Computers and Chemical Engineering 26 (2002) 1755–1769

Computers & Chemical Engineering

www.elsevier.com/locate/compchemeng

Agent-based supply chain management—1: framework

Nirupam Julka [1], Rajagopalan Srinivasan *, I. Karimi

Department of Chemical and Environmental Engineering, National University of Singapore, 4 Engineering Drive, Singapore 117576, Singapore

Received 26 July 2001; received in revised form 12 July 2002; accepted 12 July 2002

Abstract

In the face of highly competitive markets and constant pressure to reduce lead times, enterprises today consider supply chain management to be the key area where improvements can significantly impact the bottom line. More enterprises now consider the entire supply chain structure while taking business decisions. They try to identify and manage all critical relationships both upstream and downstream in their supply chains. Some impediments to this are that the necessary information usually resides across a multitude of resources, is ever changing, and is present in multiple formats. Most supply chain decision support systems (DSSs) are specific to an enterprise and its supply chain, and cannot be easily modified to assist other similar enterprises and industries. In this two-part paper, we propose a unified framework for modeling, monitoring and management of supply chains. The first part of the paper describes the framework while the second part illustrates its application to a refinery supply chain. The framework integrates the various elements of the supply chain such as enterprises, their production processes, the associated business data and knowledge and represents them in a unified, intelligent and object-oriented fashion. Supply chain elements are classified as entities, flows and relationships. Software agents are used to emulate the entities i.e. various enterprises and their internal departments. Flows—material and information—are modeled as objects. The framework helps to analyze the business policies with respect to different situations arising in the supply chain. We illustrate the framework by means of two case studies. A DSS for petrochemical cluster management is described together with a prototype DSS for crude procurement in a refinery.
© 2002 Elsevier Science Ltd. All rights reserved.

Keywords: Supply chain modeling; Software agents; Grafcets

Nomenclature

3PL	third party logistics
ADE	agent development environment
DSS	decision support system
EC	extra clustorial
*i*COP	intelligent cluster optimizer
KPI	key performance index
PRISMS	petroleum refinery supply chain modeler and simulator
RFQ	request-for-quote
RRFQ	reply-for-request-for-quote
SCM	supply chain management
SCN	supply chain network
SFC	sequential function charts

* Corresponding author. Tel.: +65-687-48041; fax: +65-6779-91936
E-mail address: chergs@nus.edu.sg (R. Srinivasan).
[1] Present address: Singapore Institute of Manufacturing Technology, Singapore.

0098-1354/02/$ - see front matter © 2002 Elsevier Science Ltd. All rights reserved.
PII: S0098-1354(02)00150-3

1. Introduction

The management of relationships across a supply chain to capture the synergy of intra- and inter-company business processes is referred to as supply chain management (SCM) (Lambert & Cooper, 2000). Historically, companies have focused only on their resources, constraints, and policies to make decisions and reduce costs. With intense competition and reducing profit margins, this approach is no longer sufficient. They need to consider the interactions with their suppliers and customers and incorporate them into their decision-making process. They also need to reformulate their business policies to enable them to incorporate the information regarding their supply chain into their decisions. Most companies are now paying close attention to SCM. This is largely due to the success stories of major corporations around the world saving billions of dollars in inventory and logistics costs by efficient management of their supply chains. SCM has risen to

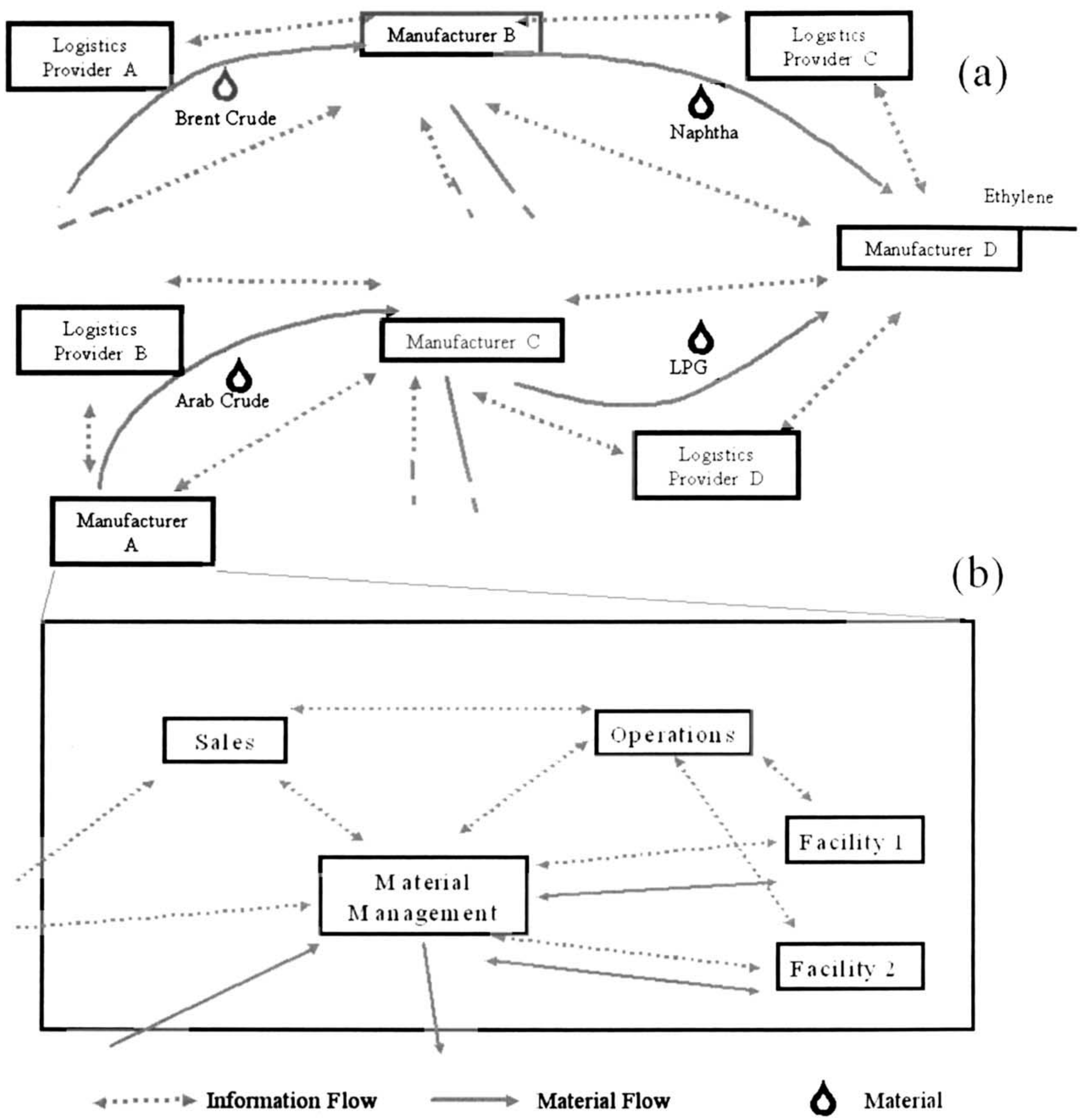

Fig. 1. (a) A chemical supply chain and (b) the internal departments of a typical manufacturer.

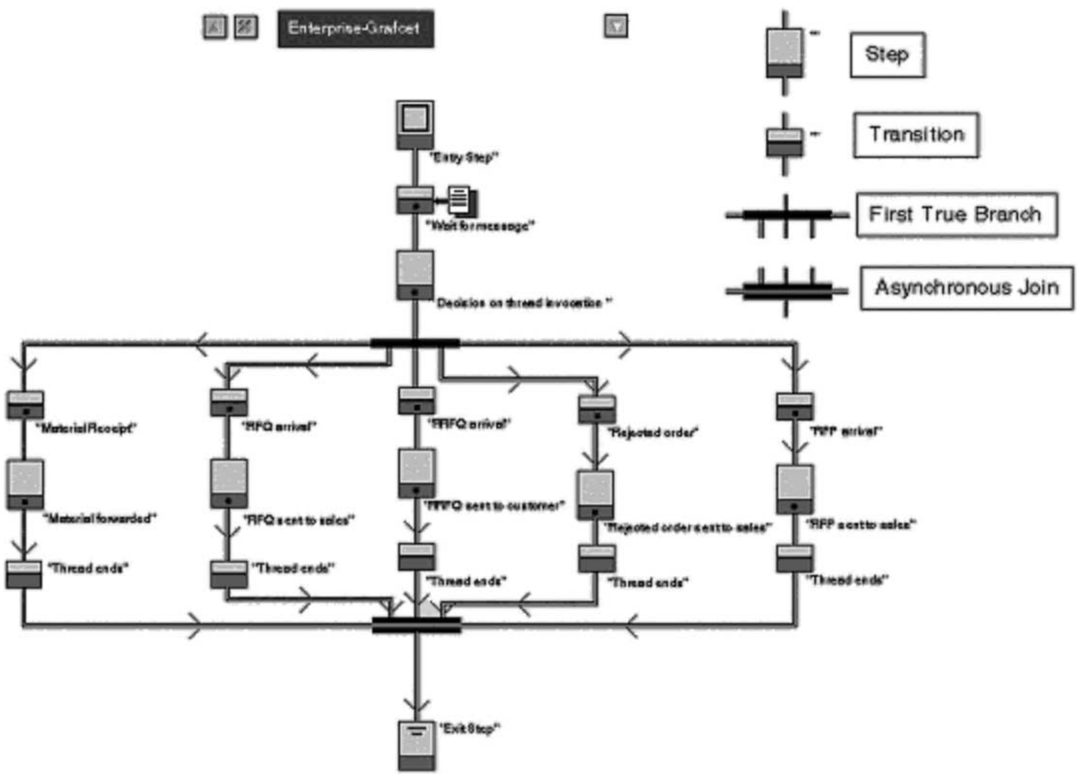

Fig. 2. Grafcet of an enterprise agent.

be a key strategic area that has direct impact on the success of any enterprise in today's highly competitive business environment.

There are several challenges in effective supply chain decision-making. The first challenge is that the information across all the departments and enterprises is distributed, dynamic, and disparate in nature. Secondly, in a present-day enterprise, decision centers reside in different departments. For instance, consider the refinery supply chain shown in Fig. 1. The enterprises and their internal departments take a number of decisions regarding information and material flows across the supply chain. These decisions are usually based on the department's own constraints, and the upper management handles the conflicts. Most often, the decisions are taken independent of the interdependency of various factors contributing to the overall business process of the enterprise. The decisions are optimized locally within the departments but do not assure a global optimum for the enterprise. Of course, decision support tools exist for local decision-making, e.g. planning and scheduling systems, inventory management systems, market trading optimization systems, etc. A mere electronic integration of these tools would not solve the problem and there is a need for a unified approach for modeling and analysis of supply chains, which explicitly captures the interactions among enterprises and within the departments of an enterprise. Such an approach would enable integrated supply chain decision-making.

This paper addresses this critical need by presenting an agent-based framework for supply chain analysis. Software agents exhibit collaboration, intelligence, and mobility, and thus are ideal for modeling and analysis of supply chains. We propose an agent-based framework that can emulate a supply chain at different levels of granularity—cluster, inter-enterprise, and inter-department. We begin with a discussion of software agents and their application to different chemical engineering problems, as well as SCM related problems in Section 2. Previous work related to SCM is reviewed in Section 3. Then, we present our agent-based framework in Section 4 and discuss its features in Section 5. In Section 6, we illustrate a decision support system (DSS) for the management of a petrochemical cluster. In the second part of this paper, we describe in detail another application for managing the crude oil procurement process in a refinery.

2. Software agents

Software agents can be defined in different ways depending on the way they are implemented and the tasks they perform. Wooldridge and Jennings (1995) suggest that any computer system (software or hardware) should have the following properties to be termed as an agent:

(a) Autonomy. It should have some control over its actions and should work without human intervention.
(b) Social ability. It should be able to communicate with other agents and/or with human operators.
(c) Reactivity. It should be able to react to changes in its environment.
(d) Pro-activeness. It should also be able to take initiative based on pre-specified goals.

The above-mentioned properties are generic for an agent. An agent may exhibit more of one property than another based on its architecture and embedded intelligence.

Toolkits are commonly used to develop multi-agent systems. In our work, we used Agent Development Environment (ADE), which is built on G2, Gensym's expert system shell. ADE has the following main components: agent, message, activity, host and environment. These components are explained in detail below.

2.1. Agent

An ADE agent is an autonomous, multi-threaded object, which communicates with other agents through messages. Every agent in ADE has a unique name, which serves as its address. This enables an agent to communicate with other agents irrespective of their location across a computer network. ADE provides a facility similar to "Yellow pages" to find the addresses of agents with some particular pre-defined property. Agents can send messages to other agents based on these properties. The task of an agent is described in the form of one or more activity classes (explained below), which is embedded inside the agent. At any particular time, an agent may perform multiple tasks, each of which is represented as an instance of a specific activity.

2.2. Message

The base level message class provided by ADE is called AdeMessage. All agents communicate with each other by sending objects of AdeMessage class or its subclasses. Information is embedded inside these objects in the form of attributes.

2.3. Activity

An activity defines a specific behavior of an agent. An agent may perform multiple activities of the same or different types with multiple threads active in an activity. Once an activity is initiated, messages can be sent specifically to it. An agent handler defines the destination activity for each message. Thus, activities are initiated by the receipt of a particular message class, defined in the message handler of the agent. AdeGrafcet is used to define the activity. Alternatively, a program can also define the activity. AdeGrafcet is based on Grafcet or Sequential Function Charts (SFC), a graphical language that has been accepted as an industrial standard (IEC 848 and IEC 1131-3) for PLC-level sequential logic control (David & Alla, 1992). The Grafcet chart is made of nodes and links (see Fig. 2). Nodes represent states of a system. The main nodes are transitions and steps and the main links are branches and joins. An AdeGrafcet step may contain small programs, unlike standard Grafcets, where the operations in the steps are Boolean in nature. A transition marks the change of the system from one state to another. It represents an event or condition, which is necessary to change the state of the system. Links connect two nodes in a Grafcet. Examples of links include First-True Branch and Asynchronous Join.

2.4. Host

There is a host for every multi-agent application. Each agent, when created, is assigned to a particular host. The most important task of the host is to provide message delivery service. All messages exchanged between agents are routed through the host. In this, the host serves the function of the post office.

2.5. Environment

An environment is a group of agents. The agents may be registered to different hosts and can change hosts but they cannot change their environment. Agents within an environment can be disallowed from communicating with agents external to their environment.

The above architecture permits agents to perform parallel tasks just as a human would do. Addition of more activities or threads to a particular activity will also add new functionalities to an agent. Further, the same agent class may be used inside different applications. Tasks performed at each step may be as simple as querying a database or as complex as collaborating with other agents to jointly solve an optimization problem.

Agents have been used to accomplish a variety of tasks. They have been used to assist process and equipment design, perform enterprise integration studies, and address the decision-making processes in a business. But a question often asked is what makes a task suitable for the use of multi-agent systems? In other words, what should be the characteristics of a problem to be addressed using multi-agent systems? Aylett, Brazier, Jennings, Luck, Nwana, and Priest (1997) justify the application of multi-agent systems to a problem that is inherently distributive in nature and requires the use of AI. However, they also mention that multi-agent systems are not required merely to produce modularity, extra speed, reliability, flexibility or reusability. We discuss multi-agent applications in chemical engineering as well as those in the field of SCM.

2.6. Chemical engineering applications

Most chemical engineering problems addressed by multi-agent systems are ones in which coordination of multiple entities is required. One such activity is the design process. Han, Douglas, and Stephanopoulos (1995) use an agent-based approach to develop a computer-aided design support system. The entire design process is decomposed into tasks and an agent is assigned to each task. Central agents as well as a human operator do the coordination between these agents to assist in the design task. Maguire, Scott, Paterson, and Struthers (1995) propose an agent-based modeling environment for process design. They introduce "autonomous cognitive entities" which follow four simple steps to take any decision. These steps are assess, decide, act, and review (ADAR approach). They use this modeling environment to model a flash process in Maguire, Struthers, Scott, and Paterson, (1998). Yang and Yuan (1999) present an agent-based framework for process plant operations where agents model tasks in the plant operations, e.g. monitoring, fault diagnosis, etc. Eo, Chang, Shin, and Yoon (2000) propose an agent-based framework for the diagnosis of chemical processes. Agents take care of a set of process units and communicate observations with each other. They use a knowledge base pertaining to the unit to take decisions.

McGreavy, Wang, Lu, Zhang, and Yang (1996) propose an agent architecture for concurrent engineering. They define three main agents as process, control, and equipment design agents. Each of these agents has subagents, e.g. the process design agent has reaction, separation, energy integration, and utility subagents. The main agents exchange design data to perform iterations in the design and negotiate for improvement in the complete design process. The agents inherently use application programs for the design and simulation of various sub-problems. The results from these programs are then compiled and sent to the other agents for improvements.

Batres, Lu, and Naka (1997), Batres, Asprey, Fuchino, and Naka (1999) address a concurrent process-engineering problem where programs which perform different design tasks are distributed over a local network and reside in different computers. The agents deployed at different locations and using applications written in different languages communicate with each other using Knowledge Query and Manipulation Language (KQML). Struthers (1997) presents an application of multi-agent systems to the task of organizing and managing a Pressure Relief and Blow down study. The agents are designed based on the Advanced Decision Environment for Process Tasks (ADEPT). Each agent comprises five smaller modules. These modules represent and manage communication, service availability, service execution, external agent service information, and situation assessment. Garcia-Flores, Wang, and Goltz (2000) present an agent-based framework to represent information flow in process industry supply chains. The agents are based on the ADEPT agent architecture. The entities in the supply chain perform the tasks of either clients (placing orders) or servers (fulfilling orders). Gjerdrum, Shah, and Papageorgiou (2001) present a combination of numerical optimization with expert system techniques to address supply chain modeling and performance assessment. They present a multi-tier supply chain with customers, inbound and outbound logistics, and production facilities. Their work focuses on ordering and replenishment and does not consider cross-department decision support within an organization.

3. Previous work in supply chain analysis

Beamon (1998) provides a focused review of the literature on modeling multi-stage supply chains. Modeling approaches are classified as deterministic analytical (Cohen & Moon, 1990; Williams, 1983), stochastic analytical (Cohen & Lee, 1998; Lee & Billington, 1993; Svoronos & Zipkin, 1991) and simulation (Towill, 1991; Towill, Naim, & Wikner, 1992; Wikner, Towill, & Naim, 1991). Most of the works focus on specific problem areas of SCM, e.g. inventory management, distribution systems, demand forecasting, etc. These solutions are thus inadequate to consider the complete supply chain structure and all the related knowledge and information. Also, they do not utilize the Internet, which today is a very important channel for doing business and sharing business related information in a seamless manner. To address these shortcomings, in the last few years, multi-agent approaches to supply chain problems have emerged. In this section, we summarize the literature in supply chain optimization and management in process industry. We also discuss existing multi-agent systems for SCM.

3.1. Chemical supply chain optimization and management

Chemical industry supply chains are typically long and plant configurations are rigid. Also, chemical plants are normally their own customers. Thus, supply chain solutions from other industries are not directly applicable in the chemical industry.

Perea-Lopez, Grossman, Ydstie, and Tahmassebi (2001) present a framework for dynamically modeling decentralized supply chains to assist in evaluating different heuristic control laws in their ability to dampen demand amplification in supply chains. They use a polymer manufacturer's supply chain as their case study. They consider decisions as the control variables of the dynamic system and associate the optimization of the decision-making process to designing a control law for a dynamic system. The approach though novel in nature, is specific to the demand amplification problem. Another approach in managing chemical manufacturing supply chains under uncertainty and risk is presented by Applequist, Pekny, and Reklaitis (1999, 2000).

Papageorgiou, Rotstein, and Shah (2001) address the issue of strategic supply chain design for pharmaceutical companies. They develop a mixed-integer linear programming (MILP) model to select a product development and introduction strategy and a capacity planning and investment strategy. Similarly, Bok, Grossman, and Park (2000) have addressed a multi-site continuous flexible process network problem. The objective is to maximize the profit of a SCN. The MILP model provides the sales, production and inventory profiles, and the production shortfalls for a scenario. Mathematical models are capable of providing accurate results, but cannot handle the computational complexity of the entire SCM problem. These are again examples of decision-support tools for specific problems.

Voudouris (1996) addresses the fine chemical industry supply chain problem. He emphasizes improving the efficiency of the supply chain by improving the logistics of the downstream formulation and packaging plants. An MILP model is developed by introducing a number of binary variables to represent discrete decisions pertaining to the supply chain. A crude oil supply problem is addressed by Shah (1996). The scope of the problem is limited to the allocation of crudes to the jetties, tanks and crude distillation units (CDUs). A similar problem of distribution of crudes to distillation columns is addressed by Kim, Choi, Kim, and Lee (1999).

3.2. Supply chain management using multi-agent systems

SCM problems are both distributive in nature, and require extensive intelligent decision-making. Thus, in the last few years, multi-agent systems have been a preferred tool for solving supply chain problems. Goodwin, Keskinocak, Murthy, Wu, and Akkiraju (1999) present a framework for providing decision support for an online exchange. They use a multi-agent system to find matches of demand and supply on the exchange and provide the user with the best set of transactions. The user then chooses the best match based on his/her discretion. Sauter, Parunak, and Goic (1999) present an architecture called Agent Network for Task Scheduling (ANTS), inspired by insect colonies and humans. Agents represent elements in the supply chain and within a factory. Each firm in turn is viewed as a small supply chain, thus the interface between agents within a firm is similar to those between different firms. This helps in easy integration of the supply chain elements. Agents also represent the resources of the firm. Each agent has a committed capacity profile, which shows its time-varying load. In case a new task has to be scheduled, each agent bids for that task with a bid that is inversely proportional to the committed capacity of the agent. Thus, resources are allocated to tasks without resorting to rescheduling the entire supply chain.

Swaminathan, Smith, and Sadeh (1998) present a modeling and simulation framework for developing customized decision support tools for supply chain reengineering. Agents represent supply chain entities, e.g. customers, manufacturers, and transportation. These agents use different interaction protocols and help in simulation of material, information, and cash flows. These interaction protocols are in the form of messages of various classes. Message handlers are associated with each message class and consider the agent receiving the message, to decide upon the message-processing semantics. The agents use various control policies to manage inventory, procure components, and determine optimal transportation routes. The framework helps in the configuration of a simulation model by selection, instantiation, and composition of sets of components without the need for extensive programming expertise. These simulation models are then used to analyze different reengineering options for a company.

In the next section, we describe our framework for supply chain analysis. We discuss how our framework is particularly suited for the chemical industry SCNs and addresses some of the drawbacks of the existing approaches.

4. Framework for supply chain analysis

A supply chain is defined as a network of suppliers, factories, warehouses, distribution centers, and retailers through which raw materials are acquired, transformed, and delivered to customers (Fox, Barbuceanu, & Teigen, 2000). In order to make supply chain decisions, it is necessary to identify the entities and flows in a supply chain. This calls for first modeling them to understand the supply chain and classifying them as critical and non-critical entities (Lambert & Cooper, 2000). The next step involves monitoring the critical elements. Finally, the identified and monitored elements need to be managed to improve the overall working of the supply chain. This would eventually optimize the supply chain.

We identify the elements in a supply chain, their features, and the challenges associated with SCM. We classify the elements in a supply chain as entities and flows. Entities include all manufacturers, logistics pro-

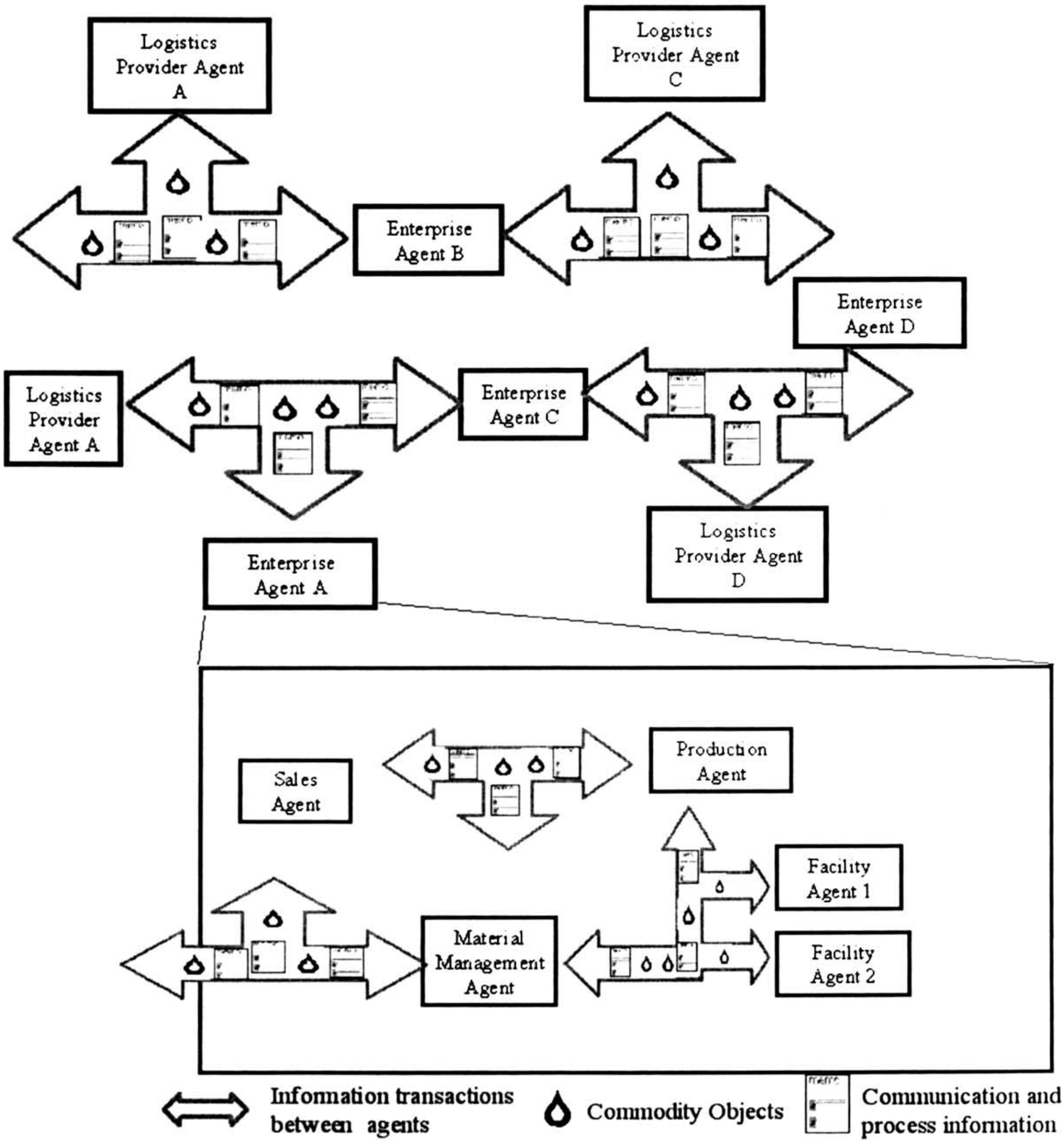

Fig. 3. Agent modeling of the supply chain entities.

viders, electronic exchanges and all their internal departments that participate in the business process. These entities are essentially the operators in the supply chain. Flows are of three types—material, information and finance, and these are the operands in the supply chain. These entities have three common features:

1) Dynamic: The supply chains are more flexible now. In today's business environment, there are no obligations for companies to be part of a supply chain for a certain time period and they may join or leave based on their own interest. This changes the structure and flows in the supply chain. Information in the supply chain e.g. prices, demands, technologies, etc. is also changing continuously.
2) Distributed: The elements are distributed across various geographical locations. The planning and operating systems used by an entity may also be geographically distributed e.g. there may be a dedicated inventory database residing at each warehouse of a manufacturer. The SCM related information might even reside as rules-of-thumb with the people responsible for performing the various tasks in the business process.

[2] We have currently not considered cash flows in our framework.

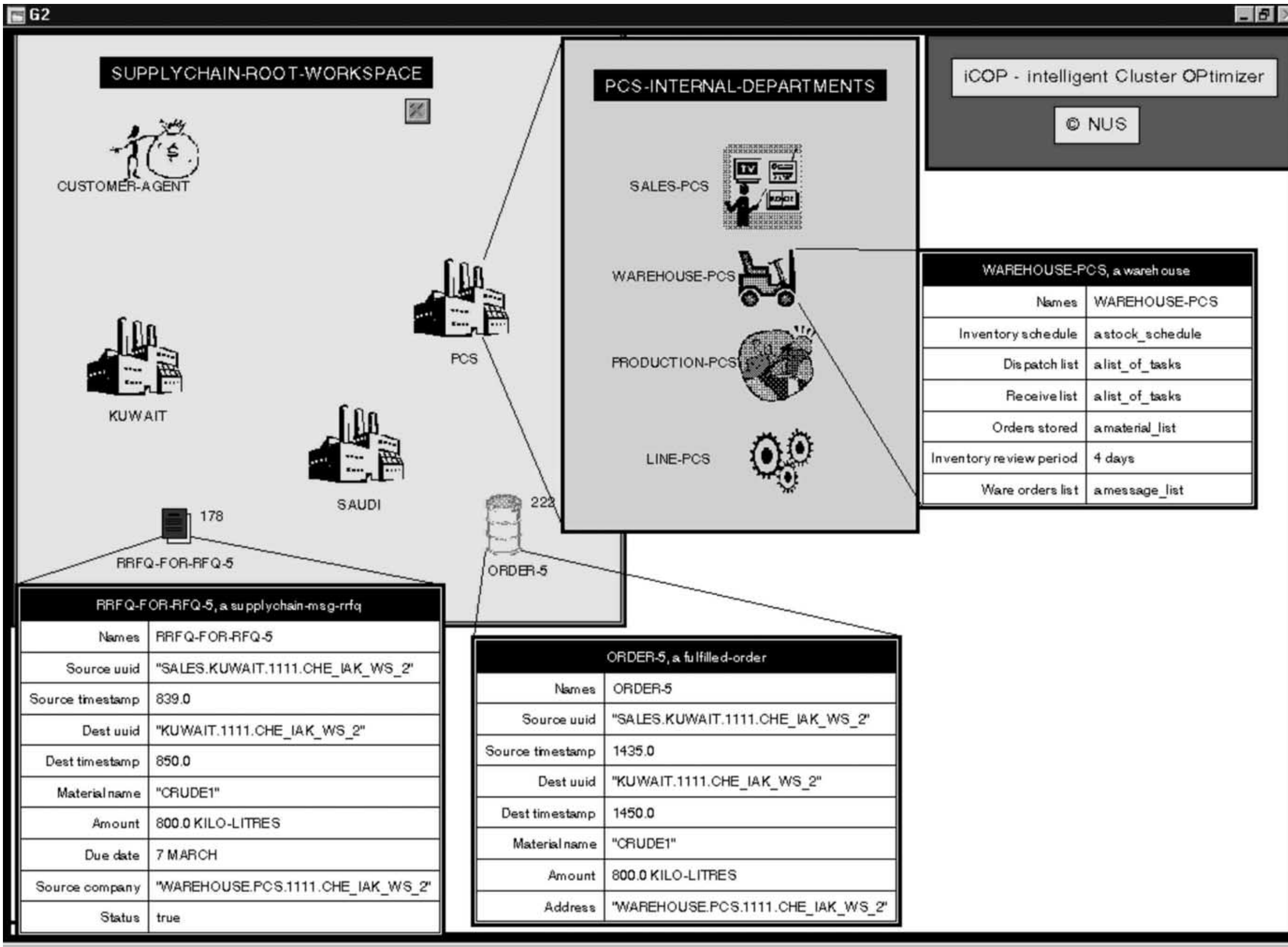

Fig. 4. Snapshot of *i*COP displaying the agent hierarchy and attributes of agents and objects.

3) Disparate: The entities in a supply chain use different systems built on different platforms for planning and management of their business. Information pertaining to the various elements is also disparate in form. For example, tracking of chemical shipment could be through emails, faxes, telephone calls or online reports.

Any attempt to model these elements and flows as a mathematical formulation to yield optimized results would be cumbersome. Furthermore, this task would be specific to an enterprise and modification to suit the needs of other enterprises or authorities taking supply chain related decisions would not be straightforward. Given this reality, there is a clear need for a new paradigm for DSSs, which is capable of gathering the dispersed information required for the normal function of an enterprise and providing a structured way to make decisions.

The foundation of our framework is a unified agent-based modeling of the entire supply chain network (SCN) in an object-oriented fashion. This supply chain model provides an environment where all business processes can be emulated. Fig. 3 illustrates the modeling of the supply chain based on the framework. Materials and information are modeled as objects. Each supply chain entity such as manufacturers and third party logistics (3PLs) are modeled as agents. The internal departments of these entities are modeled as sub-agents. The material and information flows are emulated by the exchange of the objects between these agents.[2]

The framework has two important elements. The first is the object modeling of supply chain flows such as material and information. This modeling involves two classes of objects, ‘commodity’ and ‘message’. These objects are exchanged between agents to simulate the material and information flow in the SCN. The details of the material and information are stored as attributes of the respective object. Fig. 4 shows the attributes of an instance of a fulfilled order, a commodity object. The details of the commodity object include name, amount of material, sender, receiver, time when it was dispatched, time when it was received, and the present location. Fig. 4 also shows an instance of a reply-to-request-for-quote (RRFQ). This message object has the details of name of the material, amount of material, status of order, source agent, destination agent and time of dispatch and arrival of message.

The second important element of the framework is the agent modeling. We use three classes of agents: emulation agent, query agent and project agent. The three classes are explained in detail below.

4.1. Emulation agents

Emulation agents model the supply chain entities. These include the manufacturers, 3PLs, internal departments of an enterprise, etc. Each agent may have one or more sub-agents. For example, in Fig. 4, the PCS enterprise agent that models the manufacturer has sub-agents modeling the internal sales, warehouse, production, and line departments of the manufacturer. These internal departments have generic functions in all enterprises, which can be customized to the practices of an enterprise. Some of the agents and their roles are explained below.

4.1.1. Enterprise agent

In our framework, an enterprise is defined as a plant site producing a set of products from a set of raw materials by utilizing a set of technologies. Enterprise agent can have sub-agents that perform tasks internal to an enterprise. This agent handles all communication with the enterprise. On receiving messages from other enterprises, this agent decides the further course of action. For example, if a request-for-quote (RFQ) is received, it forwards the message to its sales agent. Similarly, if a message requesting inventory information is received, it creates an internal request for this information and sends it to the warehouse agent. The enterprise agent also holds information on its products and the technologies available for use by its line agents.

4.1.2. Sales agent

This agent models the sales department of an enterprise. It is responsible for processing of RFQs and subsequent orders received by the enterprise. Customer information resides only with the sales agent. It keeps track of market demand and prepares demand plans. For this, it takes into account long-term alliances with customers, earlier demand patterns, and demands for related products on the exchanges. It keeps a record of all orders received, fulfilled, or queued. In case of made-to-stock goods, the sales agent generates the order. Consider a situation where the enterprise agent receives a RFQ from a customer. It forwards the RFQ to its sales agent. The sales agent queries the warehouse and the production agents regarding the order and its specifications and if they can fulfill it. The warehouse and production agents bid for that order based on present stock and production line availability, respectively. The sales agent processes these bids and decides on the outcome of the RFQ e.g. to be supplied from stock, to be produced or cannot be fulfilled. The sales agent replies to its enterprise agent, who then forwards the message to the customer.

4.1.3. Warehouse agent

This agent models the material-handling department of an enterprise. It issues the required commodity to other agents and replenishes raw materials stock by procuring from other enterprises. Thus the warehouse agent also acts as the "sticky-end" attaching the enterprise to a network of suppliers. Our framework enables the incorporation of different inventory policies such as periodic review, stock level maintenance, etc. into the warehouse agent.

4.1.4. Production agent

This agent models the operations or production department of an enterprise. It has sub-agents, which model the various production lines or facilities of an enterprise. This agent awards received orders to the production lines. On receiving a RFQ, it calls for bids from the line agents representing the production facilities. Based on these bids it generates a RRFQ. The production agent also procures material from the warehouse and allocates them to the lines based on their needs.

4.1.5. Line agent

This agent models the physical setup converting raw materials into finished products. Each agent has a schedule of tasks to be performed. Whenever the production agent calls for bids, the line agents capable of making that product estimate the cost to produce the requested amount of product. This cost takes into account the availability of the line as well as the technology used. Each line may use a different technology to convert raw materials to finished products. The technologies available are defined under the enterprise agent's technology library. Different scheduling algorithms can be incorporated into these agents.

The enterprise agent with its sub-agents is one example of an emulation agent. There is another category of emulation agents that can be used to model non-manufacturing enterprises such as the Internet exchanges of the present e-commerce era. An agent belonging to this category is the exchange agent. This agent models a real life online exchange. Each posting on the exchange is modeled as an object and is an attribute of the exchange agent. The deletion of the object representing the posting reflects the removal of a posting from the online exchange. Multiple instances of the exchange agent represent different electronic exchanges. Apart from including information on the postings, the information associated with the online exchange is also encapsulated in the exchange agent. Other atypical businesses can also be similarly modeled as emulation agents. But our goal is not just modeling of the supply chain. We also wish to analyze the supply chain and perform studies to get an insight into the supply chain behavior to assist in design, management, and operation of the supply chain. In our framework, the query agent assists in analysis of the supply chain models.

4.2. Query Agents

These agents do not emulate any supply chain entity. They handle queries from the user and assist in supply chain analysis. Each enterprise has one or more query agent. There are also query agents for clusters of enterprises. The query agents are active only in the analysis mode (described in Section 5). For example, a user may wish to know the status of backorders across an industrial cluster comprising of a number of manufacturing enterprises. The cluster level query agent receives the request, processes it, and identifies the information needed to reply to this request. It then contacts the query agents of the enterprises from which it needs the data. The query agents of the enterprise access the data of the agents under that enterprise and reply to the cluster level query agent. In the above example, the enterprise level query agent contacts the sales agent and production agent of the enterprise and collects the data on backorders from them. This data is processed, a reply generated and sent to the cluster level query agent. The cluster level query agent compiles the data from all enterprise level query agents and displays it to the user. Sometimes studies are to be performed or certain problems are to be solved using the supply chain model, which require more diverse steps than that provided by the query agent. Such studies or problem solving methodologies can be modeled as project agents.

4.3. Project agents

The project agent models any new project related to the SCN. A project is defined as a study or problem to be solved in the SCN. A project agent performs the tasks needed to perform the study or solve the problem. The solution methodology includes querying various other agents present in the framework either directly or through the query agent, accessing databases to retrieve information, creating or deleting material objects to simulate production and running optimization modules for local decision making. The concept of a project agent provides a way of adding analysis algorithms to the system in an easy manner without requiring extensive modification to the system. An example of a study modeled as a project agent is the cumulative inventory report for a petrochemical cluster. A problem solved using a project agent is the identification of new plants for a petrochemical cluster. Both of these are discussed later in Section 6.

Having presented the design and elements of the framework, we now discuss the implementation of the framework and how it facilitates supply chain DSSs for the new e-commerce scenario.

5. Framework features

We first identify features needed in any supply chain DSS. We then discuss how the proposed framework enables DSSs to address problems at different levels. Finally, we present the different modes in which a DSS based on the proposed framework could work. These would jointly bring out the advantages and features of the framework.

5.1. Needs of supply chain decision support systems

Fox, Barbuceanu, and Teigen (2000), Bui and Lee (1999) have identified the following issues for development of DSS for SCM:

- Distribution of supply chain activities among agents
- Coordination among DSS components
- Responsiveness to the modeling environment
- Interface with the present modules e.g. MRP systems
- Maximum re-use of DSS components

With the new e-commerce scenario in mind (Julka, Srinivasan, Karimi, Viswanadham, and Behl, 2000) we propose the following features for the modern day supply chain DSSs:

1) Knowledge encapsulation: Knowledge is an invaluable resource to an organization and the DSS should be able to use all this knowledge. The knowledge includes information on the structure of various entities, their working, and their relations with other entities. It also includes the material and information flow details. The DSS should have the knowledge organized in a manner which aids addition, deletion, modification and easy access. It should facilitate handling very specific queries.We use an object-oriented approach to encapsulate the data and knowledge in a SCN. The working of the different entities is captured using Grafcets as explained in Section 2.
2) Intelligent inference: Intelligent inference is an important feature for efficient query handling and problem solving. Many intelligent inference engines are available for the development of DSSs. In our work, we have used G2, Gensym's intelligent expert system shell.

3) Connectivity: The knowledge bases used by the system have to be constantly updated with the latest information. An example of this could be the current offers for various chemicals at the chemical exchanges on the World Wide Web (WWW). The system should have the capability to interface with the WWW, Wide Area Networks (WANs), and available enterprise databases. Furthermore, it should be able to intelligently mine for information from these sources. Our framework facilitates connection to external sources of information by deploying agents for the task. These agents pick up information from the source, convert it into the required format, and feed it to the system for incorporation into the decision-making process.
4) Flexibility: The system should be able to provide support in a variety of manners. It should be able to address queries regarding policy decisions, suppliers, information on inventory levels, pending orders, etc. It should help the user visualize the entire network from the view of a single enterprise as well as from the view of a planning authority managing a large industrial cluster. It should not only allow queries of 'what is', 'how much' and 'when' but also regarding 'what if'.Apart from intelligent query handling, our framework allows the user to add problem solving functionalities to the system using project agents. It also permits performing simulation of the business processes based on user-defined parameters and analysis based on user-defined performance metrics.
5) Collaboration and scalability: The DSS should be able to collaborate with legacy systems. This is an essential feature as a large amount of knowledge is available in existing systems and rewriting them for a new DSS would require enormous amounts of time, effort and money, if not altogether infeasible. The system should be scalable and modular in nature. This is to avoid a large number of changes in case two different systems need to be merged as may occur in a situation where the same company has multiple sites operating at different geographical locations but has a central planning facility. Modularity should also assist incorporation of DSS components developed elsewhere into the present system.

Since our framework models each entity in a supply chain as an agent, merging of multiple DSS into one is possible by just placing all the agents in the same environment.

5.2. Levels of decision support

The framework helps in decision support of supply chain problems at three levels, namely—cluster, inter-enterprise, and intra-enterprise. The cluster level constitutes a number of enterprises in a business environment forming a SCN e.g. a petrochemical cluster. The inter-enterprise level deals with a particular company and its supply chain e.g. a computer manufacturer buying various components from its suppliers, assembling them and selling to its dealers. The intra-enterprise level views the supply chain activities within an enterprise e.g. the crude oil procurement process in a refinery. Each of the decision levels is explained in detail below.

5.2.1. Cluster-level

Carrie (2000) defines a cluster to be "a network of companies, their customers and suppliers of all the relevant factors, including materials and components, equipment, training, finance, and so on." Decisions at this level are usually taken by agencies responsible for industrial and economic development of geographical regions. For instance, an agency is responsible for managing an industrial cluster. Investors planning projects in the cluster approach the agency with proposals. The agency would evaluate each proposal based on the value-added to the cluster and the project's impact on space availability, synergies with existing plants, environmental and other constraints. For making the best decision, the agency uses information on the present product spectrum of the cluster, the demand and supply of each product and the imports and exports associated with the cluster. It then studies which projects fit well in the cluster and increase the synergy within the cluster. With our framework, the user can include enterprises on the fly and perform "what-if" studies regarding total value addition of materials in related supply chains, effect on the logistics and other support industries, changes in the environment, and associated issues. Another example for cluster-level analysis is an agency handling the logistics of a large industrial cluster. This agency needs information on various materials and modes of logistics used for their transportation, the total bulk volume handled by each mode of logistics, etc. A DSS based on our framework can provide the agency answers to these questions and thus help in taking better decisions.

5.2.2. Inter-enterprise level

The decisions at this level are related to a particular company and its supply chain. Choosing the right suppliers, estimating demands, setting distribution points, and developing new products, etc., are some issues that can be addressed at this level. Often the critical links in a company's supply chain may not be the immediately adjacent firms; instead the bottleneck may lie in the second or third tier of suppliers or customers. An example, from the semiconductor industry illustrates such a situation (Lambert & Cooper, 2000) where all tier-one suppliers purchase from the same tier-two supplier. Thus, in case of shortages, the bottlenecks are the tier-two suppliers. A revelation of such dependencies may help the manufacturer decide on collaborating with the tier-two suppliers in decisions related to production planning. With our framework, the user can get overviews of various consumer–supplier relationships and discover such dependencies.

5.2.3. Intra-enterprise level

The decisions at this level are mostly related to internal departments of an enterprise and their functions. The decision domain may lie completely within the enterprise or at the interfaces between enterprises. The decisions on inventory policies, scheduling methodology and prioritizing orders belong to this level. The agent architecture allows the modeling of a company and its business practices and policies. The model can then be used to study a specific problem using the project agent. A simulation of material and information flow in this model is performed. The user can view the results with the help of pre-defined metrics or by using project agents to perform studies and "what-if" analysis.

5.3. Modes of decision support

A DSS based on the framework can work in two different modes—simulation and analysis. Both these modes complement each other. The former is a platform to simulate supply chain operations while the latter provides an environment to address queries and perform "what-if" analysis. The modes are usually used one after the other.

5.3.1. Simulation

In this mode, based on the parameters entered by the user, the DSS recreates the scenario of how a supply chain system with those parameters would perform. The user is able to view different key performance indices (KPIs) like inventory profiles, order fulfillment, etc., but is not permitted to dynamically change any parameter values. Only emulation agents are active in this mode. They simulate the processes of the supply chain under study. The simulation mode can be run based on a time period or on the basis of occurrence of an event. An example of an event may be a backorder of 50% of the capacity of a firm. Once the simulation time is over or the event has occurred the system can be used in the analysis mode to perform various studies.

5.3.2. Analysis

In this mode, the DSS addresses user queries about the simulation performed and allows him to configure the system for another simulation based on the results of the previous run. It helps the user evaluate the efficacy of policies and strategies. The user configures the scenario by specifying parameters like inventory levels, technologies involved and entities to be considered. The mode allows the user to configure very specific queries. The assistant agents, that are active only in this mode, handle these queries. The assistant agents reply to these queries using different presentation methods. For example, a query regarding the production path to get from material A to material B at minimum cost shall be displayed in the form of a serial supply chain of existing facilities. Each facility detail can then be accessed. A query regarding the inventory status of a particular material over the entire supply chain would be displayed in the form of a spreadsheet.

6. A decision support system based on the framework

We have used our framework to address two problems related to chemical SCNs. The first pertains to decision support to manage an industrial cluster. A system, called intelligent Cluster Optimizer (*i*COP), has been customized to manage a petrochemical cluster. The second system, called Petroleum Refinery Integrated Supply chain Modeler and Simulator (PRISMS), has been developed to provide decision support of the crude oil procurement process in a refinery. In this paper, we describe *i*COP while PRISMS will be presented in Part 2.

Consider a petrochemical cluster similar to that in Rotterdam (The Netherlands), Houston (USA), Jubail (Saudi Arabia), or Jurong Island (Singapore). An agency responsible for the overall strategic management of the cluster would be required to make decisions related to:

- Selecting projects from a group of proposals
  - What are the effects of the new projects on the demand and supply patterns of materials in the cluster?
  - Will the new projects increase the overall efficiency of the cluster?
- Net value generation by the cluster and measures to increase it
  - What are the total imports and exports for the cluster?
  - What is the net value addition?
  - Enterprises with what product spectra will be able to contribute positively to this figure?
- Effect on support organizations and shared services of the cluster.

The agent-based framework described in Section 4 can be used to build a DSS for such an agency. *i*COP is an example of such a DSS. The various chemical manufacturers in the cluster are modeled as enterprise agents. Companies in the cluster may import from or export to enterprises outside the cluster. These external

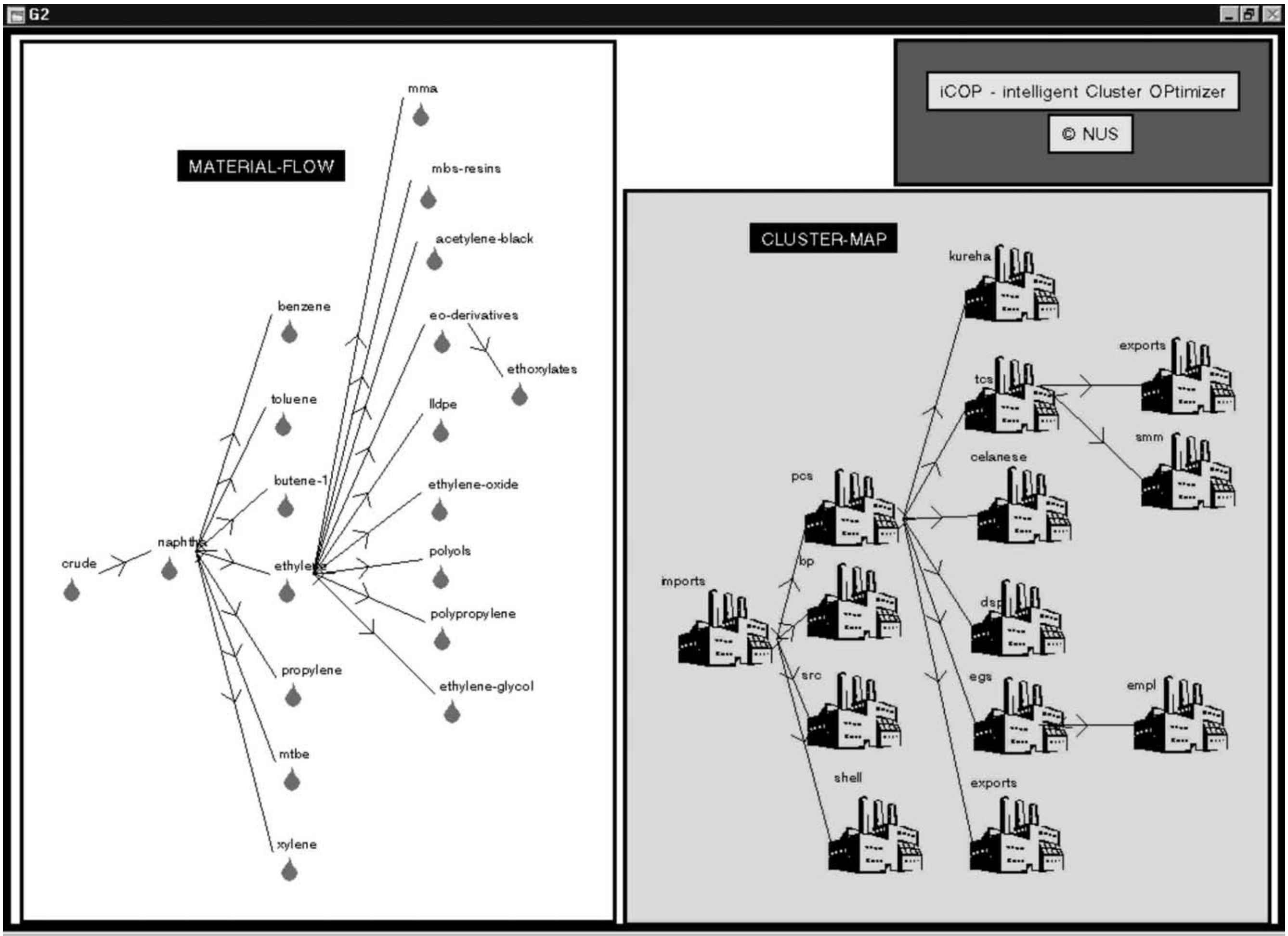

Fig. 5. Mapping of the SCN and the material chain network in a cluster using *i*COP.

enterprises are modeled as 'Extra Clustorial' (EC) agents. A scenario is configured by choosing the agents and customizing them. These scenarios are then analyzed using project agents. An example of a particular study performed by the project agent is the compilation of the information of the total inventory in the cluster. The various steps followed by the project agent in this case are as follows:

1) Contact the query agent of the cluster and request the complete list of all enterprises in the cluster.
2) Request the query agent of the each enterprise for information of their respective inventories.
3) Prepare a list of all materials and their quantities (stored and in process).
4) Take input from the user regarding the format of display (decreasing amount, total value, according to the enterprise).
5) Display the information of the inventory of materials over the cluster.

Another example of a problem solved by the project agent is the identification of potential projects for the cluster. The steps followed by the project agent are as follows:

1) Request the sub-agents of the cluster for all customer contracts with EC agents.
2) Prepare a list of all materials that are exported.
3) Find out available commercial technologies which use the materials in the list as raw materials.
4) Evaluate if technology specific constraints are satisfied.
5) Display the potential projects.

The data needed for *i*COP include product spectra of the companies and their capacities, present business links and supply chains, logistics information, and export–import information. All this information is incorporated into the DSS based on the guidelines given in the framework. The methodology of solving some of the above problems is provided by the agency. These methodologies are converted into decision algorithms and included as Grafcets. A prototype of the system has been put to use to address a few of the problems defined above. The user specifies the information related to a proposed project. *i*COP is able to provide the user with the changes in the export and import amounts of products over the whole cluster. It provides information regarding excess capacities at different enterprises of the network. It is able to provide the user with the changes in the supply pattern of the product produced by the project and corresponding demand present in the system. The above study can be repeated for different products and the effects can be compared. The DSS can provide a map of the transformation path followed by a material both with respect to the manufacturers and the material conversion. Fig. 5 shows a mapping of the SCN and the crude oil material conversion in a cluster as generated by *i*COP. By including specific decision-support algorithms and providing the required data, the system can be customized for any cluster management agency.

7. Conclusions

SCM has become the key strategic area that has direct impact over the success of any enterprise in today's highly competitive business environment. We have represented the supply chain and related problems through a unified, flexible, and scalable framework. The framework has two basic elements: object modeling of supply chain flows and agent modeling of supply chain entities. We use three classes of agents: emulation agents, query agents and project agents. This allows modeling the entire supply chain and the problem-solving methodologies required for decision-support. DSSs based on the framework can work in simulation and analysis modes. The user can configure supply chain scenarios and compare the effect of different policies and decisions using various KPIs. The framework enables decision support at cluster, intra-enterprise, and inter-enterprise levels and enables better decision support. We also illustrated the application of the framework using *i*COP, a DSS for the management of a petrochemical cluster. The same framework can be applied to provide decision support at the intra-enterprise level as illustrated in Part 2 using a refinery crude procurement process.

Acknowledgements

The authors would like to acknowledge financial support provided by The Logistics Institute—Asia Pacific, National University of Singapore.

References

Applequist, G. E., Pekny, J. F., & Reklaitis, G. V. (1999). Economic risk management for design and planning of chemical manufacturing supply chains. In *Proceedings of the fifth international conference on foundations of computer-aided process design*. c01

Applequist, G. E., Pekny, J. F. & Reklaitis, G. V. (2000). Risk and uncertainty in managing chemical manufacturing supply chains. *Computers and Chemical Engineering 24* (9), 2211.

Aylett, R., Brazier, F., Jennings, N., Luck, M., Nwana, H., & Priest, C. (1997). Agent systems and applications. In *Panel discussion report. Second UK workshop on foundations of multi-agent applications (FoMAS)*.

Batres, R., Asprey, S. P., Fuchino, T. & Naka, Y. (1999). A KQML multi-agent environment for concurrent process engineering. *Computers and Chemical Engineering 23* (Supplement), S653.

Batres, R., Lu, M. L. & Naka, Y. (1997). An agent-based environment for operational design. *Computers and Chemical Engineering 21* (Supplement), S71.

Beamon, B. M. (1998). Supply chain design and analysis: Models and methods. *International Journal of Production Economics 55* (3), 281.

Bok, J. K., Grossman, I. E. & Park, S. (2000). Supply chain optimization in continuous flexible process networks. *Industrial and Engineering Chemistry Research 39* (5), 1279.

Bui, T. & Lee, J. (1999). An agent-based framework for building decision support systems. *Decision Support Systems 25* (3), 225.

Carrie, A. S. (2000). From integrated enterprises to regional clusters: the changing basis of competition. *Computers in Industry 42* (2–3), 289.

Cohen, M. A. & Lee, H. L. (1998). Strategic analysis of integrated production-distribution systems: Models and methods. *Operations Research 36* (2), 216.

Cohen, M. A. & Moon, S. (1990). Impact of production scale economies, manufacturing complexity, and transportation costs on supply chain facility networks. *Journal of Manufacturing and Operations Management 3*, 269.

David, R. & Alla, H. (1992). *Petri Nets and Grafcet: tools for modeling discrete events systems*. Prentice Hall International.

Eo, S. Y., Chang, T. S., Shin, D. & Yoon, E. S. (2000). Cooperative problem solving in diagnostic agents for chemical processes. *Computers and Chemical Engineering 24* (2–7), 729.

Fox, M. S., Barbuceanu, M. & Teigen, R. (2000). Agent-oriented supply-chain management. *The International Journal of Flexible Manufacturing Systems 12* (2–3), 165.

Garcia-Flores, R., Wang, X. Z. & Goltz, G. E. (2000). Agent-based information flow for process industries' supply chain modeling. *Computers and Chemical Engineering 24*, 1135.

Gjerdrum, J., Shah, N. & Papageorgiou, L. G. (2001). A combined optimization and agent-based approach to supply chain modelling and performance assessment. *Production Planning and Control 12* (1), 81.

Goodwin, R., Keskinocak, P., Murthy, S., Wu, F., & Akkiraju, R. (1999). Intelligent decision support for the e-supply chain. In *Proceedings of artificial intelligence for electronic commerce—AAAI workshop 99* (p. 77).

Han, C., Douglas, J. M. & Stephanopoulos, G. (1995). Agent-based approach to a design support system for the synthesis of continuous chemical processes. *Computers and Chemical Engineering 19* (Supplement), S63.

Julka, N., Srinivasan, R., Karimi, I., Viswanadham, N., & Behl A. (2000) Enabling Framework for Decision Support Systems in the e-Commerce Era. Presented at AIChE Annual Meeting, 12–17 November, Los Angeles, CA, USA.

Kim, M., Choi, D. H., Kim, M. H., & Lee, I. B. (1999). Scheduling system for distribution of crudes to distillation columns in refinery processes. In *AIChE annual meeting*. Dallas, TX. 211i

Lambert, D. M. & Cooper, M. C. (2000). Issues in supply chain management. *Industrial Marketing Management 29* (1), 65.

Lee, H. L. & Billington, C. (1993). Material management in decentralised supply chains. *Operations Research 41* (5), 835.

Maguire, P. Z., Scott, D. M., Paterson, W. R. & Struthers, A. (1995). Development of an advanced modelling environment. *Computers and Chemical Engineering 19* (Supplement), S265.

Maguire, P. Z., Struthers, A., Scott, D. M. & Paterson, W. R. (1998). The use of agents both to represent and to implement process engineering models. *Computers in Chemical Engineering 22* (Supplement), S571.

McGreavy, C., Wang, X. Z., Lu, M. L., Zhang, M. & Yang, S. H. (1996). Objects, agents and workflow modeling for concurrent engineering process design. *Computers and Chemical Engineering 20* (Supplement), S1167.

Papageorgiou, L. G., Rotstein, G. E. & Shah, N. (2001). Strategic supply chain optimization for the pharmaceutical industries. *Industrial and Engineering Chemistry Research 40* (1), 275.

Perea-Lopez, E., Grossman, I. E., Ydstie, B. E., & Tahmassebi, T. (2001). Dynamic modeling and decentralized control of supply chains. *Industrial and Engineering Chemistry Research*, ASAP article (June 19, 2001).

Sauter, J. A., Parunak, H. V. D., & Goic, J. (1999). ANTS in the supply chain. In *Proceedings of workshop on agents for electronic commerce at agents '99*.

Shah, N. (1996). Mathematical programming techniques for crude oil scheduling. *Computer and Chemical Engineering 20* (Supplement), S1227.

Struthers, A. (1997). Use of intelligent software agents in an industrial pressure relief and blowdown design tool. *Computers and Chemical Engineering 21* (Supplement), S83.

Svoronos, A. & Zipkin, P. (1991). Evaluation of one-for-one replenishment policies for multiechelon inventory systems. *Management Science 37* (1), 68.

Swaminathan, J. M., Smith, S. F. & Sadeh, N. M. (1998). Modeling supply chain dynamics: a multiagent approach. *Decision Sciences 29* (3), 607.

Towill, D. R. (1991). Supply chain dynamics. *International Journal of Computer Integrated Manufacturing 4* (4), 197.

Towill, D. R., Naim, M. M. & Wikner, J. (1992). Industrial dynamics simulation models in the design of supply chains. *International Journal of Physical Distribution and Logistics Management 22* (5), 3.

Voudouris, V. T. (1996). Mathematical programming techniques to debottleneck the supply chain of fine chemical industries. *Computers and Chemical Engineering 20* (Supplement), S1269.

Wikner, J., Towill, D. R. & Naim, M. (1991). Smoothing supply chain dynamics. *International Journal of Production Economics 22* (3), 231.

Williams, J. F. (1983). A hybrid algorithm for simultaneous scheduling of production and distribution in multi-echelon structures. *Management Science 29* (1), 77.

Wooldridge, M. & Jennings, N. (1995). Intelligent agents: theory and practice. *Knowledge Engineering Review 10* (2), 115.

Yang, A., & Yuan, Y. (1999). An object/agent based system modeling framework for integrated process plant operations. In *AIChE annual meeting*. Dallas, TX. 211f.