December 2004

Supply Chain System Analysis and Modeling Using Ontology Engineering

Charu Chandra
University of Michigan - Dearborn

Armen Tumanyan
University of Michigan - Dearborn

Supply Chain System Analysis and Modeling Using Ontology Engineering

Charu Chandra
University of Michigan – Dearborn
Industrial and Manufacturing Systems
Engineering Department
charu@umich.edu

Armen Tumanyan
University of Michigan – Dearborn
Industrial and Manufacturing Systems
Engineering Department
armen@umich.edu

ABSTRACT

One of the primary objectives of supply chain (SC) system analysis and modeling is to develop conceptual design of organizational and process knowledge models, which facilitates optimal supply chain management. SC knowledge modeling consists of two components: 1) modeling SC workflows, and 2) capturing and organizing knowledge necessary for managing them. Workflow modeling deals with handling activities to generate and utilize knowledge, whereas Ontology Engineering formalizes knowledge content. This paper proposes a framework comprising both aspects of knowledge modeling. For knowledge representation, two standards are introduced: situation calculus, and supply chain markup language. The former is utilized for capturing process logic with mathematical expressions, and the latter for coding this logic with a computational language. An example of production scheduling for a steel supply chain is provided for illustration.

Keywords

Ontology Engineering, Supply Chain Modeling, Knowledge Management, Information Systems Design.

INTRODUCTION

The problem of system analysis, modeling, and representation has always been important from the perspective of understanding the organization, its processes, and supporting process management. A system model is a description of its constituents, viz., goals, processes, relationships between processes, and mechanisms for managing these processes. The fundamental question this paper is seeking to address is, "how SC can be modeled to facilitate the development of supporting information system, defined appropriately for problem solving methods". Towards this end, Crubezy and Musen (2003) suggest knowledge management methodology based on ontologies. The rationale is to develop problem-solving methods as independent components for reuse in decision modeling application systems. This paper adopts their approach and extends it to propose a framework, whereby problem specific ontologies can be modeled using a systematic architecture of supply chain (SC) system analysis and modeling.

A holistic approach is applied to SC modeling, where SC is considered as a system consisting of processes interrelated to each other. Each process may have various views reflecting different system management perspectives. In order to design these process views, conceptual models are proposed, which are graphical representations used for symbolizing both static (things and their concepts) and dynamic phenomena (events and processes), respectively (Wand and Weber, 2002).

One of the primary objectives of SC system analysis and modeling is to develop conceptual design of organizational and process knowledge models, which facilitate optimal supply chain management. SC knowledge modeling consists of two components: 1) modeling SC workflows, and 2) capturing and organizing knowledge necessary for managing these workflows. For knowledge organization, this paper suggests utilizing concepts from Ontology Engineering. Workflow Modeling deals with handling activities to generate and utilize knowledge, whereas Ontology Engineering formalizes knowledge content. The majority of knowledge that needs to be managed in an organization is generated and utilized in-house along the workflow of organization (Kang, Park, and Kim 2003). Therefore, it is meaningful as well as useful to build the process knowledge content, based on the structure and logic of processes and tasks defined in the workflow.

BACKGROUND AND MOTIVATION

Supply chain (SC) is a network of facilities, distribution options, and approaches utilized to effectively integrate suppliers, manufacturers, and distributors through performing functions of procurement of materials, transformation of these materials into intermediate and finished products, and their distribution to customers in the right quantities, to the right locations, and at the right time, in order to meet required service level with minimal cost. In order to make these functions and processes streamlined, these have to be managed in a holistic and synergistic manner. The former considers SC as a whole consisting of parts, and the latter is a set of independent components collectively working on common problems. Centralized control is not very effective for managing SC. Fox, Barbuceanu and Teigen, (2000) propose agent oriented supply chain management (SCM), where SC is viewed as being managed by a set of intelligent agents, each responsible for one or more tasks in SC, and interacting with other agents in the planning and execution of their responsibilities. Every agent requires specific knowledge for its operation. Ontologies provide the vocabulary for representing domain or problem knowledge. These are also crucial for enabling knowledge level interoperations of agents.

The role of system analysis and conceptual modeling is acknowledged as a powerful tool for understanding system processes, tasks, and activities (Carlsen, 1997). Typically conceptual models are required to:

- understand SC organization,
- document SC processes,
- specify functional and user requirements for SC system,
- design information system to support processes, and
- evaluate the change in business process reengineering.

The necessity of SC modeling arises based on several reasons:

- The extent of knowledge becomes intractably large,
- Business units are geographically decentralized, but more closely networked,
- The importance of collaboration among individual workers, and
- Challenges faced in eliciting requirements when user cohorts are large, decentralized and unknown.

Research findings in various fields of study spur the development of knowledge-based frameworks for conceptual modeling. These incentives can be specified as:

- The emergence of object-oriented approach,
- The use of process models,
- The potential of conceptual models to assist business process reengineering,
- Ontology application in domain and process knowledge representation, and
- Agent technology for automatic process management and control.

THE STATE OF THE ART

The system analysis and modeling framework proposed in this paper consists of two components - workflow conceptual modeling and ontology engineering. The state of the current research in these two fields of study are examined and presented in this section.

Workflow modeling

Workflow modeling is identified as the means to define and administer business processes automatically. Most of the research initiatives in this field are focused on information system design for implementing real-time collaboration systems and process automation. The workflow management coalition (WFMC) (www.wfms.org) has established standards for reference model to design workflow and specifications of data and documents to realize interpretability between workflow systems. According to WFMC, workflow is "the automation of a business process, in whole or in part, during which documents, information or tasks are passed from one participant to another for action according to a set of procedural rules. Process modeling is the most important component of Workflow management. This and other three components viz., goal, structure and object views are described in IBM (2002). This paper considers only process view of workflow management.

Process modeling enables systematic analysis of business, focusing on tasks (functions) that are performed regularly, controls to ensure their proper performance, resources needed to perform a task, results of a task, and inputs (raw materials) on which the task operates. Several research efforts have proposed methodologies to improve enterprise performance through modeling and design of business processes (Presley and Liles, 2001; Shunk, Kim and Nam, 2003). The common thread in these approaches is the use of process models as an aid to understand and design systems. A process can be looked from different perspectives depending on the type of information required. Previous research have defined a number of views with corresponding methodologies, viz., integrated definition (IDEF), computer integrated manufacturing open systems architecture (CIM-OSA), architecture of integrated information system (ARIS), Petri Nets. Recently, with the emerging growth of object-oriented paradigms for analyzing and designing systems, unified modeling language (UML) is in use for business process design.

Ontology engineering

Ontologies have shown their usefulness in various areas of applications, such as knowledge representation, intelligent information integration, expert systems, active database systems, etc. Ontology refers to an engineering artifact, constituted by a specific vocabulary, used to describe a certain reality, in addition to a set of explicit assumptions regarding the intended meaning of words in the vocabulary (Guarino, 1998). Ontology has been applied for capturing the dynamics of the system for which situation calculus is utilized (Levesque, Reiter, Lesperance, Lin and Scherl, 1994).

A survey of literature reveals three dimensions of ontology engineering process. The first dimension is the building stage, consisting of several activities: specification, conceptualization, formalization, implementation, and maintenance. The second dimension is the type of ontology: domain and problem. The third dimension is ontology modeling components. Chandra and Tumanyan (2004) specify three components of SC ontology: 1) domain concepts with their relationships, 2) axioms for representing rules held on this domain, and 3) problem solving algorithms, if it is a problem ontology.

Knowledge intensive workflow management

Researchers trying to model workflows and make processes managed effectively are offering techniques borrowed from the knowledge management discipline. Casati, Ceri, Pernici and Pozzi, (1996) propose automatic derivation techniques of active rules that may form workflow specification. For describing and representing these rules, active database systems have been utilized. Through active rules, workflow performance is represented with operational semantics. Workflow management using e-commerce is proposed by Basu and Kumar (2002). For modeling workflows, and controlling and monitoring their performances, organizational meta-models are proposed to be designed. Meta-models incorporate organizational constraints as they relate to resources, roles, tasks, policies, etc. Knowledge management technique is introduced in Kang et al. (2003) to combine process, and content management. Process management is concerned with handling activities to generate and utilize knowledge, whereas content management deals with the knowledge content.

CONCEPTUAL FRAMEWORK

This paper proposes an approach to SC system analysis and a conceptual model design in the form of ontologies. The framework consists of three major modeling environments: SC system, workflow management, and ontology engineering (Figure 1). The result of SC modeling is a unified representation of SC processes in a hierarchy identifying relationships among them. Workflow management is concerned with process modeling. At this stage, processes are documented with explicit models. Ontology engineering deals with capturing knowledge for every process and task designed in the previous stage. As a result of application of these models, knowledge modules in the form of ontologies are developed.

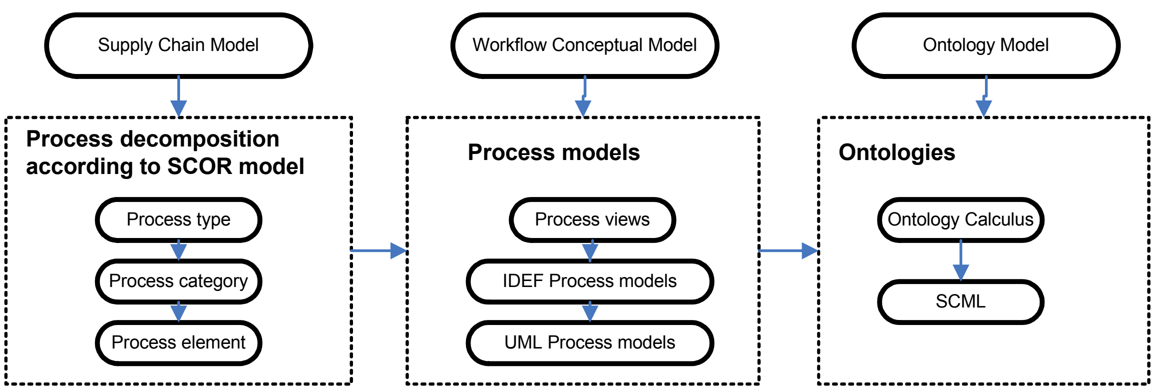

Figure 1. Supply chain analysis and modeling conceptual framework

Supply chain model

SC model is the collection of business processes, their relationships, and sets of characteristics, necessary for evaluating these processes. For SC model representation, supply chain operations reference-model (SCOR) is proposed (www.supply-chain.org). SCOR integrates concepts of business processes, benchmarking, and best practices into a cross-functional framework. SCOR model builds a hierarchy of SC processes, which can be divided into three levels: Process type, process category, and process element. Process type defines five basic management processes in SC (Plan, Source, Make, Deliver, Return) that provide the organizational structure of the SCOR-model.

The second level defines three process categories: planning, execution, and enable. A planning element is a process that aligns expected resources to meet anticipated demands. Execution processes are triggered by planned or actual demand that changes the state of products. They include scheduling and sequencing, transforming materials and services, and moving product. Enable processes prepare, maintain, and manage information or relationships upon which planning and execution processes rely. The SCOR second level also defines criteria for process classification, e.g. for Make process type, three categories are identified in SCOR: M1 make-to-stock, M2 make-to-order, and M3 engineer-to-order. The third level presents detailed process elements' information on each process category described at the second level, particularly process input, output, process flows, performance attributes, and best practices for their implementation.

Workflow modeling

The fourth, implementation level is out of scope of SCOR. This level is defined using a combined methodology, a best breed of IDEF and UML. Workflow or process modeling aims to represent processes specified in SCOR third level as a collection of tasks executed by various resources within a SC. Each process transforms a specific set of inputs into a specific set of outputs to achieve some functional goals. Kim, Kang and Kim (2000) in their study of WFMS suggest nested process modeling, where each business process can be broken down into sub-processes or tasks. A structure can be provided for hierarchically arranging them in a taxonomy, making it easier to grasp the relationship between processes and tasks. In turn, tasks can be decomposed into activities yielding another level in problem taxonomy. SC has some specifics that cannot be adequately represented with WFMS, such as, distributed nature, loose connection among SC members, different process management standards, etc. The framework for workflow modeling proposed in this paper will be integrated with standards promoted by WFMC.

The above-described features of workflow modeling can be captured by using explicit models. In comparing various business process modeling methods provided by Lin, Yang and Pai (2002), four methods have been selected: IDEF0, IDEF1, IDEF3, and Object-oriented modeling with UML formalism. IDEF0 method is designed to model decisions, actions, and activities of a SC targeted to analyze its functional perspectives. IDEF1 is an information modeling method used in identifying 1) information collected, stored, and managed by a SC, 2) rules governing the management of information, 3) logical relationships within enterprise reflected in information, and 4) problems resulting from lack of good information modeling (Mayer, Benjamin, Caraway and Painter, 1995). IDEF3 describes processes as sequence of events and activities. It is a scenario-driven process modeling technique based on precedence and causal relationships between events and situations. IDEF3 model provides the method for expressing and documenting SC domain experts' knowledge about how a particular process works, in contrast to IDEF0, which is concerned with what activities a SC performs.

IDEF formalism documents processes with semantic diagrams, which is a part of workflow management. The other part is how to manage these processes intelligently by designing information system to support functions and activities. Transferring the business model into a software model is necessary to design adequate information system. An object-oriented process modeling technique is proposed to accomplish the transformation from business to software view. UML modeling formalism provides a unique opportunity in this respect. UML offers a library of diagrams to semantically present process views captured by IDEF formalism. UML meta-models define constructs that designers can use for modeling a software system. This paper offers the best breed of these two techniques, whereby IDEF is utilized for handling low-level process granularity, and UML offers object representation and migration to software applications.

Ontology engineering

Ontology development is the last stage of the SC conceptual modeling framework proposed in this paper. Chandra and Tumanyan (2004) have proposed ontology development as a backbone for SC information system design, where ontology participation in information system is described and ontology development stages are presented. The ontology-engineering framework presented in this paper proposes two ontology development specifications from the perspectives of knowledge and software engineers. The former is situation and predicate calculus and the latter a new XML language specification, called supply chain markup language (SCML). The proposed ontology engineering framework is depicted in Figure 2.

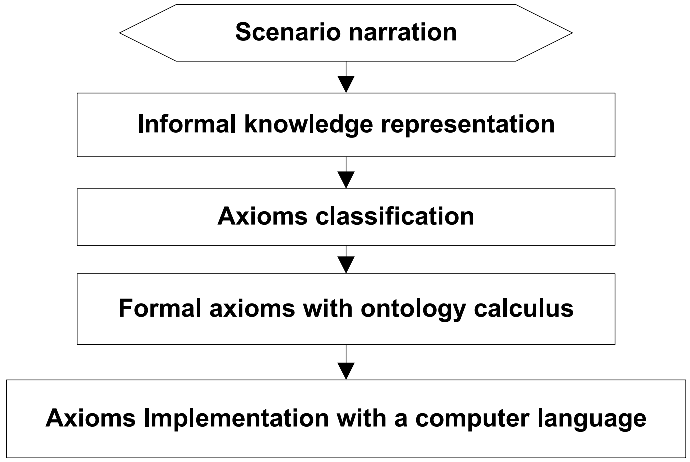

Figure 2. Ontology engineering conceptual framework

Processes documented in workflow modeling environment are studied with domain experts and narrative description of the necessary knowledge is captured in a text or another word processing format. Scenario narration describes the case study in English and presents the situation that is to be modeled. It has a form of story problems, or examples that are not adequately addressed by existing information system. Scenario narration may also contain problem analysis and possible solutions to the described problem. Informal knowledge representation captures questions that ontology must address. These are English descriptions presented in a modular way. For each module, an axiom is to be developed. Axioms classification structures capture modules. Ideally axioms should be arranged in hierarchy with higher-level questions requiring the solution for the lower level questions. Formal ontology axioms formulation can be accomplished through ontology calculus. The latter is based on situation calculus and is defined for SC ontology representation. Ontology modeling calculus contains knowledge formulated for both type of Ontologies: domain and problem (Chandra and Tumanyan, 2004). Implementation of axioms is the coding of formal mathematical models in a software language. XML is utilized as the knowledge representation language for deploying ontology constituents in the proposed framework.

For each process item (process, task, or activity), ontology or a set of ontologies is designed. Ontologies conceptualize the knowledge necessary for planning and executing these process items. The knowledge encapsulated in ontologies consists of three components: data model, axioms defining constraints and rules held on data model, and algorithms, which are step-by-step conditional descriptions of process flows.

CASE STUDY: AUTOMOTIVE INDUSTRY SUPPLY CHAIN

A prototype of the proposed approach is developed with a case study describing a decision support system (DSS) for steel processing and shipment (SPS) SC. SCOR definitions are used for building high-level process flow. Process models are designed for activity view (IBM 2002). IDEF graphical models are utilized for decomposing SCOR processes into tasks presenting low level flows of activities. UML class diagrams are used in transforming concepts and their relationships regarding SPS configuration / scheduling problem into software constructs. UML class diagram utilization is twofold. It can be used as a pseudo-code for developing a software application. UML class diagram can also serve as a semantic network for the configuration domain, and be a part of the domain ontology. For the task ontology example, two specifications are demonstrated – situation calculus and SCML.

Problem statement

Steel processing and shipment (SPS) is a multi-stage manufacturing process. The automobile SC consists of several raw material suppliers and a stamping plant (Figure 3), which bus steel from it to produce assembly components.

The stamping plant consists of blanking, pressing and assembly departments. The blanking department cuts the raw steel into rectangular pieces. The pressing department stamps the blanks into parts. There are substantial setup times. Welding and other operations are performed on stamped parts at the metal assembly department.

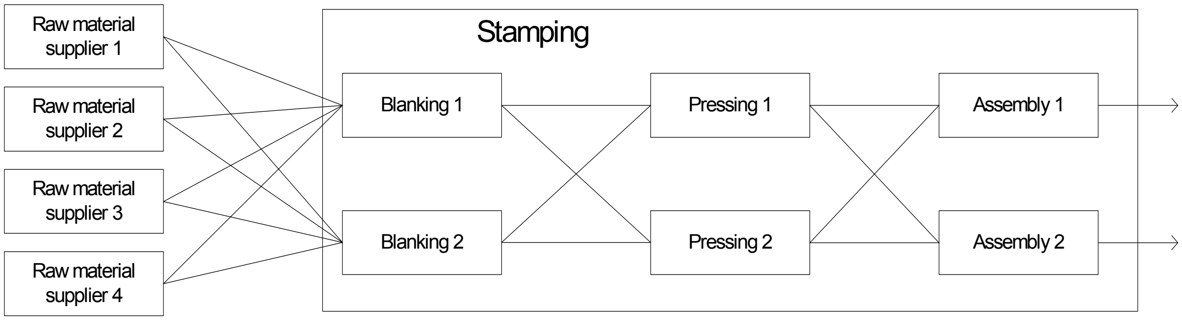

Figure 3. Steel processing and shipment supply chain

Supply chain model

The case study described is concerned with specific problems in SC. These are supplier selection, raw materials delivery, production of steel sub-assembly components, and delivery using various carriers. SPS SC model is a projection of the SCOR process taxonomy regarding these specific issues identified in the problem statement. Figure 4 depicts the sequence in which these issues are addressed in the SC.

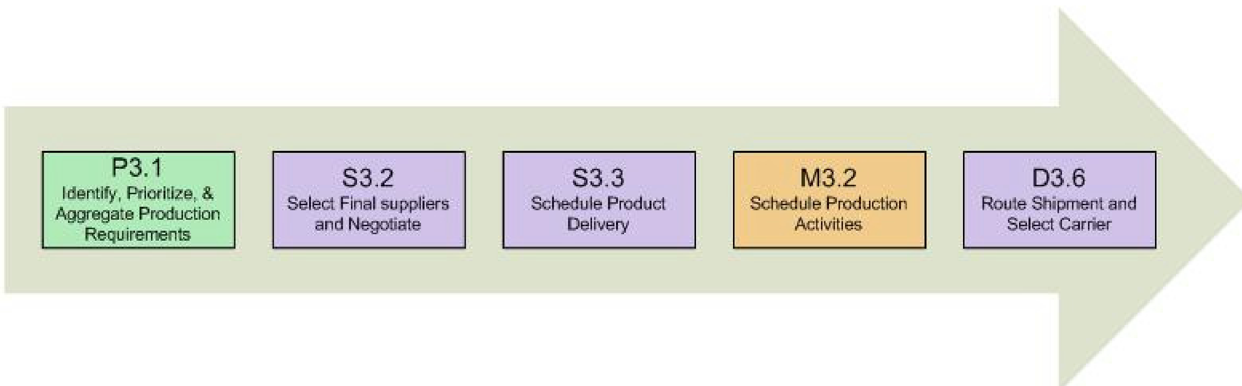

Figure 4. SCOR model for steel shipment and processing supply chain

P3.1 is a process of identifying, prioritizing, and considering as a whole with constituent parts, all sources of demand in the creation of the steel product. S3.2 is the identification of the final supplier(s), based on the evaluation of supplier qualifications and the generation of a contract defining costs and terms and conditions of product availability. S3.3 is Scheduling and managing the execution of the individual deliveries of product against the contract. The requirements for product deliveries are determined based on the detailed sourcing plan. M3.2 are plans for the production of specific parts, products, or formulations in specified quantities and planned availability of required sourced products, and the scheduling of operations to be performed in accordance with these plans. Scheduling includes sequencing, and, depending on the factory layout, any standards for setup and run. In general, intermediate production activities are coordinated prior to the scheduling of operations to be performed in producing a finished product. D3.6 is the process of consolidating and routing shipments by mode, lane, and location. Carriers are selected and shipments are routed.

These processes are captured from SCOR model level three representations. In SCOR specification, ontology designers can find performance attributes and best practices for each process. Ontology at this level is considered as the highest-level abstraction of knowledge capturing. This paper discusses only low level ontologies that can be ultimately used by decision modeling agents.

M3.2 – Schedule Production Activity process model

Process model development consists of two stages: 1) two IDEF0 diagrams are developed for decomposing "Schedule Production Activity" process into tasks and activities, and 2) UML class development for representing the process model in a suitable format for building information system elements. This paper advocates the pivotal role of ontology in information system design. Consequently, these process models will be used for designing domain and task ontologies.

The IDEF0 diagram in Figure 5 depicts a simple decomposition of "Schedule Production Activity" process decomposed into four elementary tasks. Three tasks are identified as:

1. Forecast finished goods production and material requirements.
2. Plan production.
3. Schedule production.

Relationships among these activities identify input and output information necessary for accomplishing these tasks. Further decomposition of one of its tasks is depicted in Figure 6. Four activities are identified in "Plan Production" task: schedule load capacity, schedule finishing capacity, determine outsourcing requirements, and generate production plan.

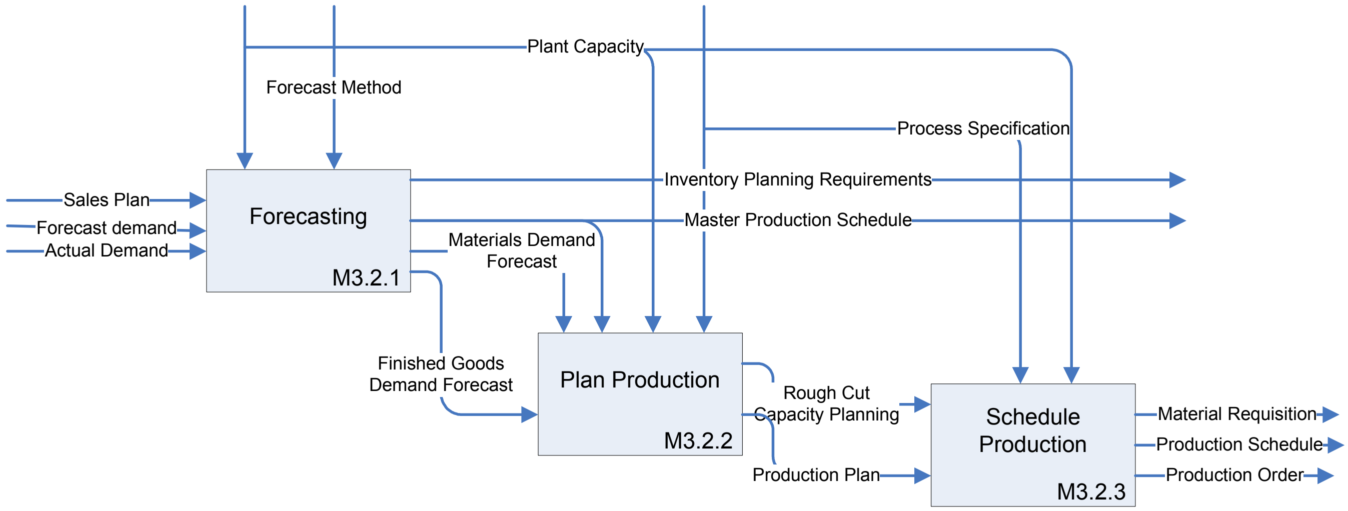

Figure 5. IDEF0 diagram for the "Schedule Production Activity" process

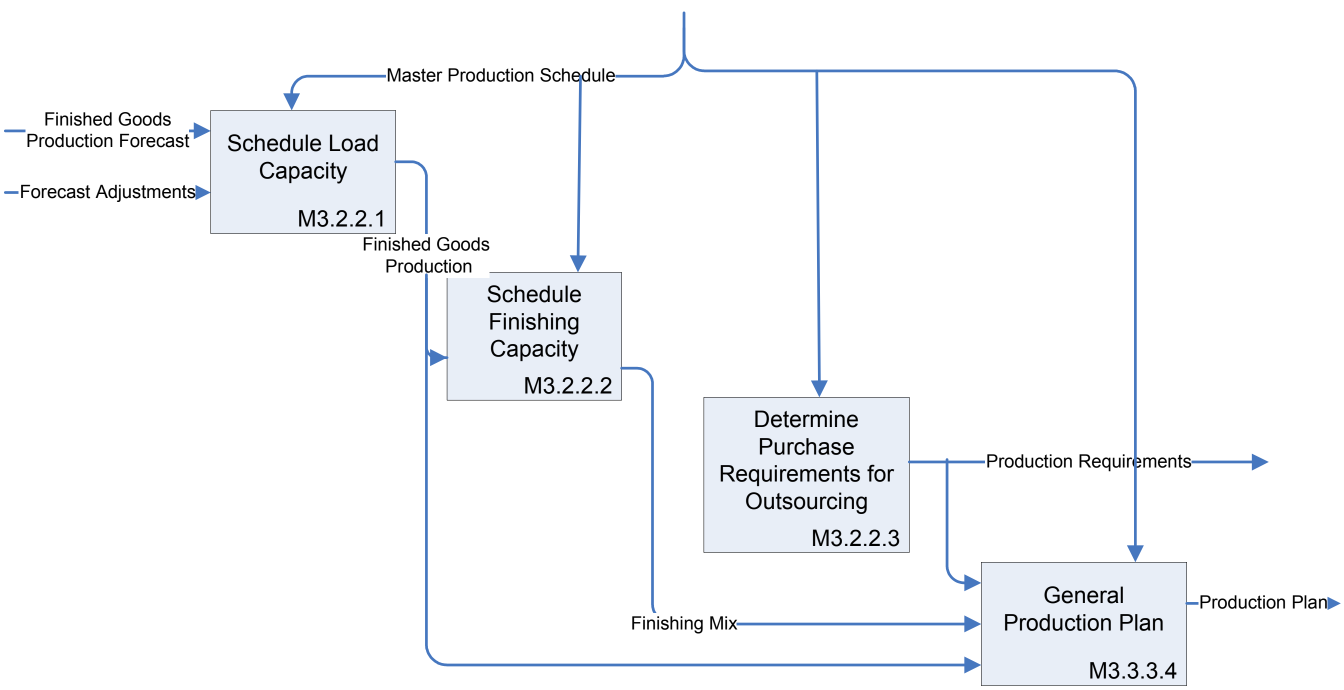

Figure 6. IDEF0 diagram for the "Plan Production" process

The second part of workflow modeling is the representation of business processes with constructs that can be used for designing ontologies as information system components. UML model development starts with studying the process models and their information needs. Thorough analysis of SPS SC reveals a list of parameters necessary for building the hierarchy of concepts of configuration / scheduling process. Concepts are gathered into classes, and classes are related to each other. An analyzed system for the steel SC problem is depicted in Figure 7. The central class is production unit, which has transportation facilities. Association link defines a type of relationship, where transportation is presented with one parameter inside productionUnit class. This parameter is an object, which has multiple attributes. Resource class is associated with productionUnit class, and has object parameter resourceAttribute. Product class is associated with productionUnit with many to one relationship. Product has object parameters demand, productMaterials, and production. typeProb class is for defining distribution function of three attributes for production class, viz., BreakdownDuration, BreakdownFrequency, and defectiveness.

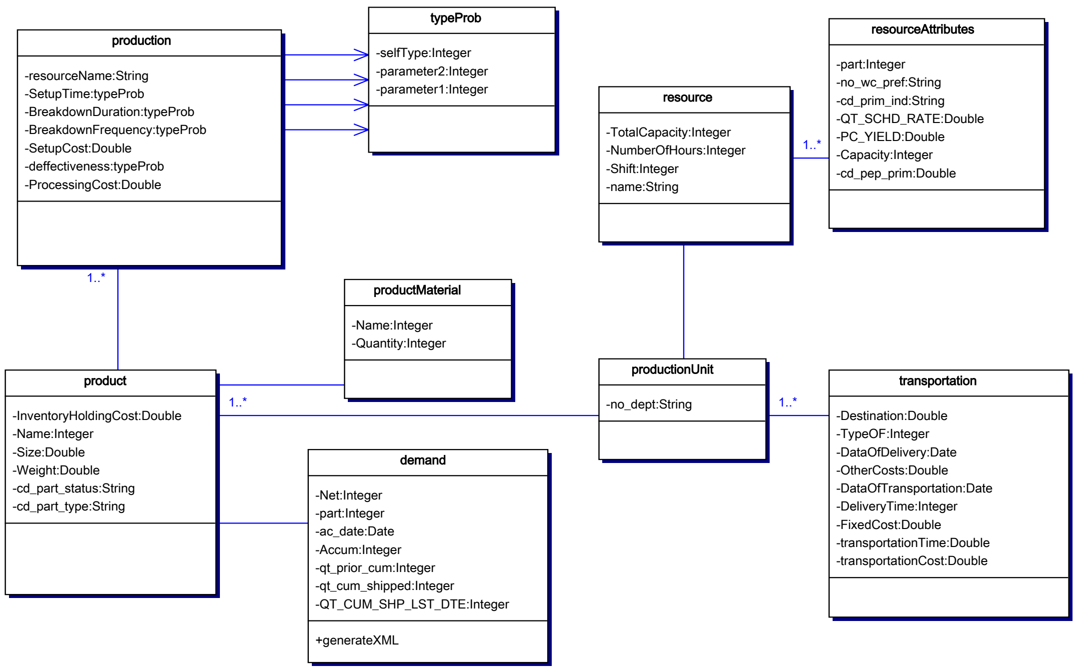

Figure 7. UML class diagram for configuration / scheduling process

Supply chain ontology engineering

SC ontology engineering consists of development of its three components: semantic network, which is concepts and their relationships; axioms defining constrains on concepts and other horizontal relationships among them, and algorithms, which are step-by-step procedures for managing tasks and activities. An UML model defines the first component. Formalizing axioms and algorithms can be implemented through thorough analysis of developed process models.

The proposed framework describes steps necessary for formalizing ontologies. Scenario narration is provided in problem statement section. SPS SC informal knowledge representation is accomplished by examining process models. Observations held on configuration problem domain are written down as English sentences. Examples of informal representation of knowledge are as follows:

- For every product, there should be a demand,
- If a product is ordered, and its resource is busy with other product without order, switch the resource load,
- Inventory level should be less than the maximum allowed,
- Resource utilization cannot be more than its capacity,
- If a product is assigned to a resource, all materials should be available,
- Processes can start when resources and materials are available.

Axioms and algorithm capture is a process of a search of rules held in the domain of interest, for which ontology is to be built. Rules and regulations defined in axioms and algorithms are based on concepts identified in UML model and are formulated in the form of equations relating these concepts to each other.

The theory for axiom and algorithm representation is based on situation calculus and predicate calculus for representing the dynamically changing world (Lesperance, Levesque, Lin and Scherl, 1995). Situation theory views domain as having a state (or situation). When the state is changed, there is necessity for an action. Predicate theory defines conditions on which specific actions can be taken. According to the framework introduced in this paper, ontology is to capture both dynamics.

Examples of formal representation of axioms for above described situation are provided below.

$$Exist(demand, Product)$$

$$Less(MaxInventory, CurrInventory)$$

The first axiom states that, if there is a product, its demand should exist. The second axiom constrains current inventory with the maximum inventory level. Axioms are reusable knowledge constructs and can be used for various problem representations, and so have to be shared among these problems.

An example of an algorithm can be the formula according to which order size is calculated. Inventory replenishment algorithm assumes checking the inventory level periodically. If it is less than a predefined level, place an order equal to a specified value. This narrated knowledge can be formalized using ontology calculus as follows:

$$Poss(do((L * AVG + z * STD) = s) > Il) \equiv MakeOrder(s - Il)$$

Where s is the reorder level, L is lead-time, AVG, STD are forecasted demand means and standard deviation, z is customer service indicator. If inventory level (IL) is less than the calculated reorder level, an order is placed (Order), which is equal to the difference of reorder and inventory levels

Ontology calculus, like IDEF process model, can document a process or a task. For using these processes (or tasks) in information system and applying process management techniques, a software construct is required for their representation.

For ontology representation, different programming languages and standards have been utilized. Ontolingua and KIF (Farquhar, Fikes and Rice, 1997) and OIL (Ontology Interchange Language) (Fensel, Harmelen, Horrocks, McGuinness and Patel-Schnaider, 2001). XML is emerging as a standard for communication between heterogeneous systems and is widely used on the Internet. These environments present new opportunities for knowledge representation and acquisition. This opportunity has two aspects. First, XML documents can easily be translated into knowledge representation format and parsed by problem solving environments or domains. Second, XML can directly connect with data storage repositories (RDBMS or ERP systems), thus providing database queries to be more expressive, accurate and powerful. These two objectives can be achieved by enhancing the semantic expressiveness of XML, especially XML data schemas (XSD). This research paper proposes a new supply chain markup language (SCML) for presenting knowledge about SC. The specification of SCML is formulated as a XSD data schema depicted in Figure 8.

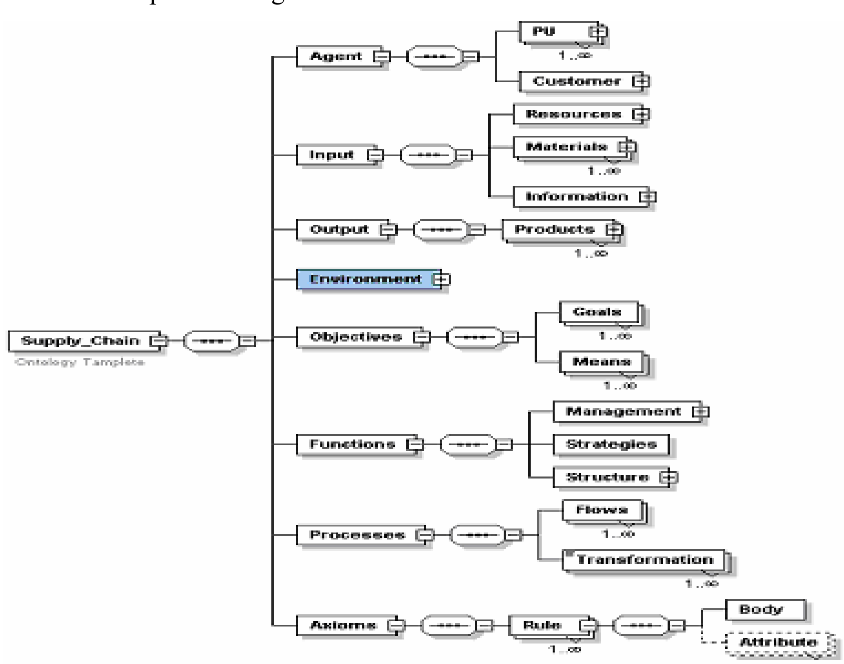

Figure 8. Data schema for supply chain markup language

SCML is the computational implementation of the system taxonomy introduced by Chandra and Tumanyan (2003). Seven components (Input, Output, Process, Function, Environment, Agent, and mechanism) of System adopted from Nadler (1990), constitute the data model. Axiom entity is utilized for specifying axioms and algorithms.

A fragment of SCML is depicted in Figure 9. It defines the entity "Axioms", elements it may have, and entities it may contain. Axioms entity class may have 1 or many "Rules" ("unbounded") entities, which may have "Attributes entities (0 or many). "Argument" entity may have two attributes: "Name" and "Description". The entity "Rule" may have one and only one "Body" entity, and two attributes.

```
<xs:element name="Axioms">
    <xs:complexType>
        <xs:sequence>
            <xs:element name="Rule" maxOccurs="unbounded">
                <xs:complexType>
                    <xs:sequence>
                        <xs:element name="Body"/>
                        <xs:element name="Attribute" minOccurs="0" maxOccurs="unbounded">
                            <xs:complexType>
                                <xs:attribute name="Name" type="xs:string" use="optional"/>
                                <xs:attribute name="description" type="xs:string" use="optional"/>
                            </xs:complexType>
                        </xs:element>
                    </xs:sequence>
                    <xs:attribute name="Number" type="xs:string" use="optional"/>
                    <xs:attribute name="Name" type="xs:string" use="optional"/>
                </xs:complexType>
            </xs:element>
        </xs:sequence>
        <xs:attribute name="Name" type="xs:string" use="optional"/>
        <xs:attribute name="ID" type="xs:string" use="optional"/>
    </xs:complexType>
</xs:element>
```

Figure 9. SCML fragment: axioms

Informally represented and presented by ontology calculus, rules are introduced as SCML data files in Figure 10. This XML file structure and content is defined by SCML schema depicted in Figure 9.

```
<SupplyChain>
    <Axioms>
        <Rule Number="1" Name="Service level is 100%">
            <Argument Name="Inv" Description="Inventory"/>
            <Argument Name="Dm" Description="Demand"/>
            <Body>Inv&gt;=Dm</Body>
        </Rule>
        <Rule Number="2" Name="Product Resouce assignment">
            <Argument Name="P" Description="Product"/>
            <Argument Name="R" Description="Resource"/>
            <Body>FOR EACH P EXIST R</Body>
        </Rule>
        <Rule Number="4" Name="Process Production Unit correspondance/">
            <Argument Name="R" Description="Resource"/>
            <Argument Name="PU" Description="Production Unit"/>
            <Body>FOR EACH R EXIST PU</Body>
        </Rule>
        <Rule Number="5" Name="Resource utilization should be less then its capacity">
            <Argument Name="R_U" Description="Resource utilization"/>
            <Argument Name="R_C" Description="Resource capacity"/>
            <Body>R_U&gt;=R_C</Body>
        </Rule>
        <Rule Number="6" Name="Processes can start when resource and materials are available">
            <Argument Name="R" Description="Resource"/>
            <Argument Name="M" Description="Material"/>
            <Argument Name="Pr" Description="Process"/>
            <Body>For each Pr exist all R, M</Body>
        </Rule>
        <Rule Number="10" Name="If a product is assigned to a resource all materials should be available">
            <Argument Name="P" Description="Product"/>
            <Argument Name="R" Description="Resource"/>
            <Argument Name="M" Description="Material"/>
            <Body> IF P.resource CHECK ALL P.materials</Body>
        </Rule>
        <Rule Number="17" Name="Each product should have demand, internal or external">
            <Argument Name="P" Description="Product"/>
            <Body>FOR EACH P EXIST (P.demand or P.order)</Body>
        </Rule>
    </Axioms>
</SupplyChain>
```

Figure 10. Ontology fragment: Axioms

Axioms are building blocks for ontology engineering. Along with concepts and relations, axioms can present the knowledge in a formalism that can be accessed and processed by software agents.

CONCLUSION

As a result of this research effort, an approach is proposed for modeling SC as a collection of ontology constructs. The process of SC domain analysis and knowledge models design is presented. It consists of (1) modeling SC as a set of processes (SCOR), (b) modeling SC business processes from generic process to specific tasks and activities (IDEF), (c) transforming process models into software meta-models (UML), and (d) engineering ontologies based on previous analysis.

This research effort is an organic part of overall research initiative of SC decision support system design. Ontologies developed based on principles presented in this paper have been utilized for managing SPS SC at an industrial partner. This research complements the creation of ontology development environment, where domain experts, knowledge workers, and software engineers will be able to work collectively on building enterprise ontology.

REFERENCES

1. Basu, A. and Kumar, A. (2002) Research Commentary: Workflow Management Issues in e-Business *Information Systems Research*, *INFORMS,* 13, 1, 1–14.
2. Carlsen, S. (1997) Conceptual Modeling and the Composition of Flexible Workflow Models, *Doctoral dissertation*.
3. Casati, F., Ceri. S., Pernici, B. and Pozzi, G. (1996) Deriving active rules for workflow enactment, *Proceedings of 7th International Conference on Database and Expert Systems Applications,* Springer-Verlag, New York, NY, 94-110.
4. Chandra C. and Tumanyan A. (2004) Ontology Driven Knowledge Design and Development for Supply Chain Management*, Proceedings of the 13th Annual Industrial Engineering Research Conference IERC-2004*, Houston, Texas USA.
5. Chandra, C. and Tumanyan, A. (2003) System Taxonomy development and application, *Proceedings of the 12th Annual Industrial Engineering Research Conference IERC-2003, Portland, Oregon USA.*
6. Crubézy, M. and Musen, M. A. (2003) Ontologies in Support of Problem Solving, *SMI Report Number: SMI-2003-0957.*
7. Farquhar, A., Fikes, R. and Rice, J. (1997) The Ontolingua Server: A tool for collaborative ontology construction, *International Journal of Human-Computer Studies*, 46, 6, 707—727.
8. Fensel, D., Harmelen, F., Horrocks, I., McGuinness, D. L. and Patel-Schnaider, P. F. (2001) OIL: An Ontology Infrastructure for Semantic Web, *IEEE Intelligent systems*, March/April, 38-45.
9. Fox, M., Barbuceanu, M. and Teigen, R. (2000) Agent-Oriented Supply-Chain Management, *The International Journal of Flexible Manufacturing Systems*, 12, 165–188.
10. Guarino, N. (1998) Formal Ontology and Information systems, *Proceedings of FOIS'98*, Amsterdam, IOS Press.
11. IBM (2002) Business modeling with UML: A Business Process Centered Architecture, *Technical report.*
12. Kang, I., Park, Y. and Kim, Y. (2003) A framework for designing a workflow-based knowledge map, *Business Process Management*, 9, 3, 281-294.
13. Kim, Y., Kang, S. and Kim, D. (2000) WW-FLOW: Web-based workflow management runtime encapsulation, *IEEE Internet Computing*, 4, 3, 55-64.
14. Lesperance, Y., Levesque, H. J., Lin, F. and Scherl R. B. (1995) Ability and knowing how in the situation calculus. *Studia Logica*, 66, 1, 165-186, October.
15. Levesque, H. Reiter, R. Lesperance, Y. Lin, F. and Scherl, R. (1994) GOLOG: A logic programming language for dynamic domains, *Journal of Logic Programming*, 31,1-3, 59-84, April-June.
16. Lin, F., Yang, M. and Pai, Y. (2002) A generic structure for business process modeling, *Business Process Management Journal*, 8, 1, 19-41.
17. Mayer, R. J., Benjamin, P., Caraway, B. E. and Painter, M. K. (1995) A Framework and a Suite of Methods for Business Process Reengineering, in B. Kettinger & V. Grover (Eds.), *Business Process Reengineering: A Managerial Perspective*, 245-90.
18. Nadler, G. (1970) Work Design: A System Concept, Richard D. Irwin Inc., Homewood, IL.
19. Presley, A. R. and Liles, D. H. (2001) A holon-based process modeling methodology, *International Journal of Operations and Production Management*, 21,6, 565–581.
20. Shunk, D. L., Kim, J. and Nam, H. Y. (2003) The application of an integrated enterprise modeling methodology – FIDO – to supply chain integration modeling, *Computers & Industrial Engineering*, 45, 167-193.
21. Wand, Y. and Weber, R. (2002) Research Commentary: Information Systems and Conceptual Modeling – A Research Agenda, *Information Systems Research*, 13, 4, 363-376.