Ontology-based Product Tracking System

Vikram N. Ketkar, Larry Whitman & Don Malzahn
Department of Industrial and Manufacturing Engineering
Wichita State University
Wichita, KS 67260

Abstract

Product tracking enables efficient management of inventory through better synchronization in the supply chain. This paper examines the use of intelligent software agents that utilize protocols, neutral languages and ontologies for product tracking. Software agents operating on behalf of individual enterprises and communicating using a common ontology provide the interoperability required to access current product information from multiple systems. This paper proposes an architecture for such a multi agent ontology based product tracking system. It also provides an outline of a product ontology that can be used in this system.

Keywords
Supply chain management, Product tracking, Intelligent software agents, Ontology

1.0 Introduction
Supply chain is a collaboration between different enterprises. The collaboration succeeds due to the coordination between the processes and operations of the partners of the chain. Product tracking is an important function that enables synchronization among the partners. Selective real time product status information can be shared among the members of the supply chain with the help of a network of intelligent software agents in which a separate software agent represents each member. The software agents are interoperable and they all refer to a common set of protocols, formats, and ontologies. To enable this interoperability, the software agents have to communicate in a common representative language called the Agent Communication Language (ACL). The Knowledge Interchange Format (KIF), the Knowledge Query and Manipulation Language (KQML), and an ontology are the components of ACL. A specific ontology has to be created to suit the particular requirements of a supply chain product tracking system. In this paper an architecture for a multi-agent ontology based product-tracking system and an outline of a product ontology is presented.

2.0 Literature review
2.1 Product Tracking
Product tracking involves the use of all means of information gathering about objects during the logistics processes. Typically, in advanced industrial practices, all objects including consignments, parcels, pallets, containers, and even modes of transport like trucks or vans are tagged with machine-readable identification labels like barcodes or GPS-devices. The status of objects is recorded at certain milestones, which could be the transshipment point, or a refueling stop during transportation, or could be the shipping and the receiving department at an enterprise. It could also be the transaction when an object passes from one department or process to another department in an enterprise. At such milestones the status information about the object like the part number, batch number, quantity, customer order number along with the current date and time can be recorded in a centralized database with the help of barcode scanners. This information is shared with all the concerned members of the supply chain and is used to control and synchronize the processes in the entire supply chain [1,2].

2.1.1 Supply Chain
"A supply chain is a web of autonomous enterprises collectively responsible for satisfying the customer, by creating an extended enterprise that conducts all phases of design, procurement, manufacturing, and distribution of products" [4]. In a supply chain, material is acquired, transformed, and finally delivered to the customer as an end product. The ability to optimize the tactical and operational decision-making to control this process, depends on the timely dissemination of information. Information sharing synchronizes activities and processes in the supply chain and helps to utilize resources optimally. Product tracking fulfills this important function of tracking real time product status information in a supply chain [3,4].

2.1.2 Benefits of Product tracking

Product tracking systems enable better planning and coordination by enhanced information sharing. The paperless, seamless, no-touch communication has very few process failures and can convey 100% accurate information to any point in the chain. The control loop as depicted by figure 1 can use the real time product status information from tracking systems for optimizing supply chain operations [5]. Product tracking acts like fulfillment checks for comparing the scheduled processes and functions with the actual processes and functions. The analysis of the discrepancies will provide a feedback for improving the ways of fulfillment checks. The analysis will also help to create methods and measures for improving the processes and functions. The real-time product status information can empower marketing functions with better decision-making capabilities. This information also proves to be a major differentiation strategy for many companies. It is highly valued by the internal customers and can be effectively used by management for controlling personnel and for optimizing business processes. The information on the location of objects in transit helps to co-ordinate trucks at transshipment locations. The trucks can be made available at the right time and place with the required trucking capacity [5]. Large logistics companies have been profitably using product-tracking systems for many years [1,6].

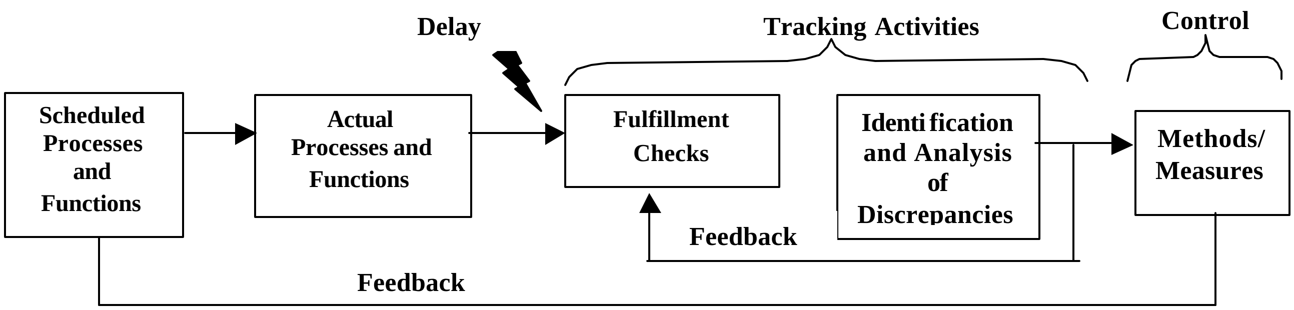

Figure 1. Control Loop

2.1.3 Software agents for product tracking

A supply chain can be defined by a collection of software agents in a virtual market that operate on behalf of their clients conducting business between enterprises for a duration of time as is mutually beneficial for all [3]. A software agent, operating in a virtual market, is an ideal entity to work on behalf of partners of a supply chain. A software agent can work with an agenda, negotiate with other agents, travel to other systems, and understand the content of information [3]. The next section of the paper discusses the basic concepts of intelligent software agents and their components. The use of such agents in a product tracking system is then examined.

2.2 Software agents

2.2.1 Concepts

"An agent is a software entity, which functions continuously and autonomously in a particular environment often inhabited by other agents and processes" [7]. Agents communicate and negotiate with their peers by exchanging messages and data in a common communication language [8,9]. The expressiveness of this language enables them to exchange data, logical information, individual commands, and scripts. The agents are designed to work in close coordination with their human clients [10].

2.2.2 Agent Communication Language

Software agents, working on behalf of different enterprises face a problem when interacting with other agents that use a different syntax and/or vocabulary. An agent might use a word or expression that can mean something entirely different than what another agent understands from the same word or expression. Common communication languages facilitate interoperability by decoupling implementation from the interface. Researchers at the DARPA (Defense Advanced Research Projects Agency) have defined the components of an Agent Communication Language (ACL) that can make an agent truly interoperable and intelligent to execute various tasks [11]. The ACL consists of three parts: A vocabulary or ontology, an inner language like Knowledge Interchange Format (KIF), and an outer language like Knowledge Query and Manipulation Language(KQML). "An ACL message can be described as an KQML expression in which the arguments are terms or sentences in KIF formed from words in the ACL vocabulary" [11]. These components of ACL are discussed below.

2.2.2.1 Knowledge Interchange Format (KIF)

"Knowledge Interchange Format (KIF) is a formal language for the interchange of knowledge among disparate computer programs" [12]. This neutral language was created by the Interlingua Working Group of DARPA's Knowledge Sharing effort in 1992. KIF can be used as a neutral language for exchanging information among humans and as an internal representation of knowledge within computer programs. Like any natural language KIF has both syntax and semantics. The KIF syntax defines the way in which the words and expressions are written in KIF. The declarative semantics logically explain meaning of expressions used in the representation. They establish a correlation between the terms and sentences of the language and the 'conceptualization of the world'. The semantics interpret the meaning of complex expressions with true or false conditions and by defining rules and axioms [12]. KIF is being used as a content language for many agent systems like Fujitsu's AGENTPRO 1 product [13].

2.2.2.2 Knowledge Query and Manipulation Language (KQML)

KQML complements representation languages like KIF. It is a message format and a message-handling protocol that supports knowledge sharing among agents. KQML is used as a language by application programs to interact with a network of intelligent agents and to share knowledge in support of cooperative problem solving [14]. The protocols decide and define how the agents will send and receive messages (Transport), determine the meaning of individual messages (Language), structure their conversations (Policy), and connect their systems with different interfaces (Architecture) [8]. KQML contains sub-expressions in other content languages and is independent of format making it autonomous and asynchronous. This makes KQML suitable for agent communication [15].

2.2.2.3 Ontology

Gruber defines an ontology as "an explicit specification of a conceptualization." [16]. An agent needs a conceptualization of the domain (ontology), if it has to ask queries, make statements and exchange information about any subject domain. A conceptualization has to express concepts, terms and quantities that will describe an object (physical or abstract) in the 'universe of discourse' in its entirety. This description has to be expressed in some language that is highly expressive like a natural language. That language also has to be accurate and unambiguous [17]. The ontology names and describes entities that may exist in a domain and specifies the relationships between them. It provides a vocabulary for representing and communicating knowledge about the domain [18,19]. Ontologies are like common technical dictionaries used for referencing transactions and communications that create a common understanding of terms and definitions. They also define the behavior of the entities by using formal axioms enforcing constraints. Ontologies are essential for the development and optimal utilization of intelligent agent systems as they enable the agents to interoperate in heterogeneous systems [11]. The construction of an ontology is a very difficult and time-consuming task. To simplify this task integrated tool systems, like Protégé-2000, Ontolingua and Chimaera, are available online which help to develop domain-specific ontologies [20,21].

2.2.2.3.1 The Need for an Ontology

The major difficulty for exchanging product information among different enterprises and their applications is that these applications sometimes associate different meanings to the same terms or have different terms for representing the same product information. For example, the batch quantity represented as 'Batch Quantity' in one application may be called as lot quantity in another. A delivery time could mean different times for different applications. Direct mapping between the two terms would undoubtedly cause a lot of confusion. A precisely defined product terminology, relevant for exchanging product status information, can enable accurate information integration in the extended enterprise. This combination of terminology and definitions will constitute a product ontology. For a software agent to make statements and ask queries about a subject domain, which for the purpose of product tracking is product status information, it will have to refer to a conceptualization of that domain. An product ontology would provide a conceptualization of the product status domain to software agent. And when represented in a neutral application format like XML, the ontology can also support interoperability by providing a common vocabulary with shared machine-interpretable semantics [17]. An ontology will also help to achieve consensus among a set of managers about the meaning of terms. Thus to create a product tracking system using software agents, a product ontology for representing generic product status information needs to be created. Software agents, operating in a multi-agent architecture, will use this ontology as a reference for tracking product status information and thus enabling coordination among the enterprises in a supply chain [17,18].

3. 0 The Product tracking system

3.1 Multi agent network

This proposed product tracking system will operate in a multi agent environment as depicted in figure 2. In this model every member of the supply chain has a database and applications that exchange data and information through a network of intelligent software agents. Each of these agents will have specific tasks to perform. There are three types of agents in this system: Mediator Agents, Database agents and Application agents. The specific roles of these agents are described below.

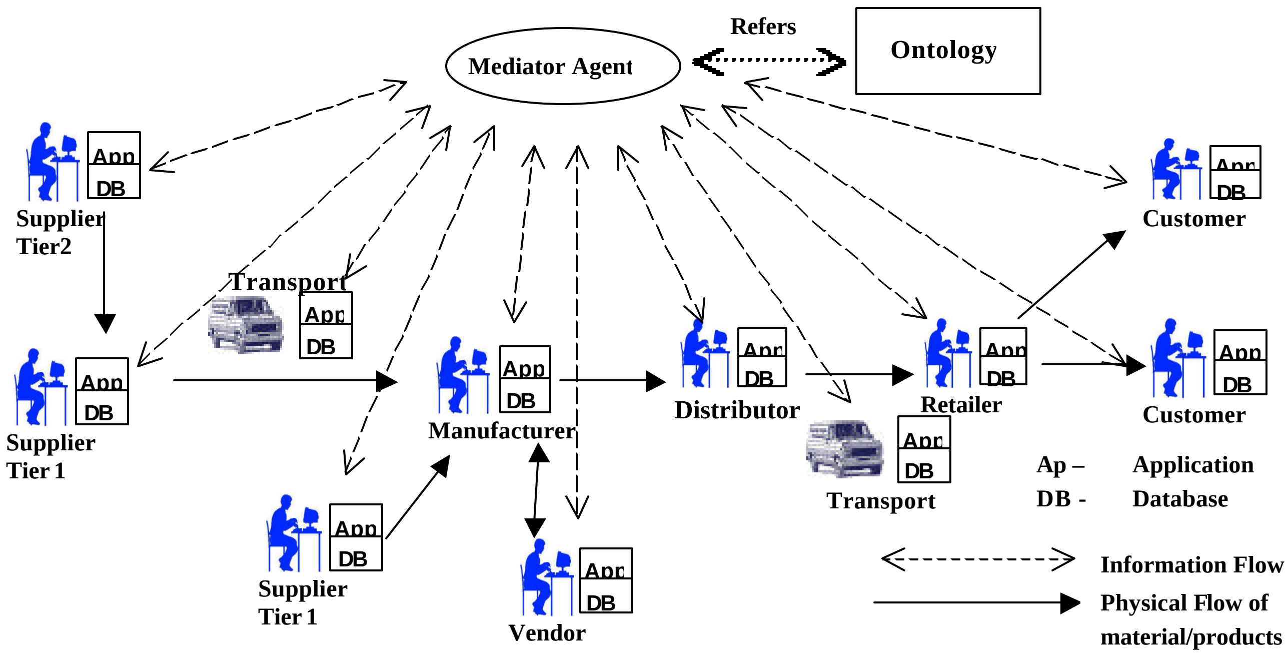

Figure 2. Multi Agent System Architecture

3.1.1 Mediator Agents (MA)

Mediator agent is the entity that binds together the databases and the applications used throughout the supply chain, enabling interoperability in the system. MA receives requests from client applications and supplies them with information and data as per their needs. MA refers to the product ontology and verifies information obtained from the application and database from across the supply chain. If two or more information sources are found in response to a request then the MA will interact with each source in parallel and integrate the information to be sent back as a response.

3.1.2 Application Agents (AA)

All the user applications in the product tracking system will have application agents (AA). The application agent receives requests for information from user applications in a particular format, converts them to ACL strings and passes them to MA. On receiving a response from the mediator the AA will translate and pass on the information to the user application in its own format. The information received is parsed by the application and data is extracted. The AA protects the application software from the location, format and the style of the information it seeks.

3.1.3 Database agents (DBA)

The DBA's are the agents wrapped around the user databases of the product tracking system. The main function of DBA's is to process conversions between ACL messages and SQL queries and respond to the requests of the MA's. SQL is the universal query language of database management systems like Oracle and Microsoft Access. The DBA's convert DB data into virtual knowledge that can be used for sending ACL messages that can advertise the capabilities of DBAs to MA's.

3.2 Product ontology

The ontology for the Product tracking system will be built according to the guidelines recommended by Natalya Noy and Deborah McGuinness from the Stanford University [22]. The iterative ontology building exercise will aim to learn and import components from ontologies hosted on Ontolingua and other servers. The domain of this ontology is the representation of product status information in a supply chain. The aim of the ontology is to be a part of a multi agent system that will assist in tracking a product as it travels in a supply chain. The scope and the domain of the proposed ontology can be better specified by the specific queries that the ontology will be capable of answering. The ontology will be able to answer specific questions like -

- A manufacturer's query: What is the total inventory of this part number in the supply chain?
- A manufacturer /distributor/ retailer/ end customer's query: When will my consignment reach me?
- A logistics company's query : At what time will be consignment be ready for me to ship?

The main hierarchy of classes and subclasses will be the bill of material for a particular product. In this hierarchy the product family is identified by a product number, a component number identifies the components that make up the product, and instances of these components or batches of components are identified by a product number or a batch number. The properties of these classes and subclasses will be the status information that is recorded at every milestone of the object's journey. These properties are called slots and they include details like the location of the tracked object, the quantity, invoice value, the shipping/ received date and time, and other shipping company details. The constraints on the values of these properties are called facets. The facets will define constraints on the property values like the value type, allowed values, and the number of values (cardinality).

4.0 Conclusion

Product tracking plays a very important role in supply chain optimization. A network of intelligent software agents can disseminate product status information most efficiently in a supply chain. A framework of a multi agent product tracking system is described and illustrated in the paper. Agents can be interoperable and intelligent when they share a common set of protocols (KQML), a common language (KIF) and a common ontology. The paper discussed components of the Agent Communication Language. An example ontology is developed for explaining the use of ontologies in the product tracking system.

Product tracking for the entire business world, and not just for a single supply chain, can be a reality if all the major business groups agree on common protocols, neutral languages and ontologies. In addition to the product tracking system intelligent agents can be used in virtual markets as brokers, they can handle negotiations for contracts or tenders, and gather data and information from a various sources. Majority of the web content today is not designed to be understood by computers. It is still the human brain and effort that has to make some sense out of the flood of data. The inventor of the Web, Tim Berners-Lee, believes that by using semantics and ontologies the Web will be structured into web pages that have a meaningful content for software agents who will roam from page to page carrying out the various tasks assigned to them by their users [23]. There is no doubt that agents will increasingly take over all those jobs from humans that can be explained in vocabularies, represented in neutral languages and defined by axioms and rules.

Acknowledgments

This material is based upon work supported by the National Science Foundation under Grant No. EPS-9874732 and matching support from the State of Kansas.

References

1. Kitajima, H., Masuoka, R., & Maruyama, F., 2000, "Integrating information and knowledge with software agents," *Fujitsu Scientific & Technical Journal*, 32, 2, 162 – 174.
2. Schwartz, B., 1997, "Tracking down need." *Transportation & Distribution*, 38, 50.
3. Fox, M., Barbuceanu, M., & Teigen, R.,2000, "Agent-oriented supply chain management," *The International Journal of Flexible Manufacturing Systems*, 12, 165–188.
4. Chidambaram, S., Whitman, L., & Cheraghi, H.,1999, " A supply chain transformation methodology, *Proceedings of The 4th Annual International Conference on Industrial Engineering Theory, Applications and Practice*, San Antonio, TX, USA.
5. Zimmerman, R., Hofmann, O., Bodendorf, F., Gunther, G., & Huber, A., 2001, "Agent based tracking and tracing for controlling fulfillment processes in supply chains," *Proceedings of the Second Workshop on Agent-Based Simulation*, Passau.
6. Gruber, T., Pang, D., & Rice.J., 1990, "Ontolingua: A language to Support Shared Ontologies," Knowledge Systems Laboratory website: <ftp://ftp.ksl.stanford.edu/pub/KSL_Reports/KSL-90-84.Z> (September 2001).

7. Whitman, L., Rogers, K., Johnson, M., & Huff, B., 1999, “Understanding the Supply Chain Impact of a Manufacturing Process Change,” *Proceedings of the Portland International Conference on Management of Engineering and Technology*, Portland.
8. Fikes, R., Farquhar, A., & Pratt, W., 1996, “Information brokers: Gathering information from heterogeneous information sources,” *Proceedings of the Ninth Florida Artificial Intelligence Research Symposium, Florida*, 192-197
9. Chen, Y., Finin, T., Labrou, Y., & Cost, S., 1999, “A negotiating agents for supply chain management,” *Proceedings of The AAAI Workshop on Artificial Intelligence for Electronic Commerce,* Orlando, Florida.
10. Chen, Y et al, 1999, “A negotiation-based multi-agent system for supply chain management,” *Proceedings of the Agents'99 Workshop*: Agent-Based Decision-Support for Managing the Internet-Enabled Supply-Chain, Seattle, WA, 15-20.
11. Genesereth, M., & Ketchpel, S., 1994, “Software agents,” *Communications of the ACM*, 37, 48-53.
12. Genesereth,M., & Fikes,R., 1992, “Knowledge Interchange Format Version 3.0, Reference Manual,” Stanford University website < ftp://ftp.ksl.stanford.edu/pub/KSL_Reports/KSL-90-84.Z> (September 2001).
13. Masuoka, R., Sato, A., & Genesereth, M., 1999, “KIF as a FIPA-CLL content language : An ontology for actions,” Intelligent Systems Laboratory, Fujitsu Laboratories Ltd.
14. Finin, T., et al., 1994, “KQML: An information and knowledge exchange protocol,” *Proceedings of the third International Conference on Information and Knowledge.*
15. Finin, T et al., 1994, “Specification of the KQML agent-communication language,” Report of The DARPA Knowledge Sharing Initiative, External Interfaces Working Group.
16. Gruber, T. R., 1993, “Toward principles of the design of ontologies used for knowledge sharing,” *Proceedings of the International Workshop on Formal Ontology in Conceptual Analysis and Knowledge Representation,* Padova, Italy.
17. Gruber, T. R., 1993, “A Translation Approach to Portable Ontology Specifications,” *Knowledge Acquisition,* 5, 2, 199-220.
18. Farquhar, A., Fikes, R., Pratt, W., & Rice, J., 1995, “Collaborative ontology construction for information integration,” Technical report KSL-9563of Knowledge System Laboratory, Stanford University, CA, USA.
19. Farquhar, A., Fikes, R., & Rice, J., 1997, “Tools for assembling modular ontologies in Ontolingua.” Knowledge Systems Laboratory website <www.ksl.stanfords.edu> (September 2001).
20. Farquhar, A., Fikes, R., & Rice, J., 1997, “The Ontolingua server: a tool for collaborative ontology construction,” *International Journal of Human-Computer Studies*, 46, 6, 707-727.
21. Ontolingua website <www.ksl.stanford.edu>.
22. Noy, N., & McGuinness, D., 2001, “Ontology development 101: A guide to creating your first ontology,” Stanford Knowledge Systems Technical Report KSL-01-05 web site <http://www.ksl.stanford.edu/people/dlm/papers/ontology-tutorial-noy-mcguinness-abstract.html> (September 2001.)
23. Berners-Lee, T., Hendler, J., & Lassila, O, 2001,May, “The Semantic Web.” *Scientific American*.