Implementing the Internet Enabled Supply Chain through a Collaborative Agent System

Weiming Shen, Mihaela Ulieru & Douglas Norrie
Div. of Manufacturing Engineering
The University of Calgary
2500 University, Dr. NW, Calgary, AB, Canada
Tel: +1 403 220-5787
[wshen | ulieru | norrie]@enme.ucalgary.ca

Rob Kremer
Dept. of Computer Science
The University of Calgary
2500 University, Dr. NW, Calgary, AB, Canada
Tel: +1 403 220-5112
kremer@cpsc.ucalgary.ca

ABSTRACT

This paper presents some preliminary results of our ongoing research projects related to the development of a new Collaborative Agent System Architecture (CASA) and an Infrastructure for Collaborative Agent Systems (ICAS). Initially being proposed as a general architecture for Internet based collaborative agent systems, the proposed architecture is very suitable for managing the Internet enabled complex supply chain for a large manufacturing enterprise. The main features of the proposed architecture are described, some domain independent mechanisms and components are briefly introduced, and a case study of supply chain management is presented.

Keywords

Agents, Collaborative Agent Systems, Supply Chain, Internet.

1. INTRODUCTION

The Internet has evolved from an information space to a market space in the past few years. Electronic Commerce has become a very active research area within in a short period, with agent technology being recognized as a promising approach for its implementation (Guttman et al., 1997). Agent-based approaches have been quite widely applied for enterprise integration, manufacturing production planning, scheduling and process control, material handling within the past decade (Shen and Norrie, 1999), but few significant research results have been reported for applying agent technology for supply chain management, particularly for large manufacturing enterprises.

In this paper, we consider the supply chain of a manufacturing enterprise as a world-wide network of suppliers, factories, warehouses, distribution centers and retailers through which raw materials are acquired, transformed into products which are then delivered to customers. This type of supply chain network, in general, involves heterogeneous environments. Such a supply chain is much more complex than that for ordering, production and delivery of a simple commodity as can easily be imagined, not only for the volume and complexity of transactions, but also (and perhaps more importantly) because of its dynamic and heterogeneous manufacturing environments. The rapidly expanding Internet provides a promising networking medium, while agent technology provides a natural way to design and implement supply chain networks within such environments.

Fox et al (1993) may have been the first to propose organizing the supply chain as a network of cooperating, intelligent agents. A similar proposal has been made by Swaminathan et al (1996) using a multi-agent framework for modeling supply chain dynamics. In ISCM (Fox et al 1993), each agent performs one or more supply chain functions and coordinates its actions with other agents. In the supply chain library proposed by Swaminathan et al (1996), two categories of elements are distinguished: structural elements and control elements. Structural elements including production elements (retailers, distribution centers, plants, suppliers) and transportation elements are modeled as agents. Control elements (inventory, demand, supply, flow and information controls) are used to help in coordinating the flow of products in an efficient manner with the use of messages. MetaMorph II (Shen and Norrie, 1998) proposed to use a hybrid agent-based mediator-centric architecture to integrate partners, suppliers and customers dynamically with the main enterprise through their respective mediators within a supply chain network via the Internet and Intranets. In MetaMorph II, agents can be used to represent manufacturing resources (machines, tools etc) and parts, to encapsulate existing software systems, to function as system/subsystem coordinators (mediators), and to perform one or more supply chain functions. Recently, Brugali et al (1998) have proposed applying mobile agent technology to implement supply chain networks.

Based on eight years research work on applications of agent technology in intelligent manufacturing (Norrie and Shen, 1999), our group is now involved with industrial partners to develop and test a new Collaborative Agent System Architecture (CASA) in a complex supply chain network.

The rest of the paper is organized as follows: Section 2 briefly describes the collaborative agent system architecture; Section 3 presents some other domain independent mechanisms and components; Section 4 discusses the application of the proposed architecture to Internet enabled supply chain management; Section 5 depicts a scenario based on the proposed architecture and infrastructure; and finally Section 6 give some concluding remarks and perspectives.

2. THE COLLABORATIVE AGENT SYSTEM ARCHITECTURE

The development of the Collaborative Agent System Architecture (CASA) is a joint research project of Intelligent Manufacturing Systems Group (IMSG) in the Department of Mechanical and Manufacturing Engineering and Knowledge Systems Institute (KSI) in the Department of Computer Science at The University of Calgary. This section discusses the motivations and objectives of this project, introduces briefly the basic CASA, main components of this architecture and communication in CASA. A more detailed CASA description is being reported separately.

2.1 Motivations and Objectives

Eight years research experience related to applications of agent technology in developing agent-based intelligent manufacturing systems give us an insight into requirements and key issues of agent-based manufacturing systems. An overview of related projects at the University of Calgary with a summary of several interesting techniques and mechanisms developed during these projects and a discussion of key issues can be found in (Norrie and Shen, 1999).

Based on our previous research results, a new general purpose Collaborative Agent System Architecture (CASA) and an Infrastructure for Collaborative Agent Systems (ICAS) are proposed and being developed for the following reasons:

- a generic domain independent architecture is more reusable than a domain specific architecture;
- several interesting techniques and mechanisms have been developed and proved during our previous projects (Norrie and Shen, 1999), and they may become more interesting and useful after being developed into domain independent modules and mechanisms;
- emergent Internet technology and the Java programming language allow us to develop agent-based systems in a different way than previously.

Figure 1 shows an Infrastructure for Collaborative Agent Systems (ICAS). It also indicates our basic ideas and motivations to develop a generic collaborative agent system architecture.

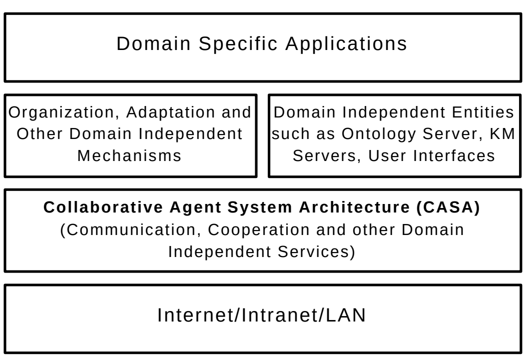

Figure 1. Infrastructure for Collaborative Agent Systems

The generic Collaborative Agent System Architecture (CASA) is composed of collaborative elements such as agents, local area coordinators (LACs), yellow pages and cooperation domain servers. The objective of the CASA work is to support collaborative software agents by providing easy-to-use domain independent communication and cooperation services over the Internet. These services include conversation messaging services, lookup and search services, and remote call services etc. The provision of these services should significantly reduce the complexity of developing systems of collaborative agents.

2.2 Cooperation Domain Servers

The cooperation domain server listens to a specified port for new connections. Other agents may send requests to this port to create a new cooperation domain, or join an active or inactive cooperation domain. The server may allow this request by creating a new cooperation domain server (in the case of a new cooperation domain or a join to an inactive cooperation domain) and replying to the agent with the port number that the cooperation domain server will use. In addition, the server offers data storage services to the cooperation domain – storing and retrieving named stream data on request.

The cooperation domain server is the receiver of all messages sent by the agents in the cooperation domain. Although the exact content of the messages may be in any language (whether known or unknown to the server), all messages are in an extended KQML format (Finin and Labrou, 1997). Thus, the cooperation domain server can determine the ultimate destination and some of the high level semantics of the message. The cooperation domain server takes care of forwarding the message transparently to its destination, whether it is an individual agent, a list of agents, all agents of a particular service type (role), or all agents in the cooperation domain. In addition, the cooperation domain server may record all transactions to enable history playback and "group undo" operations. A collaborative agent system may have one or more cooperation domain servers.

2.3 Yellow Page Agents

Yellow Page agents (also called Yellow Pages, or YPs) are responsible to accept messages for registering services and to record this information in a local database. Other agents may later query the yellow pages to determine what agents offer a specified service and may retrieve descriptions of those agents' locations.

2.4 Area and Local Area Coordinators

An area is a convenient quasi-physical division of the network that can be controlled by a local area coordinator (LAC). The area may be a single computer, some arbitrary division of a single computer, or a cluster of computers. A local area coordinator acts both as a representative of the area to the outside world, and a manager for the local agents within the area. Any agent may send a message to the LAC to request that the LAC ask one of its local agents to join a particular cooperation domain. The LAC will invoke the agent if necessary, and pass along the request to the agent.

The LAC also facilitates local agent's communication to the outside world by providing an interface service to the outside world. Instead of each agent having to keep a model of network locations of servers and yellow pages, the LAC provides a central cache of such data. Local agents may query the LAC to obtain yellow page services; the LAC may, in turn, pass these requests to one or more known yellow pages and broker the responses. This simplifies network modeling requirements for individual agents.

2.5 Communication in CASA

In CASA, agents do not communicate with one another in an ad-hoc, point-to-point manner. Instead, agents working together form cooperation domains. Each agent in a cooperation domain routes all its outgoing messages through the cooperation domain server, which can direct it to a specific agent (imitating point-to-point communication), to several agents (imitating multicast communication), or to all agents in the cooperation domain (imitating broadcast communication). All incoming messages are received from the cooperation domain server as well (the original sender's identity is contained in the message header). The benefit of all messages going through the cooperation domain server is that the server can then provide several services (described in Section 2.2) and agents can address roles and services, even if they don't know the identity of the agent or agents performing these roles or services. This type of communication is implemented by the Collaborative Agent Communication Language (CACL) that is developed by the same group. Initially, CACL allows for three simple actions: *request, reply* and *inform*. Messages are in an extended KQML format.

2.6 A Two-Agent Example

In this subsection, a two-agent example is used to illustrate the proposed agent system architecture and the supporting infrastructure. As shown in Figure 2, an agent A (e.g., a graph manager) needs the services of another agent which is capable of laying out graphs. The agent A finds such an agent (with the name B) which is capable of, and willing to, provide the needed services. Agent A and agent B engage in a conversation. Two areas are presumed: an area containing agent A (named: area A), and an area containing agent B (named: area B). The following entities are present in this scenario:

- Agent A: a graph manager;
- LAC A: the coordinator of all agents in area A;
- Yellow Page Server: the yellow page directory which is accessible from area A and contains information on area B;
- Cooperation Domain Server: a cooperation domain server available to areas A and B;
- Agent B: a graph layout application agent;
- LAC B: the coordinator of all agents in area B.

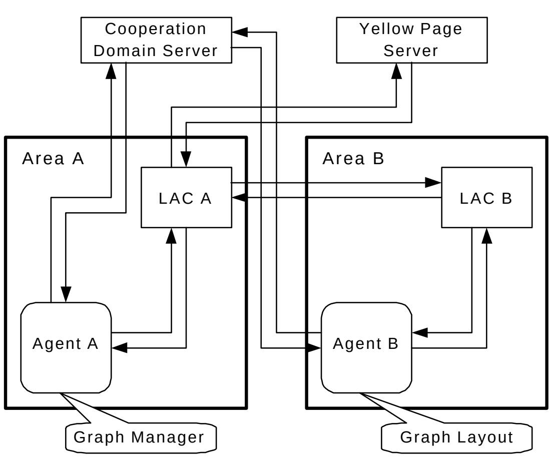

Figure 2. A Two-Agent Example

A simplified scenario is as follows:

- Agent A sends a request to the Cooperation Domain Server (CDS) to initial a cooperation domain (start a conversation).
- CDS responds that a cooperation domain has been initiated and that agent A is member of that cooperation domain and should on a particular port, CDS1. Data is received.
- Agent A sends a request to LAC A that it needs the location of an agent which is capable of laying out a graph.
- The LAC A interprets the request. It does not know of agents (residing in area A) which fulfill the request. LAC A sends a request to the Yellow Page Server (YP) to find the location of an agent capable of laying out a graph.
- LAC A receives results from YP. This is a list of candidate agents (and their local area coordinators).
- LAC A sends to the agent A the results it obtained from YP.
- The agent A chooses a particular agent from the list of candidate agents. Assume the selected agent has name B, and its local area coordinator has name LAC B.
- Agent A sends a request to its LAC A that it wishes that agent B (coordinated by LAC B) joins cooperation domain CDS1 in the capacity of a graph layouter.
- LAC A sends a request to LAC B stating that agent A wishes that agent B joins cooperation domain CDS1 in the capacity of a graph layouter.
- LAC B activates agent B (which may not be active), and sends it the request that it is supposed to join cooperation domain CDS1 with agent A in the capacity of graph layouter.
- Agent B informs LAC B that it is willing to collaborate with agent A on cooperation domain CDS1 in the capacity of graph layouter.
- LAC B informs LAC A that its request is honored: the requested agent B shall join the cooperation domain CDS1 in the capacity of graph layouter.
- LAC A informs agent A that the requested agent B is joining the cooperation domain CDS1 in the capacity of graph layouter.
- Agent B addressed the Cooperation Domain Server CDS1 and requests to join that conversation.
- Cooperation Domain Server CDS1 sends a message to agent B that agent B has now joined the cooperation domain CDS1.
- Cooperation Domain Server CDS1 sends a message to agent A that agent B has now joined the cooperation domain CDS1.

3. OTHER DOMAIN INDEPENDENT MECHANISMS AND COMPONENTS

The organization, adaptation and some other domain independent mechanisms (such as task decomposition, virtual clustering, agent cloning, etc.) as shown in Figure 1 have been separately proposed and developed during our previous research projects (Norrie and Shen, 1999). We are now working on defining and developing several domain independent components such as collaborative user interfaces, high-level collaboration agents, knowledge management (KM) servers (or agents), ontology server(s) and so on. All these components will become domain specific when implemented for a specific application and filled with domain specific information and knowledge. Details of these components are being reported separately, and are also out of scope of this paper. However, we will give a brief introduction of several components related to applications described in this paper.

Collaborative Interface Agents

We consider that (domain independent) collaborative interface agents may have the following characteristics: communicative, semi-autonomous, collaborative, reactive, pro-active, adaptive, self-aware, and mobile. However, the implementation of interface agents in a real application does not need all these features. The development of collaborative interface agents is one of current research projects in our group, and the results will be reported separately.

High-Level Collaboration Agents

Although several CASA components such as YPs and LACs also provide basic collaboration (cooperation) services, some complex large collaborative agent systems often need more high-level collaboration services which cannot be covered by YPs and LACs. In order to meet such requirements, the high-level collaboration agent is proposed as one of important components for ICAS. Such collaboration agents may be implemented by mediators as proposed and developed in our previous projects (Gaines et al., 1995; Maturana et al., 1998). In most cases, they are static, but they can also be implemented using dynamic mediators.

Knowledge Management Agents

Knowledge management is one of the most important issues in developing multi-agent systems. Typically, there are two approaches to knowledge management in multi-agent systems: knowledge is distributed among agents; some knowledge management agents are used to centralize knowledge management. Our approach to this issue is to combine above mentioned two approaches, i.e., in addition to distribute some knowledge among agents, several knowledge management agents are developed for specific problems. A knowledge management agent is usually associated with one or more databases and knowledge bases. The key issues to develop knowledge management agents are to develop efficient mechanisms for knowledge acquisition, representation, learning and reasoning. Note that some simple knowledge management agents may be implemented by Yellow Page agents in CASA.

Ontology Servers

As one of the important components for our proposed infrastructure, ontology server(s) will be developed using *Ontolingua* (Gruber, 1993). The ontology server structure and related mechanisms are initially domain independent. It becomes domain specific when it filled with domain specific ontologies for a specific application, e.g., supply chain management. Of course, it is not difficult to define the structure and mechanisms for an ontology server, but it is extremely difficult to develop and complete an efficient ontology server for an application domain.

4. BENEFITS OF IMPLEMENTING THE INTERNET ENABLED SUPPLY CHAIN WITH ICAS

Although the Infrastructure for Collaborative Agent Systems (ICAS) is proposed as a general approach for developing Internet enabled collaborative agent systems, most components and mechanisms proposed and developed under this infrastructure are very useful and suitable to develop Internet enabled supply chain networks. This section discusses the benefits of implementing the Internet Enabled supply chain networks with ICAS components.

- *Cooperation Domain Servers (CDSs)*: In the Internet enabled supply chain, one of the most important services is the customer-supplier or business-business negotiation. CASA's *Cooperation Domain Servers* provide a natural way and media for such negotiation. It becomes more effective when it works together with the Virtual Clustering mechanism described below.
- *Yellow Page Agents (YPs)*: Because of the dynamics and complexity of the Internet enabled supply chain of a manufacturing enterprise, *Yellow Page Agents* will play a very important role for providing looking-up and search services.
- *Local Area Coordinators (LACs)*: *LACs* can significantly facilitate the organization and management of the Internet enabled supply chain. They are especially useful and important for the Internet enabled supply chain of large international manufacturing enterprises.
- *Collaborative Interface Agents (CIAs)*: Different types of interfaces are needed for the Internet enabled supply chain: interfaces for customers to input orders; interfaces for marketing and operation managers, production managers, enterprise general managers, etc. *Collaborative Interface Agents* proposed and developed under ICAS can be used to develop these interfaces. Each type of interface may be composed of only several modules of the general CIA model with optional components.
- *High-level Collaboration Agents (HCAs):* This type of *collaboration agents* can be used to provide collaboration services which cannot be covered by the basic CASA collaboration (cooperation) services, e.g., regional marketing coordinators, regional operation coordinators, local or regional production coordination, etc.
- *Knowledge Management Agents (KMAs):* Managing a complex supply chain needs a lot of knowledge ranging from customer requirements, marketing, product modeling, and project management etc. Several *knowledge management agents* may be developed to facilitate the agent-based supply chain management.
- *Ontology Server(s):* For the Internet enabled supply chain of a manufacturing enterprise as we described at

the beginning of this paper, the ontology problem becomes more crucial because it is related to highly heterogeneous environments, e.g., multiple communication languages (e.g., CACL, KQML, FIPA's CAL, etc) and multiple knowledge/data/information interchange formats (e.g., KIF, EDI, etc) may be used. Thus, one or more *ontology server(s)* should be employed to provide a standard for the supply chain on the terms used in communication languages and knowledge related these terms' definitions, attributes, relationships and constraints.

– *Virtual Clustering*: When a customer submits an order through a collaborative interface agent as described above, one regional marketing and operation center of the manufacturing enterprise needs to negotiate with different factories and partners for raw material (or parts) supply, parts fabrication, parts transportation, and product assembly etc. A number of collaborative agents including collaborative interface agents for customers and managers, automatic software agents for cost calculation, production planning, conflict detection and resolution and so on are involved in this negotiation process. The *Virtual Clustering* mechanism provides an efficient way to form a virtual collaboration group to facilitate such negotiation. At the sub-system level, this mechanism has been proved to be very useful during production planning and scheduling.
– *Agent Cloning*: Similar to the Virtual Clustering mechanism described above, agent cloning mechanism will be very useful for some collaborative agents to be involved in several collaboration groups (clusters) simultaneously, which will also reduce network communication load significantly. While at the sub-system level, it has also been proved to be very efficient during agent-based manufacturing planning and scheduling.

5. A CASE STUDY

Taking the benefits described in the previous section into account, a real case study of implementing the Internet enabled supply chain using collaborative agents is underway in collaboration with industrial partners. Due to space limitations, this section only gives an extraction of this case study for showing the functions of above mentioned components in the agent-based supply chain. A more detailed description of the real case study is being reported separately.

5.1 Conceptual Description

In the context of this case study, Supply Chain Management (SCM) is a corporate-wide initiative that includes teams from Corporate, Business Units, and Marketing Regions. It is the implementation of a common set of processes, applications, and performance measurements for the corporation. Therefore, it is the management of all interactions involved in the initialization of an order to the delivery of that order's products to the customer. The main objectives of the SCM system are:

– Cut Costs;
– Shorten development time;
– Speed up product delivery;
– Create a collaborative culture to fully leverage Corporate's global resources;
– Guarantee common performance measurements across the Corporation;
– Enable the Corporate organizations to quickly adapt to global market dynamics.

The real case study is very complex, here we just take an extraction of the case which involves a customer, the corporate's

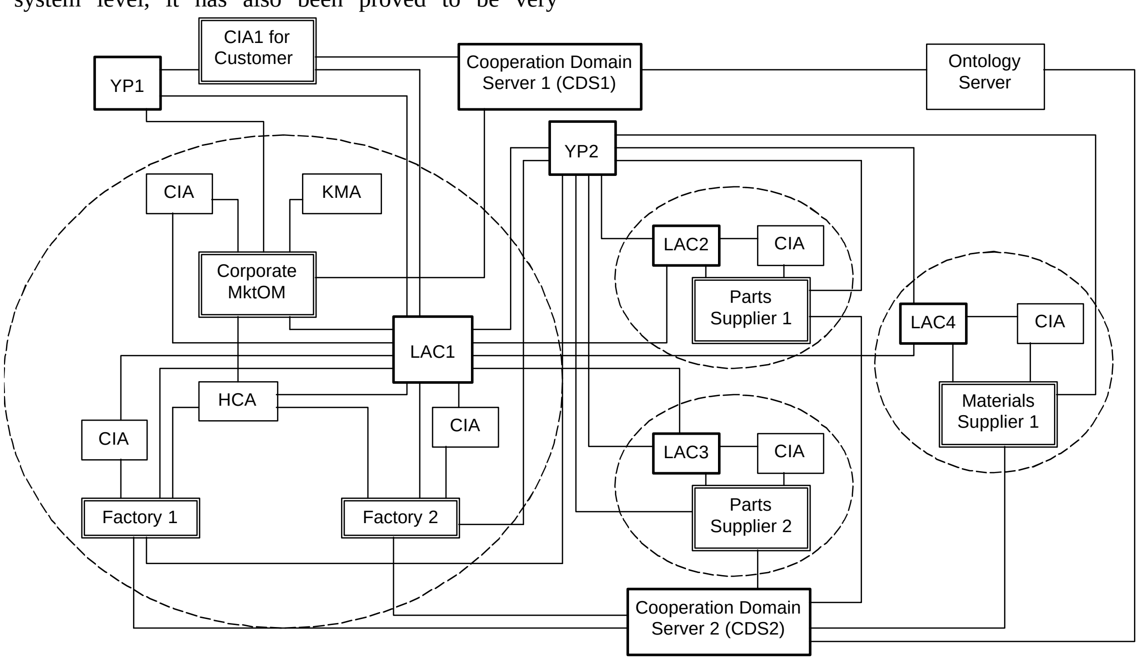

Figure 3. A Scenario for a Simplified Supply Chain

Marketing and Operations Management center (MktOM), 2 factories, 2 parts suppliers and 1 materials supplier (Figure 3). The customer places an order for buying 100 handsets (mobile phones) of type A and 50 handsets of type B. Factory 1 can produce handsets A, and needs parts from Parts Suppliers 1 & 2, and some other materials from Materials Supplier 1. Factory 2 can produce handsets B, and needs parts from Parts Supplier 2 and materials from Materials Supplier 1.

5.2 Scenario at a Glance

- The customer wants to order some handsets, and uses "CIA1 for Customers" to search a suitable marketing service;
- CIA1 finds out MktOM through YP1 and LAC1;
- The customer (through CIA1) negotiates with MktOM for product types, prices and delivery date etc through the Cooperation Domain Server 1 (CDS1), and finally places an order of 100 handsets A and 50 handsets B through the CIA1;
- MktOM receives the order and sends a confirmation to CIA1;
- MktOM requests KMA to complete task decomposition and find suitable factories for production;
- MktOM sends decomposed tasks to HCA which then assigns production tasks to related factories (Factory 1 and Factory 2);
- Factory 1 needs to contact Parts Suppliers 1 & 2 and Materials Supplier 1 (through YP2, LAC1, LAC2, LAC3 & LAC4);
- Factory 2 needs to contact Parts Supplier 2 and Materials Supplier 1 (through YP2, LAC1, LAC3 & LAC4);
- The negotiation, cooperation among factories and suppliers is facilitated by CDS2;
- Marketing managers and production managers can use related CIAs for monitoring and making decisions;
- All Cooperation Domain Servers are connected with the Ontology Server for translating messages from different agents into a common format so as to facilitate the communication, negotiation and cooperation.

5.3 ICAS Support to the Case Scenario

The ICAS infrastructure supports the situation in the case study in the following ways:

- Cooperation Domain Servers are used to facilitate the cooperation and negotiation between customers and the corporate marketing service (CDS1 in Figure 3), and between the factories and there parts and materials suppliers (so-called business-to-business commerce) (CDS2 in Figure 3).
- Collaborative Interface Agents are employed for order input by customers; displaying product orders, production scheduling and progress information for marketing, operation, and production managers; and allowing these managers to make decisions.
- The Ontology Server functions as a translator during the conversation among different agents through cooperation domain servers.
- YPs provide look-up and search services for customers to find production information and related marketing services, and for factories to find suitable suppliers for parts and materials.
- The HCA functions as a coordinator of Factory 1 and Factory 2 for coordinating production planning.
- The KMA provides support for decision-making on the part of Marketing and Operations Managers, particularly during the task decomposition process.
- In this case, the advantages of LACs are not very evident. But they become extremely useful for a complex and widely distributed supply chain.
- The Virtual Clustering mechanism is needed to form a group of agents for negotiation (e.g., Factory 1, Parts Suppliers 1 & 2 and Materials Supplier 1 are formed into a cluster for parts and materials supply negotiation).
- Other components and mechanisms are not found in this simplified case, but are needed and will be used in the real case study.

6. CONCLUSIONS AND PERSPECTIVES

The collaborative agent system architecture (CASA) and the infrastructure for collaborative agent systems (ICAS) were initially proposed as a general architecture associated with components and mechanisms for Internet-based multi-agent systems. All components (including agents, YPs, LACs, cooperation domain servers, collaborative interface agents, ontology servers, etc.) are domain independent. This paper shows the benefits of implementing the Internet enabled supply chain with ICAS. A case study is used to prove this feasibility. This paper is intended to illustrate the preliminary results of our ongoing project, which is underway in collaboration with industrial partners.

7. ACKNOWLEDGEMENTS

Thanks to all other Collaborative Agent Group members for their contribution to the development of the Collaborative Agent System Architecture, and detailed discussions on the case study. These members include Bob Brennan, Roberto A. Flores, Fuhua Lin, Sudong Shu, Matsumoto Tsutomu, Niek Wijngaards, Xiaokun Zhang, Bin Zho.

8. REFERENCES

[1] Brugali, D., Menga, G. and Galarraga, S. (1998). Inter-Company Supply Chains Integration via Mobile Agents. In Jacucci, G. (ed.), Globalization of Manufacturing in the Digital Communications Era of the 21st Century: Innovation, Agility and the Virtual Enterprise, Kluwer Academic Publisher, pp. 43-54.

[2] Finin, T. and Labrou, Y. (1997). KQML as an agent communication language. In Software Agents, J.M. Bradshaw, (ed.), MIT Press, Cambridge, pp. 291-316.

[3] Fox, M.S., Chionglo, J.F. and Barbuceanu, M. (1993) The Integrated Supply Chain Management System. Internal Report, Dept. of Industrial Engineering, Univ. of Toronto.

[4] Gaines, B.R., Norrie, D.H. and Lapsley, A.Z. (1995). Mediator: an Intelligent Information System Supporting the Virtual Manufacturing Enterprise. In Proceedings of 1995

IEEE International Conference on Systems, Man and Cybernetics, New York, IEEE, pp. 964-969.

[5] Gruber, T. (1993). A Translation Approach to Portable Ontology Specification. Knowledge Acquisition, **5**(2), 199-220.

[6] Guttman, R., Moukas, A. and Maes, P. (1998). Agent-Mediated Electronic Commerce: a Survey. Knowledge Engineering Review. Available at: http://ecommerce.media.mit.edu/

[7] Maturana F., Shen, W. and Norrie, D.H. (1998). MetaMorph: An Adaptive Agent-Based Architecture for Intelligent Manufacturing. International Journal of Production Research, (in press).

[8] Norrie, D.H. and Shen, W. (1999). Applications of Agent technology for Agent-Based Intelligent Manufacturing Systems. Submitted to IMS'99.

[9] Shen W. and Norrie, D.H. (1998). An Agent-Based Approach for Distributed Manufacturing and Supply Chain Management. In Jacucci, G. (ed.), Globalization of Manufacturing in the Digital Communications Era of the 21st Century: Innovation, Agility and the Virtual Enterprise, Kluwer Academic Publisher, pp. 579-590

[10] Shen, W. and Norrie, D.H. (1999). Agent-Based Systems for Intelligent Manufacturing: A State-of-the-Art Survey. Knowledge and Information Systems, an International Journal, to appear.

[11] Swaminathan, J.M., Smith, S.F., and Sadeh, N.M. (1996). A Multi-Agent Framework for Supply Chain Dynamics. In Proc. of NSF Research Planning Workshop on AI & Manufacturing, Albuquerque, NM.