     RS.Close
    End If
   End If
  End If
 Next

 Con.Close
 Set RS = Nothing
 Set RS2 = Nothing
 Set Con = Nothing
End Sub

Sub LoadCreateNewSimulation()
 Dim Con, SQL, RS
 Dim i, TmpStr
 Dim Products(), Product_ID()
 Dim Companys(), Company_ID()
 Dim Start_Date, End_Date

 Set Con = CreateObject("ADODB.Connection")
 Set RS = CreateObject("ADODB.Recordset")
 Con.Open DBCon

 SQL = "SELECT [Product_ID], [Product_Name] FROM [Products]"
 RS.Open SQL, Con, 1, 3
 Redim Products(RS.RecordCount - 1), Product_ID(RS.RecordCount - 1)
 i = 0
 Do Until RS.EOF
  Products(i) = RS("Product_ID") & " - " & RS("Product_Name")
  Product_ID(i) = RS("Product_ID")
  i = i + 1
  RS.MoveNext
```

```
 Loop
 RS.Close

 SQL = "SELECT [Company_ID], [Company_Name] FROM Q_Simulation WHERE Product_ID = '" &
Request("Product_ID") & "' ORDER BY [Company_ID]"
 RS.Open SQL, Con, 1, 3
 Redim Companys(RS.RecordCount - 1), Company_ID(RS.RecordCount - 1)
 i = 0
 Do Until RS.EOF
  Companys(i) = RS("Company_ID") & " - " & RS("Company_Name")
  Company_ID(i) = RS("Company_ID")
  i = i + 1
  RS.MoveNext
 Loop
 RS.Close

 SQL = "SELECT Date FROM [Q_Sales Demand] WHERE [Product_ID] = '" & Request("Product_ID") & "' ORDER BY
[Date]"
 RS.Open SQL, Con, 1, 3
 If RS.EOF = False Then
  Start_Date = ConvertDate(RS(0) - 10)
  RS.MoveLast
  End_Date = ConvertDate(RS(0) + 10)
 End If
 RS.Close

 Con.Close
 Set RS = Nothing
 Set Con = Nothing
%>
<br>
<form name="Frm" method="post" action="">
  <table border="1" cellspacing="0" cellpadding="5">
    <tr>
      <td colspan="4" bgcolor="#ECE9D8" align="center"><strong>Initial A New Simulation</strong></td>
    </tr>
    <tr>
      <td bgcolor="#ECE9D8">Product:</td>
      <td colspan="3">
                <select name="Product_ID" onChange="window.location.href='?Product_ID=' + this[this.selectedIndex].value;">
        <option>--- Please Select ---</option>
<%
For i = 0 To Ubound(Product_ID)
 Response.Write("<option value=""" & Product_ID(i) & """")
 If Request("Product_ID") = Product_ID(i) Then  Response.Write(" selected")
 Response.Write(">" & Products(i) & "</option>")
Next
%>
      </select>
                </td>
    </tr>
    <tr>
      <td colspan="4" bgcolor="#ECE9D8" align="center"><strong>Customer(s) of this product</strong></td>
    </tr>
<%
TmpStr = ""
For i = 0 To Ubound(Company_ID)
 TmpStr = TmpStr & Company_ID(i) & ";"
 Response.Write("<tr><td colspan=""3"">" & Companys(i) & "</td>")
 Response.Write("<td><select name=""Company_Str_" & i & """><option value=""0"">VMI - Manual</option><option
value=""1"">VMI - Auto</option><option value=""2"" selected>Push - Auto</option></select></td></tr>")
Next
If Len(TmpStr) > 0 Then TmpStr = Left(TmpStr, Len(TmpStr) - 1)
%>
      <input type="hidden" name="Customer_IDs" value="<%=TmpStr%>">
    <tr>
      <td colspan="4" bgcolor="#ECE9D8" align="center"><strong>Simulation Parameters</strong></td>
    </tr>
    <tr>
      <td bgcolor="#ECE9D8">Start Date:</td>
      <td><input type="text" name="Start_Date" value="1-Jan-2006"></td>
      <td bgcolor="#ECE9D8">End Date:</td>
      <td><input type="text" name="End_Date" value="31-Dec-2006"></td>
    </tr>
```

```
    <tr align="center">
      <td colspan="4" bgcolor="#ECE9D8">
        <input type="submit" value="Start Simulation">
        <input type="reset" value="Reset">
        <input type="hidden" value="True" name="Start_Sim">
        <input type="hidden" value="<%=Ubound(Company_ID)%>" name="Customer_Count">
      </td>
    </tr>
  </table>
</form>
<%
End Sub

Function CheckAllComplete()
 Dim Con, SQL, RS

 Set Con = CreateObject("ADODB.Connection")
 Set RS = CreateObject("ADODB.Recordset")
 Con.Open DBCon

 SQL = "SELECT [Customer_Complete] FROM Simulation"
 RS.Open SQL, Con, 1, 3

 If InStr(RS("Customer_Complete"), "0") > 0 Then
  CheckAllComplete = False
 Else
  CheckAllComplete = True
 End If
 RS.Close

 Con.Close
 Set RS = Nothing
 Set Con = Nothing
End Function

Sub LoadRunSimulation()
 If CheckAllComplete = True Then
  Call NextDay
  Call LoadRunSimulation
  Exit Sub
 End If

 Dim Con, SQL, RS
 Dim Product_ID, End_Date, Curr_Date, Company_ID, Company_Name(), Customer_Control(), Customer_Complete()
 Dim Curr_Inventory, Log
 Dim Product_Name, Lead_Time, Max_Supplied_Qty, Remain_Supply_Qty, Reorder_Level, Safety_Stock
 Dim i

 Set Con = CreateObject("ADODB.Connection")
 Set RS = CreateObject("ADODB.Recordset")
 Con.Open DBCon

 SQL = "SELECT * FROM Simulation"
 RS.Open SQL, Con, 1, 3
 Product_ID = RS("Product_ID")
 End_Date = RS("End_Date")
 Curr_Date = RS("Curr_Date")
 Company_ID = Split(RS("Customer_IDs"), ";")
 Redim Customer_Control(Ubound(Company_ID))
 For i = 0 To Ubound(Customer_Control)
  Customer_Control(i) = Mid(RS("Customer_Control"), i + 1, 1)
 Next
 Redim Customer_Complete(Ubound(Company_ID) + 1)
 For i = 0 To Ubound(Customer_Complete)
  Customer_Complete(i) = Mid(RS("Customer_Complete"), i + 1, 1)
 Next
 RS.Close

 If Curr_Date > End_Date Then
  Call ShowResult
  Con.Close
  Set RS = Nothing
  Set Con = Nothing
  Exit Sub
 End If
```

```
 'Get Company Name
 Redim Company_Name(Ubound(Company_ID))
 For i = 0 To Ubound(Company_ID)
  SQL = "SELECT * FROM Companys WHERE Company_ID = '" & Company_ID(i) & "'"
  RS.Open SQL, Con, 1, 3
  Company_Name(i) = RS("Company_Name")
  RS.Close
 Next

 'Get Curr Inventory
 SQL = "SELECT * FROM Simulation_Details WHERE Company_ID = '0' AND [Type] = 'Inventory' AND [Request_Date] =
#" & ConvertDate(Curr_Date) & "#"
 RS.Open SQL, Con, 1, 3
 Curr_Inventory = RS("Detail")
 RS.Close

 'Get Supplier Details
 SQL = "SELECT * FROM [Products] WHERE [Product_ID] = '" & Product_ID & "'"
 RS.Open SQL, Con, 1, 3
 Product_Name = RS("Product_Name")
 Lead_Time = RS("Lead_Time")
 Max_Supplied_Qty = RS("Max_Supplied_Qty")
 Reorder_Level = RS("Reorder_Level")
 Safety_Stock = RS("Safety_Stock")
 RS.Close
 SQL = "SELECT SUM(Detail) FROM Simulation_Details WHERE [Type] = 'PR' AND Company_ID = '0'"
 RS.Open SQL, Con, 1, 3
 If IsNull(RS(0)) = True Then
  Remain_Supply_Qty = Max_Supplied_Qty
 Else
  Remain_Supply_Qty = Max_Supplied_Qty - RS(0)
 End If
 RS.Close

 'Log
 SQL = "SELECT * FROM Simulation_Details WHERE [Type] = 'PR' OR [Type] = 'Delivery' ORDER BY [Request_Date]"
 RS.Open SQL, Con, 1, 3
 Do Until RS.EOF
  Select Case RS("Type")
   Case "PR"
    If RS("Company_ID") = "0" Then
     Log = Log & ConvertDate(RS("Request_Date")) & ": Retailer requests supplier to deliver " & RS("Detail") & " units on " &
ConvertDate(RS("Deliver_Date")) & VbCrLf
    Else
     Log = Log & ConvertDate(RS("Request_Date")) & ": " & GetCompanyName(RS("Company_ID"), Company_ID,
Company_Name) & " requests retailer to deliver " & RS("Detail") & " units on " & ConvertDate(RS("Deliver_Date")) &
VbCrLf
    End If
   Case "Delivery"
    If RS("Company_ID") = "0" Then
     Log = Log & ConvertDate(RS("Request_Date")) & ": Supplier delivers " & RS("Detail") & " units to retailer" & VbCrLf
    Else
     Log = Log & ConvertDate(RS("Request_Date")) & ": Retailer delivers " & RS("Detail") & " units to " &
GetCompanyName(RS("Company_ID"), Company_ID, Company_Name) & VbCrLf
    End If
  End Select
  RS.MoveNext
 Loop
 RS.Close
%>
<br>
<!--<meta http-equiv="refresh" content="10; url=Simulation_Retailer.asp">-->
<form name="Frm" method="post" action="">
  <table border="1" cellspacing="0" cellpadding="5">
    <tr>
      <th colspan="4" bgcolor="#ECE9D8" align="center">Current Information</th>
    </tr>
    <tr>
      <td bgcolor="#ECE9D8">Product:</td><td colspan="3"><%=Product_ID%> - <%=Product_Name%></td>
    </tr>
    <tr>
      <td bgcolor="#ECE9D8">Current Date:</td><td><%=ConvertDate(Curr_Date)%></td>
      <td bgcolor="#ECE9D8">Current Inventory:</td><td><%=Curr_Inventory%></td>
    </tr>
```

```
    <tr>
      <td bgcolor="#ECE9D8">Reorder Level:</td><td><%=Reorder_Level%></td>
      <td bgcolor="#ECE9D8">Safety Level:</td><td><%=Safety_Stock%></td>
    </tr>
    <tr>
      <th colspan="4" bgcolor="#ECE9D8" align="center">Delivery To Customer</th>
    </tr>
    <tr>
      <td bgcolor="#ECE9D8">Company:</td>
      <td><select name="Company_ID" onChange="window.location.href='?Company_ID=' + this[this.selectedIndex].value;">
          <option>--- Please Select ---</option>
<%
For i = 0 To Ubound(Company_ID)
 Response.Write("<option value=""" & Company_ID(i) & """")
 If Request("Company_ID") = Company_ID(i) Then Response.Write(" selected")
 Response.Write(">" & Company_Name(i) & "</option>")
Next
%>
      </select></td>
      <td bgcolor="#ECE9D8">Deliver Qty:</td><td><input type="Text" name="DO_Customer_Qty"></td>
    </tr>
    <tr>
      <td colspan="4" align="right">
        <input type="button" value="Issue Purchase Request" onClick="FrmSend('DO_Customer_Issue')">
        <input type="button" value="Customer Details" onClick="window.open('Customer_Detail.asp?Company_ID=' +
Frm.Company_ID[Frm.Company_ID.selectedIndex].value,'_blank')">
      </td>
    </tr>
    <tr>
      <th colspan="4" bgcolor="#ECE9D8" align="center">Purchase Request To Supplier</th>
    </tr>
    <tr>
      <td bgcolor="#ECE9D8">Qty:</td><td><input type="text" name="PR_Supplier_Qty"></td>
      <td bgcolor="#ECE9D8">Delivery Date:</td><td><%=ConvertDate(Curr_Date + Lead_Time)%></td>
    </tr>
    <tr>
      <td colspan="4" align="right">
        <input type="button" value="Issue Purchase Request" onClick="FrmSend('PR_Supplier_Issue')">
      </td>
    </tr>
    <tr align="center">
      <td colspan="4" bgcolor="#ECE9D8">
        <input type="button" value="Initial A New Simulation" onClick="FrmSend('Initial_Sim')">
        <input type="button" value="Show Current Results" onClick="FrmSend('Show_Result')">
        <input type="button" value="Next Day" onClick="FrmSend('Next_Day')">
        <input type="hidden" name="Sim_Status">
      </td>
    </tr>
    <tr>
      <th colspan="4" bgcolor="#ECE9D8" align="center">History Log</th>
    </tr>
    <tr>
      <td colspan="4"><textarea rows="10" cols="80"><%=Log%></textarea></td>
    </tr>
    <tr>
      <th colspan="4" bgcolor="#ECE9D8" align="center">Simulation Status</th>
    </tr>
    <tr>
      <td colspan="4">
<%
If Customer_Complete(Ubound(Customer_Complete)) = "0" Then
 Response.Write("Retailer - Not Complete<br>")
Else
 Response.Write("Retailer - OK<br>")
End If
For i = 0 To Ubound(Customer_Complete) - 1
 If Customer_Complete(i) = "0" Then
  Response.Write(Company_Name(i) & " - Not Complete<br>")
 Else
  Response.Write(Company_Name(i) & " - OK<br>")
 End If
Next
%>
      </td>
    </tr>
```

```
  </table>
</form>

<script language="vbscript">
Sub FrmSend(InputVal)
<%If Customer_Complete(Ubound(Customer_Complete)) = "1" Then%>
 If InputVal <> "Initial_Sim" Then
  Msgbox "You have completed this day."
  Exit Sub
 End If
<%End If%>

 Select Case InputVal
  Case "PR_Supplier_Issue"
   If Frm.PR_Supplier_Qty.Value > <%=Remain_Supply_Qty%> Then
    Msgbox "You cannot order more than the remaining supply qty. The remaining supply qty is <%=Remain_Supply_Qty%>."
    Exit Sub
   End If
   Frm.Target = "_self"
  Case "Show_Result"
   Frm.Target = "_blank"
  Case "DO_Customer_Issue"
   If Frm.DO_Customer_Qty.Value > <%=Curr_Inventory%> Then
    Msgbox "Your inventory cannot fullfill the delivery qty."
    Exit Sub
   End If
   Frm.Target = "_self"
  Case Else
   Frm.Target = "_self"
 End Select
 Frm.Sim_Status.Value = InputVal
 Frm.Submit
End Sub
</script>
<%
 Con.Close
 Set RS = Nothing
 Set Con = Nothing
End Sub

Function GetCompanyName(ID, Company_ID(), Company_Name())
 If ID = "0" Then
  GetCompanyName = "Retailer"
  Exit Function
 End If

 Dim i

 For i = 0 To Ubound(Company_ID)
  If Company_ID(i) = ID Then
   GetCompanyName = Company_Name(i)
   Exit For
  End If
 Next
End Function

Sub NextDay()
 Dim Con, SQL, RS, RS2
 Dim Product_ID, End_Date, Curr_Date, Company_ID, Company_Name(), Customer_Control(), Customer_Complete()
 Dim Curr_Inventory
 Dim i, TmpStr, TmpLng
 Dim Lead_Time, Batch_Size, Demand

 Set Con = CreateObject("ADODB.Connection")
 Set RS = CreateObject("ADODB.Recordset")
 Set RS2 = CreateObject("ADODB.Recordset")
 Con.Open DBCon

 SQL = "SELECT * FROM Simulation"
 RS.Open SQL, Con, 1, 3
 Product_ID = RS("Product_ID")
 End_Date = RS("End_Date")
 Curr_Date = RS("Curr_Date") + 1
 RS("Curr_Date") = Curr_Date
 Company_ID = Split(RS("Customer_IDs"), ";")
```

```
Redim Customer_Control(Ubound(Company_ID))
For i = 0 To Ubound(Customer_Control)
 Customer_Control(i) = Mid(RS("Customer_Control"), i + 1, 1)
Next
Redim Customer_Complete(Ubound(Company_ID) + 1)
TmpStr = ""
For i = 0 To Ubound(Customer_Control)
 If Customer_Control(i) = "0" Then
  TmpStr = TmpStr & "0"
 Else
  TmpStr = TmpStr & "1"
 End If
Next
RS("Customer_Complete") = TmpStr & "0"
RS.Update
RS.Close

'Update Customer Inventory - Demand
For i = 0 To Ubound(Company_ID)
 SQL = "SELECT SUM(Detail) FROM [Simulation_Details] WHERE Company_ID = '" & Company_ID(i) & "' AND Type =
'Demand' AND Request_Date = #" & ConvertDate(Curr_Date - 1) & "#"
 RS.Open SQL, Con, 1, 3
 If IsNull(RS(0)) Then
  TmpLng = 0
 Else
  TmpLng = RS(0)
 End If
 RS.Close
 If TmpLng > 0 Then 'Have Demand
  SQL = "SELECT * FROM [Simulation_Details] WHERE Company_ID = '" & Company_ID(i) & "' AND Type = 'Inventory'
AND Request_Date = #" & ConvertDate(Curr_Date - 1) & "#"
  RS.Open SQL, Con, 1, 3
  If RS("Detail") > TmpLng Then
   RS("Detail") = RS("Detail") - TmpLng
   TmpLng = 0
  Else 'BackOrder
   TmpLng = TmpLng - RS("Detail")
   RS("Detail") = 0
  End If
  RS.Update
  RS.Close
 End If

 If TmpLng > 0 Then 'BackOrder
  SQL = "SELECT * FROM [Simulation_Details]"
  RS.Open SQL, Con, 1, 3
  RS.AddNew
  RS("Company_ID") = Company_ID(i)
  RS("Request_Date") = ConvertDate(Curr_Date - 1)
  RS("Type") = "BackOrder"
  RS("Detail") = TmpLng
  RS.Update
  RS.Close
 End If
Next

If Curr_Date > End_Date Then
 Con.Close
 Set RS = Nothing
 Set RS2 = Nothing
 Set Con = Nothing
 Exit Sub
Else
 'Update Retailer Inventory
 SQL = "SELECT * FROM [Simulation_Details] WHERE Company_ID = '0' AND Type = 'PR' AND Deliver_Date = #" &
ConvertDate(ConvertDate(Curr_Date)) & "#"
 RS.Open SQL, Con, 1, 3
 TmpLng = 0
 Do Until RS.EOF
  SQL = "SELECT * FROM [Simulation_Details]"
  RS2.Open SQL, Con, 1, 3
  RS2.AddNew
  RS2("Company_ID") = "0"
  RS2("Request_Date") = ConvertDate(Curr_Date)
  RS2("Type") = "Delivery"
```

```
   RS2("Detail") = RS("Detail")
   TmpLng = TmpLng + RS("Detail")
   RS2.Update
   RS2.Close
   RS.MoveNext
  Loop
  RS.Close
  SQL = "SELECT * FROM [Simulation_Details] WHERE Company_ID = '0' AND Type = 'Inventory' AND Request_Date = #" & ConvertDate(Curr_Date - 1) & "#"
  RS.Open SQL, Con, 1, 3
  Curr_Inventory = RS("Detail")
  RS.Close
  SQL = "SELECT * FROM [Simulation_Details]"
  RS.Open SQL, Con, 1, 3
  RS.AddNew
  RS("Company_ID") = "0"
  RS("Request_Date") = ConvertDate(Curr_Date)
  RS("Type") = "Inventory"
  RS("Detail") = Curr_Inventory + TmpLng
  RS.Update
  RS.Close

  'Update Customer Inventory
  For i = 0 To Ubound(Company_ID)
   SQL = "SELECT Detail FROM [Simulation_Details] WHERE Company_ID = '" & Company_ID(i) & "' AND Type = 'Inventory' AND Request_Date = #" & ConvertDate(Curr_Date - 1) & "#"
   RS.Open SQL, Con, 1, 3
   TmpLng = RS(0)
   RS.Close
   SQL = "SELECT * FROM [Simulation_Details]"
   RS.Open SQL, Con, 1, 3
   RS.AddNew
   RS("Company_ID") = Company_ID(i)
   RS("Request_Date") = ConvertDate(Curr_Date)
   RS("Type") = "Inventory"
   RS("Detail") = TmpLng
   RS.Update
   RS.Close
  Next
 End If

 'Auto Customer Request PR
 For i = 0 To Ubound(Company_ID)
  If Customer_Control(i) = 2 Then 'Auto Push
   'Get Lead_Time
   If Customer_Control(i) = 1 Then 'VMI (Now No Use)
    SQL = "SELECT * FROM [VMI] WHERE Company_ID = '" & Company_ID(i) & "' AND Product_ID = '" & Product_ID & "'"
    RS.Open SQL, Con, 1, 3
    Lead_Time = RS("Lead_Time")
    Batch_Size = RS("Batch_Size") * (100 - RS("Upper/Lower_Limit")) / 100
    RS.Close
   Else
    SQL = "SELECT * FROM [Push] WHERE Company_ID = '" & Company_ID(i) & "' AND Product_ID = '" & Product_ID & "'"
    RS.Open SQL, Con, 1, 3
    Lead_Time = RS("Lead_Time")
    Batch_Size = 0
    RS.Close
   End If

   'Check Demand
   SQL = "SELECT SUM(Detail) FROM [Simulation_Details] WHERE Company_ID = '" & Company_ID(i) & "' AND Type = 'Demand' AND Request_Date = #" & ConvertDate(Curr_Date + Lead_Time) & "#"
   RS.Open SQL, Con, 1, 3
   Demand = 0
   If IsNull(RS(0)) = False Then
    Demand = RS(0)
   End If
   RS.Close
   If Demand > 0 Then
    SQL = "SELECT Detail FROM [Simulation_Details] WHERE Company_ID = '" & Company_ID(i) & "' AND Type = 'Inventory' AND Request_Date = #" & ConvertDate(Curr_Date) & "#"
    RS.Open SQL, Con, 1, 3
    TmpLng = RS(0)
```

```
   RS.Close
   If Demand > TmpLng Then 'Need To Order
    If Batch_Size = 0 Or Batch_Size < Demand - TmpLng Then
     Batch_Size = Demand - TmpLng
    End If
    SQL = "SELECT * FROM [Simulation_Details]"
    RS.Open SQL, Con, 1, 3
    RS.AddNew
    RS("Company_ID") = Company_ID(i)
    RS("Request_Date") = ConvertDate(Curr_Date)
    RS("Type") = "PR"
    RS("Detail") = Batch_Size
    RS("Deliver_Date") = ConvertDate(Curr_Date + Lead_Time)
    RS.Update
    RS.Close
   End If
  End If
 End If
Next

'Update Forecast Coef
If Day(Curr_Date) = 1 Then
 Dim Actual, Coef, CoefStr, ForecastError, NextForecast, Y()
 For i = 0 To Ubound(Company_ID)
  SQL = "SELECT * FROM [Forecast Parameters] WHERE Company_ID = '" & Company_ID(i) & "' AND Product_ID = '" &
Product_ID & "'"
  RS.Open SQL, Con, 1, 3
  If RS.EOF = False Then
   SQL = "SELECT SUM(Qty) FROM [Sales Demand] WHERE [Date] > #" & ConvertDate(DateAdd("m", -1, Curr_Date)) &
"# AND [Date] <= #" & ConvertDate(Curr_Date - 1) & "# AND [Company_ID] = '" & Company_ID(i) & "' AND
[Product_ID] = '" & Product_ID & "'"
   RS2.Open SQL, Con, 1, 3
   If IsNull(RS2(0)) Then
    Actual = 0
   Else
    Actual = RS2(0)
   End If
   RS2.Close

   ForecastError = Actual - RS("Next_Forecast")
   Coef = Split(RS("Coef"), ";")
   For j = 0 To Ubound(Coef)
    Coef(j) = Coef(j) + RS("Beta") * ForecastError * Actual
    CoefStr = CoefStr & Coef(j) & ";"
   Next
   CoefStr = Left(CoefStr, Len(CoefStr) - 1)

   Redim Y(Ubound(Coef))

   For j = 0 To Ubound(Coef)
    SQL = "SELECT SUM(Qty) FROM [Sales Demand] WHERE [Date] > #" & ConvertDate(DateAdd("m", -(j + 1),
Curr_Date)) & "# AND [Date] <= #" & ConvertDate(DateAdd("m", -j, Curr_Date) - 1) & "# AND [Company_ID] = '" &
Company_ID(i) & "' AND [Product_ID] = '" & Product_ID & "'"
    RS2.Open SQL, Con, 1, 3
    If IsNull(RS2(0)) Then
     Y(j) = 0
    Else
     Y(j) = RS2(0)
    End If
    RS2.Close
   Next

   NextForecast = 0
   For j = 0 To Ubound(Coef)
    NextForecast = NextForecast + Coef(j) * Y(j)
   Next

   RS("Coef") = CoefStr
   RS("Next_Forecast") = NextForecast
   RS.Update
  End If
  RS.Close
 Next
End If
```

```
 Con.Close
 Set RS = Nothing
 Set RS2 = Nothing
 Set Con = Nothing
End Sub

Sub ShowResult()
 Dim Con, SQL, RS
 Dim Product_ID, Start_Date, End_Date, Curr_Date, Company_ID, Company_Name(), Customer_Control(),
Customer_Complete()
 Dim i, j
 Dim OrderNo, OrderFreq, OrderAmt, BackorderAmt, FR

 Set Con = CreateObject("ADODB.Connection")
 Set RS = CreateObject("ADODB.Recordset")
 Con.Open DBCon

 SQL = "SELECT * FROM Simulation"
 RS.Open SQL, Con, 1, 3
 Product_ID = RS("Product_ID")
 Start_Date = RS("Start_Date")
 End_Date = RS("End_Date")
 Curr_Date = RS("Curr_Date")
 Company_ID = Split(RS("Customer_IDs"), ";")
 Redim Customer_Control(Ubound(Company_ID))
 For i = 0 To Ubound(Customer_Control)
  Customer_Control(i) = Mid(RS("Customer_Control"), i + 1, 1)
 Next
 RS.Close

 'Get Company Name
 Redim Company_Name(Ubound(Company_ID))
 For i = 0 To Ubound(Company_ID)
  SQL = "SELECT * FROM Companys WHERE Company_ID = '" & Company_ID(i) & "'"
  RS.Open SQL, Con, 1, 3
  Company_Name(i) = RS("Company_Name")
  RS.Close
 Next
%>
<br>
<form name="Frm" method="post" action="">
  <table border="1" cellspacing="0" cellpadding="5">
    <tr>
      <th colspan="4" bgcolor="#ECE9D8" align="center">Simulation Results - Retailer</th>
    </tr>
    <tr>
      <td bgcolor="#ECE9D8">Total Inventory:</td><td><%=GetCountNo(Con, "SELECT SUM(Detail) FROM
Simulation_Details WHERE Type = 'Inventory' AND Company_ID = '0'")%></td>
    </tr>
    <tr>
<%
OrderNo = 0
For i = 0 To Ubound(Company_ID)
 If Customer_Control(i) = 2 Then 'Push
  OrderNo = OrderNo + GetCountNo(Con, "SELECT Count(Detail) FROM Simulation_Details WHERE Type = 'PR' AND
Company_ID = '" & Company_ID(i) & "'")
 Else
  OrderNo = OrderNo + GetCountNo(Con, "SELECT Count(Detail) FROM Simulation_Details WHERE (Type = 'Delivery' OR
Type = 'Backorder') AND Company_ID = '" & Company_ID(i) & "'")
 End If
Next
OrderAmt = GetCountNo(Con, "SELECT SUM(Detail) FROM Simulation_Details WHERE (Type = 'Delivery' OR Type =
'Backorder')")
BackorderNo = GetCountNo(Con, "SELECT Count(*) FROM Simulation_Details WHERE Type = 'Backorder'")
BackorderAmt = GetCountNo(Con, "SELECT SUM(Detail) FROM Simulation_Details WHERE Type = 'Backorder'")
If OrderNo > 0 Then OrderNo = BackorderNo / OrderNo * 100
If OrderAmt > 0 Then OrderAmt = 100 - BackorderAmt / OrderAmt * 100
%>
      <td bgcolor="#ECE9D8">Overall Fill Rate:</td><td><%=Round(OrderAmt, 2)%>%</td>
      <td bgcolor="#ECE9D8">Overall Stockout Rate:</td><td><%=Round(OrderNo, 2)%>%</td>
    </tr>
    <tr>
      <td bgcolor="#ECE9D8">Total Delivery No (From Supplier):</td><td><%=GetCountNo(Con, "SELECT COUNT(Detail)
FROM Simulation_Details WHERE Type = 'PR' AND Company_ID = '0'")%></td>
```

```
    <td bgcolor="#ECE9D8">Total Delivery Amt (From Supplier):</td><td><%=GetCountNo(Con, "SELECT SUM(Detail)
FROM Simulation_Details WHERE Type = 'PR' AND Company_ID = '0'")%></td>
  </tr>
  <tr>
    <td bgcolor="#ECE9D8">Total Delivery No (To Customer):</td><td><%=GetCountNo(Con, "SELECT COUNT(*)
FROM Simulation_Details WHERE Type = 'Delivery' AND Company_ID <> '0'")%></td>
    <td bgcolor="#ECE9D8">Total Delivery Qty (To Customer):</td><td><%=GetCountNo(Con, "SELECT SUM(Detail)
FROM Simulation_Details WHERE Type = 'Delivery' AND Company_ID <> '0'")%></td>
  </tr>
<%For i = 0 To Ubound(Company_ID)%>
  <tr>
    <th colspan="4" bgcolor="#ECE9D8" align="center">Simulation Results - <%=Company_Name(i)%></th>
  </tr>
  <tr>
<%If Customer_Control(i) = 2 Then%>
    <td bgcolor="#ECE9D8">Strategy:</td><td>Push</td>
<%Else%>
    <td bgcolor="#ECE9D8">Strategy:</td><td>VMI</td>
<%End If%>
    <td bgcolor="#ECE9D8">Total Inventory:</td><td><%=GetCountNo(Con, "SELECT SUM(Detail) FROM
Simulation_Details WHERE Type = 'Inventory' AND Company_ID = '" & Company_ID(i) & "'")%></td>
  </tr>
  <tr>
<%If Customer_Control(i) = 2 Then%>
<%OrderNo = GetCountNo(Con, "SELECT COUNT(*) FROM Simulation_Details WHERE Type = 'PR' AND Company_ID =
'" & Company_ID(i) & "'")%>
    <td bgcolor="#ECE9D8">Total Order No:</td><td><%=OrderNo%></td>
<%OrderAmt = GetCountNo(Con, "SELECT SUM(Detail) FROM Simulation_Details WHERE Type = 'PR' AND
Company_ID = '" & Company_ID(i) & "'")%>
    <td bgcolor="#ECE9D8">Total Order Amt:</td><td><%=OrderAmt%></td>
<%Else%>
<%OrderNo = GetCountNo(Con, "SELECT COUNT(*) FROM Simulation_Details WHERE Type = 'Delivery' AND
Company_ID = '" & Company_ID(i) & "'")%>
    <td bgcolor="#ECE9D8">Total Delivery No:</td><td><%=OrderNo%></td>
    <td bgcolor="#ECE9D8">Total Delivery Amt:</td><td><%=GetCountNo(Con, "SELECT SUM(Detail) FROM
Simulation_Details WHERE Type = 'Delivery' AND Company_ID = '" & Company_ID(i) & "'")%></td>
<%OrderFreq = GetCountNo(Con, "SELECT [Order_Frequency] FROM VMI WHERE Product_ID = '" & Product_ID & "'
AND Company_ID = '" & Company_ID(i) & "'")%>
<%If OrderNo > OrderFreq Then%>
  </tr>
  <tr>
    <td bgcolor="#ECE9D8">Remarks:</td><td colspan="3">The Order Frequency of agreement is smaller than the current
Order No.</td>
<%End If%>
<%OrderAmt = GetCountNo(Con, "SELECT SUM(Detail) FROM Simulation_Details WHERE (Type = 'Delivery' OR Type =
'Backorder') AND Company_ID = '" & Company_ID(i) & "'")%>
<%End If%>
  </tr>
  <tr>
    <td bgcolor="#ECE9D8">Total Demand:</td><td><%=GetCountNo(Con, "SELECT SUM(Detail) FROM
Simulation_Details WHERE (Type = 'Demand' And Request_Date <= #" & Curr_Date & "#) AND Company_ID = '" &
Company_ID(i) & "'")%></td>
  </tr>
  <tr>
    <td bgcolor="#ECE9D8">Total Stock out No:</td><td><%=GetCountNo(Con, "SELECT Count(*) FROM
Simulation_Details WHERE Type = 'Backorder' AND Company_ID = '" & Company_ID(i) & "'")%></td>
<%BackorderAmt = GetCountNo(Con, "SELECT SUM(Detail) FROM Simulation_Details WHERE Type = 'Backorder' AND
Company_ID = '" & Company_ID(i) & "'")%>
    <td bgcolor="#ECE9D8">Total Backorder Amt:</td><td><%=BackorderAmt%></td>
  </tr>
  <tr>
<%
If OrderAmt = 0 Then
 FR = 0
Else
 FR = Round(100 - BackorderAmt / OrderAmt * 100)
End If
%>
    <td bgcolor="#ECE9D8">Fill Rate:</td><td><%=FR%>%</td>
<%Next%>
  </tr>
  <tr align="center">
    <td colspan="4" bgcolor="#ECE9D8">
      <input type="button" value="Initial A New Simulation" onClick="FrmSend('Initial_Sim')">
      <input type="button" value="Retain this Simulation" onClick="FrmSend('Retain')">
```

```
        <input type="hidden" name="Sim_Status" value="Initial_Sim">
      </td>
    </tr>
  </table>
</form>

<script language="vbscript">
Sub FrmSend(InputVal)
 Frm.Sim_Status.Value = InputVal
 Frm.Submit
End Sub
</script>

<%
 Dim Inventory()
 Redim Inventory(Ubound(Company_ID) + 1, Curr_Date - Start_Date)
 For i = 0 To Ubound(Company_ID)
  SQL = "SELECT * FROM [Simulation_Details] WHERE [Type] = 'Inventory' AND Company_ID = '" & Company_ID(i) &
"' ORDER BY [Request_Date]"
  RS.Open SQL, Con, 1, 3
  Do Until RS.EOF
   Inventory(i, RS("Request_Date") - Start_Date) = RS("Detail")
   RS.MoveNext
  Loop
  RS.Close
 Next
 SQL = "SELECT * FROM [Simulation_Details] WHERE [Type] = 'Inventory' AND Company_ID = '0' ORDER BY
[Request_Date]"
 RS.Open SQL, Con, 1, 3
 Do Until RS.EOF
  Inventory(Ubound(Company_ID) + 1, RS("Request_Date") - Start_Date) = RS("Detail")
  RS.MoveNext
 Loop
 RS.Close
%>

<%
Dim XAxis(), Series(), PicLink
Redim XAxis(Curr_Date - Start_Date), Series(Ubound(Company_Name) + 1)
For i = 0 To Ubound(XAxis)
 XAxis(i) = ConvertDate(Start_Date + i)
Next
PicLink = "Excel.gif"
For i = 0 To Ubound(Series) - 1
 Series(i) = Company_Name(i)
Next
Series(Ubound(Series)) = "Retailer"
'Call GenExcelChart(XAxis, Series, Inventory, "Inventory", "Time", "Qty", PicLink)
%>
<br>
<img src="<%=PicLink%>">
<br>
  <table border="1" cellspacing="0" cellpadding="5">
    <tr>
      <th colspan="<%=Ubound(Company_ID)+3%>" bgcolor="#ECE9D8" align="center">Inventory</th>
    </tr>
<%
 Response.Write("<tr><th>Date</th>")
 For i = 0 To Ubound(Company_ID)
  Response.Write("<th>" & Company_Name(i) & "</th>")
 Next
 Response.Write("<th>Retailer</th></tr>")
 For j = 0 To Ubound(Inventory, 2)
  Response.Write("<tr>")
  Response.Write("<td>" & ConvertDate(Start_Date + j) & "</td>")
  For i = 0 To Ubound(Inventory, 1)
   Response.Write("<td>" & Inventory(i, j) & "</td>")
  Next
  Response.Write("</tr>")
 Next
%>
</table>
<%
 Con.Close
 Set RS = Nothing
```

```
 Set Con = Nothing
End Sub

Function GetCountNo(Con, SQL)
 Set RS = CreateObject("ADODB.Recordset")
 RS.Open SQL, Con, 1, 3
 If IsNull(RS(0)) Then GetCountNo = 0 Else GetCountNo = RS(0)
 RS.Close
 Set RS = Nothing
End Function
%>
```

Top.inc

```
<table><tr><td nowrap>
<span style="color: #333333;font-size: 18px;font-style: italic;font-weight: bold;"><img src="images/top.jpg" width="165"
height="110" /> Knowledge-based Simulation Platform for Selection of Supply Chain Strategy </span>
</td></tr></table>
```

Upload.asp

```
<SCRIPT RUNAT=SERVER LANGUAGE=VBSCRIPT>

dim upfile_5xSoft_Stream
Class upload_5xSoft

dim Form,File,Version

Private Sub Class_Initialize
dim iStart,iFileNameStart,iFileNameEnd,iEnd,vbEnter,iFormStart,iFormEnd,theFile
dim strDiv,mFormName,mFormValue,mFileName,mFileSize,mFilePath,iDivLen,mStr
Version=""
if Request.TotalBytes<1 then Exit Sub
set Form=CreateObject("Scripting.Dictionary")
set File=CreateObject("Scripting.Dictionary")
set upfile_5xSoft_Stream=CreateObject("Adodb.Stream")
upfile_5xSoft_Stream.mode=3
upfile_5xSoft_Stream.type=1
upfile_5xSoft_Stream.open
upfile_5xSoft_Stream.write Request.BinaryRead(Request.TotalBytes)

vbEnter=Chr(13)&Chr(10)
iDivLen=inString(1,vbEnter)+1
strDiv=subString(1,iDivLen)
iFormStart=iDivLen
iFormEnd=inString(iformStart,strDiv)-1
while iFormStart < iFormEnd
iStart=inString(iFormStart,"name=""")
iEnd=inString(iStart+6,"""")
mFormName=subString(iStart+6,iEnd-iStart-6)
iFileNameStart=inString(iEnd+1,"filename=""")
if iFileNameStart>0 and iFileNameStart<iFormEnd then
iFileNameEnd=inString(iFileNameStart+10,"""")
mFileName=subString(iFileNameStart+10,iFileNameEnd-iFileNameStart-10)
iStart=inString(iFileNameEnd+1,vbEnter&vbEnter)
iEnd=inString(iStart+4,vbEnter&strDiv)
if iEnd>iStart then
mFileSize=iEnd-iStart-4
else
mFileSize=0
end if
set theFile=new FileInfo
theFile.FileName=getFileName(mFileName)
theFile.FilePath=getFilePath(mFileName)
theFile.FileSize=mFileSize
theFile.FileStart=iStart+4
theFile.FormName=FormName
file.add mFormName,theFile
else
iStart=inString(iEnd+1,vbEnter&vbEnter)
iEnd=inString(iStart+4,vbEnter&strDiv)
```

```
if iEnd>iStart then
mFormValue=subString(iStart+4,iEnd-iStart-4)
else
mFormValue=""
end if
form.Add mFormName,mFormValue
end if

iFormStart=iformEnd+iDivLen
iFormEnd=inString(iformStart,strDiv)-1
wend
End Sub

Private Function subString(theStart,theLen)
dim i,c,stemp
upfile_5xSoft_Stream.Position=theStart-1
stemp=""
for i=1 to theLen
if upfile_5xSoft_Stream.EOS then Exit for
c=ascB(upfile_5xSoft_Stream.Read(1))
If c > 127 Then
if upfile_5xSoft_Stream.EOS then Exit for
stemp=stemp&Chr(AscW(ChrB(AscB(upfile_5xSoft_Stream.Read(1)))&ChrB(c)))
i=i+1
else
stemp=stemp&Chr(c)
End If
Next
subString=stemp
End function

Private Function inString(theStart,varStr)
dim i,j,bt,theLen,str
InString=0
Str=toByte(varStr)
theLen=LenB(Str)
for i=theStart to upfile_5xSoft_Stream.Size-theLen
if i>upfile_5xSoft_Stream.size then exit Function
upfile_5xSoft_Stream.Position=i-1
if AscB(upfile_5xSoft_Stream.Read(1))=AscB(midB(Str,1)) then
InString=i
for j=2 to theLen
if upfile_5xSoft_Stream.EOS then
inString=0
Exit for
end if
if AscB(upfile_5xSoft_Stream.Read(1))<>AscB(MidB(Str,j,1)) then
InString=0
Exit For
end if
next
if InString<>0 then Exit Function
end if
next
End Function

Private Sub Class_Terminate
form.RemoveAll
file.RemoveAll
set form=nothing
set file=nothing
upfile_5xSoft_Stream.close
set upfile_5xSoft_Stream=nothing
End Sub
Private function GetFilePath(FullPath)
If FullPath <> "" Then
GetFilePath = left(FullPath,InStrRev(FullPath, "\"))
Else
GetFilePath = ""
End If
End function

Private function GetFileName(FullPath)
If FullPath <> "" Then
GetFileName = mid(FullPath,InStrRev(FullPath, "\")+1)
```

```
Else
GetFileName = ""
End If
End function

Private function toByte(Str)
dim i,iCode,c,iLow,iHigh
toByte=""
For i=1 To Len(Str)
c=mid(Str,i,1)
iCode =Asc(c)
If iCode<0 Then iCode = iCode + 65535
If iCode>255 Then
iLow = Left(Hex(Asc(c)),2)
iHigh =Right(Hex(Asc(c)),2)
toByte = toByte & chrB("&H"&iLow) & chrB("&H"&iHigh)
Else
toByte = toByte & chrB(AscB(c))
End If
Next
End function
End Class

Class FileInfo
dim FormName,FileName,FilePath,FileSize,FileStart
Private Sub Class_Initialize
FileName = ""
FilePath = ""
FileSize = 0
FileStart= 0
FormName = ""
End Sub

Public function SaveAs(FullPath)
dim dr,ErrorChar,i
SaveAs=1
if trim(fullpath)="" or FileSize=0 or FileStart=0 or FileName="" then exit function
if FileStart=0 or right(fullpath,1)="/" then exit function
set dr=CreateObject("Adodb.Stream")
dr.Mode=3
dr.Type=1
dr.Open
upfile_5xSoft_Stream.position=FileStart-1
upfile_5xSoft_Stream.copyto dr,FileSize
dr.SaveToFile FullPath,2
dr.Close
set dr=nothing
SaveAs=0
end function
End Class
</SCRIPT>
```

Upload_Process.asp

```
<!--#include file="upload.asp"-->
<%
Set upLoad = New upload_5xSoft
Set file = upLoad.file("ImgLink")
If file.FileSize > 0 Then
 file.SaveAs Server.mappath("Spectrum\" & file.FileName)
End If
Pic_Link = file.FileName
Set file = Nothing
%>
<script language="vbscript">
 Msgbox "File Uploaded"
 window.opener.Frm.Submit
 window.close()
</script>
```