ONTOLOGY-ORIENTED FRAMEWORK FOR VIRTUAL ENTERPRISES

Accomplished within the Project: Future Network-based Semantic Technologies (FUNSET-Science)

Gottfried Koppensteiner, Munir Merdan, Wilfried Lepuschitz

Automation and Control Institute, Vienna University of Technology,Gußhausstrasse 27-29, E376, 1040 Vienna, Austria

Erhard List, Lisa Vittori

Department of Information Technology, Vienna Institute of Technology (TGM), Vienna, Austria

Keywords: Virtual Enterprise, Ontology, Negotiation, Multi Agent System.

Abstract: In current networked organizations knowledge is distributed among the organization and their partners resulting in the loss of transparency regarding the kind and the place of knowledge within the network. Our approach is to use the semantic technology together with software agents in order to improve knowledge capturing, reuse and transfer. Our paper describes an ontology-based multi-agent approach for the knowledge exchange and process control with and within virtual enterprises. Different case-studies with different ontologies are combined with a negotiation ontology, which is used as intercrossing, to support semantic interoperability between heterogeneous inter- as well as cross-company levels.

1 INTRODUCTION

Current markets are operating in turbulent and dynamic environments being influenced with permanent requirements for higher quality and lower price of the products and services. Such circumstances as well as rapid technological achievements force manufacturing companies to emerge in a new way of organizational and production paradigms, such as virtual enterprises (Kulvatunyou et al. 2005). A virtual enterprise (VE) is seen as an integrated network of regular companies that join their core services and resources in order to respond to unexpected business opportunities collaborating on an ad hoc basis. Such a network includes also suppliers, distributors, retailers and consumers requiring from involved companies to gather and share data and information about markets, customers and internal competences (Aerts et al. 2002). The potential benefits from virtual organizations are: agility reflected through a fast reaction on the unpredictable changes in the environment, utilization of synergies between companies that can improve business opportunities and gain new markets, reaching a critical mass and appearing in the market with a larger "visible" size, improved competitiveness and resource optimization as well as innovation potential (Camarinha-Matos, 2002). The capability of companies to form virtual enterprises and cooperate with partners is an important factor for keeping a competitive position on the market.

Nevertheless, the presented advantages alone are not enough to ensure the widespread adoption of the VE concept, which is still missing. Especially "Small & medium enterprises" (SME) are still missing an adequate approach to co-operative manufacturing (Ktenidis and Paraskevopoulos, 1999). A new approach for virtual enterprise modeling as well as the fulfilment and consideration of several research challenges, such as improved knowledge exchanging and sharing, fast reaction to customer demand, re-organization capability, and integration of heterogeneous entities, are required (Roche et al. 1998). The introduction of tools, techniques and methodologies that will support interoperability, information search and selection, contract bidding and negotiation, process management and monitoring, etc., is also highly

Koppensteiner G., Merdan M., Lepuschitz W., List E. and Vittori L. (2009).
ONTOLOGY-ORIENTED FRAMEWORK FOR VIRTUAL ENTERPRISES - Accomplished within the Project: Future Network-based Semantic Technologies (FUNSET-Science).
In *Proceedings of the International Conference on Knowledge Engineering and Ontology Development*, pages 300-307
DOI: 10.5220/0002311103000307

required (Camarinha-Matos, 2002). In this context, the information and knowledge exchange between partners plays a critical role for the success of such networks. This particularly due to the extreme heterogeneity of the VE environment, in which it is usually not transparent to the partners, which knowledge is available at whose partner's site or even if so, then in most cases the knowledge is not understandable due to the usage of different formats and tools. It is of biggest importance to have an optimized information flow to find the appropriate knowledge source in the desired quality within an adequate time. The information search and representation are seen as the two biggest challenges for the information technology (Stuckenschmidt and Harmelen, 2005).

Ontologies have been developed and investigated for quite a while in artificial intelligence and natural language processing to facilitate knowledge sharing and reuse (Kulvatunyou et al. 2005). They are of vital importance for enabling knowledge interoperations between partners and, at the same time, a fluent flow of different data from diverse domains. Ontologies allow the explicit specification of a domain of discourse, increasing the level of specification of knowledge by incorporating semantics into the data, and promote its exchange in an explicitly understandable form (Silva and Rocha, 2003). Semantic means in this context that all relevant concepts important for partners will be modeled in an ontology by capturing the associations between the domains ensuring at the same time the understanding of exchanged knowledge during the inter- as well as inside-company communication. This allows business partners to build open communities that define and share the semantics of the information exchanged in their domain.

Furthermore, the distributed nature of the VE sets requirements related to the supervision, coordination and execution of local (company intern) goals as well as global (VE) goals. The challenge is to introduce technologies that can support understanding as well as automation, and control processes connected with the creation, operation, and dissolution of VEs (Marik and McFarlane, 2005). Moreover, the companies are internally confronted with permanent requirements to optimize the workflow and improve effectiveness as well as efficiency. The currently mostly applied centralized control structures respond weakly to frequently changing customer demands in terms of performing necessary changes in the manufacturing environment itself due to their rigid character and limited adaptation capabilities (Parunak, 1996; Shen and Norrie, 1999). Making the control of the system decentralized, intelligent agents offer a convenient way of modeling processes and systems that are distributed over space and time, thereby reducing the complexity, increasing flexibility and enhancing fault tolerance (Jennings and Bussmann, 2003).

Our approach is to use semantic technology together with software agents in order to improve knowledge capturing, knowledge reuse and knowledge transfer as well as to answer the shortcomings mentioned above. The software agents are used, on the one side within companies to control certain components and processes (domains) and on the other side to establish the link to other partners within the VE. In this paper we use three diverse SMEs as test cases, representing their basic concept in ontologies and supporting their internal control with related multi-agent architectures to demonstrate a concept which offers the directions towards solving the interoperability problems within the VE. The suggested ontology-based communication and coordination between the agents enables also companies to improve and adjust their internal processes (Merdan, 2009).

2 ONTOLOGY

Knowledge and information sharing within a company (product design, process planning, and scheduling, supply, intern transportation, inspection, handling, etc.) as well as with other companies (selling, cooperation, servicing, etc.) is from crucial importance for the company's survival. Their representation needs to go beyond heterogeneous formats to enable exchange across the intranets and extranets as well as between various enterprise applications.

Encoding the meanings separate from the data, content and applications, and integrating them via a shared ontology, semantic technology enables their easier sharing and managing. An ontology is defined as an explicit specification of conceptualization (Gruber, 1993), with conceptualization meaning the shared view of environment representation. The ontologies and embedded semantics can be used to formalize the knowledge representation and to achieve overall "understanding". Nevertheless, the lack of common ontologies among the cooperating organizations (Camarinha-Matos, 2002) is seen as a serious limitation to tap the full potential of the VE concept. Common ontologies allow an easier integration of the underlined domain concepts, thus enabling the effective share of information between heterogonous environments. Such ontologies can specify and address the related concepts and classes when two different ontologies have to be merged or part of one mapped to other ones. However, due to the complex and dynamic nature of the VE it is hard to capture all related concepts in a persistent ontology. It is much simpler to isolate the ontology part that is only related to data exchange and communication processes associated with it. In this context it is necessary to define an ontology that will detail the representation and semantics of data about negotiation and present the link to all other concepts in the company. This negotiation ontology will include a description of basic company internal concepts (order, user, product/service, interfaces, etc.).

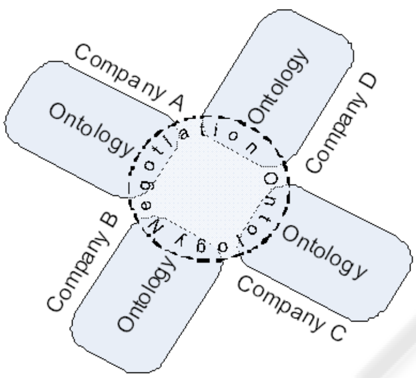

Figure 1: Negotiation Ontology.

Adhering to the reasons above this ontology should be accepted and implemented by all partners within the VE in the same form. We specified an order as a major concept that links diverse partners between each other as well as the VE with other companies. The order, labeled with the product/service type (though this is related to the product/service design/description), deadline and quantity, sets the key borders to the production planning process impacting directly the resource exploitation. This is the main reason why the product/service order parameters should be "understandable" and presented in the whole production chain, from the order over production until the final delivery.

As stated in the introduction, the VE concept is supported by a related multi-agent architecture. In our previous work we developed a knowledge-based multi agent architecture applied in the assembly domain (Merdan, 2009). Although implemented in the assembly domain, due to its generic nature this architecture can cover any other manufacturing domains. In this architecture, an agent is defined as an autonomous semantic entity that has specific tasks and knowledge about its domain of application, about strategies that can be used to achieve a specific goal, and about (other) relevant agents involved in the system. This architecture consists of four major agent classes. The Contact Agent (CA) has responsibilities that encompass organizational and system supervisory functions. The Order Agent is responsible for the accomplishment of one order, related process planning respecting due dates and the like, and handling customer requests for modifying or cancelling their orders. The Supply Agent is in charge of coordinating the production execution in order to achieve the best possible production results, including on-time delivery, cost minimization, etc. The Machine Agents represent manufacturing resources (typically a machine) providing particular processes and services. In this paper we extended this architecture by adding the NegotiationAgent and by assigning also the negotiation administration role to the CA. In the next sections, we will present defined ontologies for three diverse manufacturing domains as well as the correlation of these ontologies to the associated negotiation ontology. Our architecture is based on agents that have a rule-based behavior. Rules are considered as if-then statements applied to the agent's knowledge base.

3 USE CASES

In order to present our concept, we selected four different companies with related products and services. The major aspect here is that associated ontologies cover different concepts and workflows. Moreover, while such companies can be placed anywhere, the possibility exists that they use different words for the description of the same concepts or vice versa.

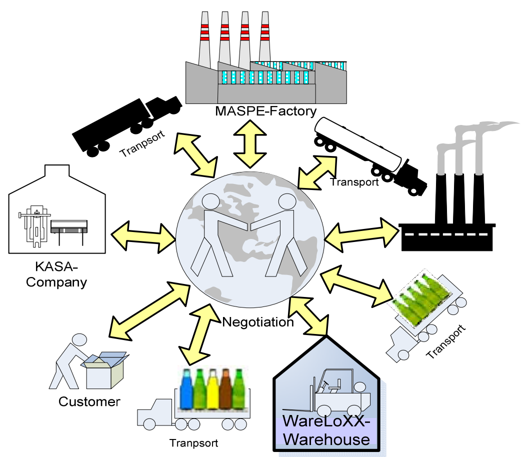

Figure 2: VE Concept.

Following Use-Cases were separately analysed and developed: from four project groups:

- KASA-Ontology: To represent a company for agent based assembly automation,
- MASPE-Ontology: An agent based batch processing factory for liquids,
- WareLoXX-Ontology: A warehouse system for the commitment of orders,
- KABA-Ontology: A bottling plant for the filling of bottles combined with LiStoSys-Ontology.

However, the usage of the negotiation ontology that links them enables the determination of an equivalent or the semantically closest concepts. Specified companies need to cooperate and negotiate with each other to be able to place products on the market. The negotiation ontology is used to ensure the overall understanding during communication and to enable the mapping of external information and knowledge into an internal company representation.

3.1 KASA-Ontology

In the KASA ontology (Merdan, 2009), a company from the assembly domain is used to offer particular product. In related ontology *Product Order* defines the type of the requested product, its quantity, design e.g. color, etc. The ontology based product model is used to extract the production/assembly operations from the product design and link particular *Steps*, which have to be performed for the production/assembly of a product, to particular resources (Figure 3).

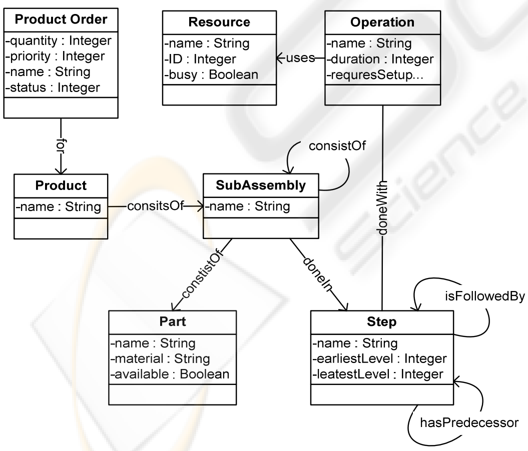

Figure 3: Assembly Ontology.

A *Product* is presented as a hierarchy of subassemblies and parts together with all their properties and relationship between them. *Parts* are defined as components, described by a set of attributes, properties, constraints and relations to other parts. The relationship between parts within a subassembly defines operations that have to be done to connect these parts and represents how these subassemblies should be put together to complete the product. A *Resource* is a physical component able to perform a certain *Operation*.

3.2 MASPE-Ontology

Concerning process automation, in particular batch automation, modifiable recipes provide a certain grade of flexibility at least from a process-oriented kind of view. However, the underlying control systems are still based on centralized structures that impede easy modifications of the system. This affects intended modifications, such as an extension of the system functionality by further components due to changing market demands, as well as unintended modifications in the case of occurred failures. Multipurpose facilities that provide reconfigureability of software and hardware components are therefore required (Kuikka, 1999; Sünder et al. 2006). Agent-technology brings certain advantages to the domain of batch automation as it provides means for the dynamic allocation of resources (such as reactor tanks), path optimization (as for instance pipes need to be cleaned before another product may be transferred) and material tracking.

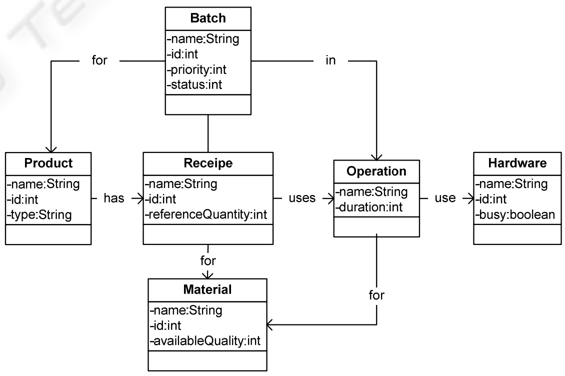

Figure 4: Batch Ontology.

MASPE (Multi Agent Systems for process engineering) represents an approach to integrate agent-technology into a batch control system. The FESTO compact unit, a laboratory process plant, acts as the first target system for this approach. The MASPE ontology (Figure 4) comprises the environment of this process plant, with all essential information, which are needed by the plant to work on its own.

The ontology incorporates a set of classes that describe the process of manufacturing a product. The concepts of certain classes, such as the concept of a batch or a recipe, are derived from the relevant standard IEC 61512 *Batch Control – Part 1: Models and terminology* (IEC, 1997). The class Product serves as a unique naming class for a product (i.e. for instance a specific amount of a pharmaceutical product) by using an ID and refers to the class *Recipe*, which contains all required material resources (such as raw material) and operations to manufacture this product step by step. One recipe can only describe one certain product and backwards – one product can only be described by one recipe. The actual execution of a recipe delivers one batch of a type of product. Hence, the class *Batch* refers to one product and one recipe. Recipes refer to one or several *Operation* classes and require one or several types of material to be executed. Operations (e.g. heating up material to a specific temperature to generate a reaction) are performed during the execution of a recipe on a batch and require at least one type of material as well as at least one type of hardware (e.g. a heater of a reactor tank). Hence, the class Operation refers to one or several *Material* as well as *Hardware* classes.

3.3 WareLoXX-Ontology

The efficiency and effectiveness in any distribution network is significantly influenced by the operation of the nodes in such a network, i.e. the warehouses (Rouwenhorst et al., 2000). Warehousing involves all movements of goods within warehouses namely: receiving, storage, order-picking, accumulation, as well as sorting and shipping (Van den Berg, 1999).

In opposition to conventional warehouse systems, our concept combines software agent technology and an ontology-based model to map a warehouse system and support automated warehousing. Therefore the following simplified ontology-concept was created (Figure 5):

First there an *Order* is issued which requests one or more *Crates*. The crates have to be requested in the reverse delivery order. A crate consists of a *Ware*. The crates are stored in a *Stock*, which is separated in multiple storage positions. Every storage position has an x-coordinate and a y-coordinate. Every storage position hosts multiple segments, which can contain only one crate. Every crate has a unique segment-coordinate. The crates are moved by a *ConveyorSystem* to the *PalletMachine*. When there are no more requests for crates, the crates are placed on a *Pallet*. If a pallet is full and more crates are required for one order, another pallet is provided. The pallets must arrive at the *Goodissue* (the place where they are loaded on a truck) in the right delivery order.

Independently from this the system ontologies offer services to external project partners like the storage of goods, the transport from the supplier to the customer and therefore the warehousing of diverse goods within the logistic chain.

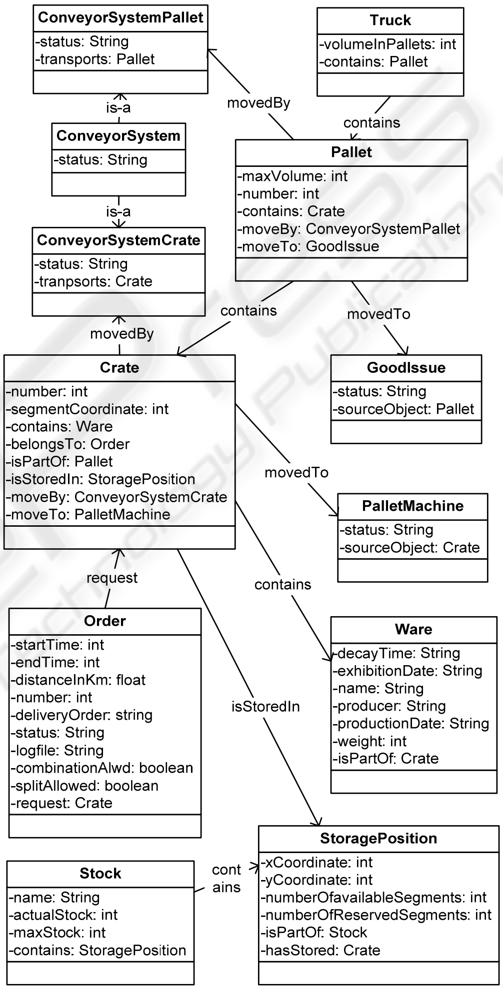

Figure 5: WareLoXX Ontology.

3.4 KABA-Ontology

The KABA ontology (Figure 6) is meant to support the information and knowledge exchange in a bottle filling plant.

Such company has to manage different kinds of bottles, crates, liquids, machines, conveyors and their disturbances. In our previous work we presented an ontology-based approach that improves flexibility and enhance fault tolerance of the transportation system (Merdan et al., 2008;

Koppensteiner et al., 2008), which is seen as backbone of such a plant.

The class *Job* is major class in the KABA-Ontology and is aimed to summarize all operations done within one particular *Order*. The class *Operations* contains subclasses which represent all available machine functions. The *Resources* class has two subclasses named *Equipment* and *Package*. Equipment is represented as entity able to perform a certain operation. *Conveyors* are used as buffers, which capacity is defined with their dimension and speed. On another side, the class Package has a *Pallet*, *Crate* and *Bottle* subclasses, which are means to encapsulate particular item within a order.

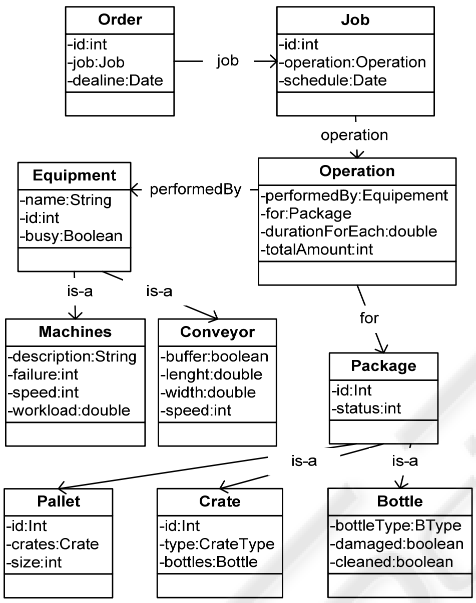

Figure 6: KABA Ontology.

4 NEGOTIATION ONTOLOGY

Negotiation can be understood as the process of reaching agreement on one or more matters of common interest. Traditional negotiation approaches have several constraints on the type of interactions (only pre-determined protocols are allowed or agents identified) and have protocols, which are coded implicitly within agents and hard to modify (Tamma et al, 2005). The negotiation ontology (Figure 7) acts as a general framework that defines the basic terminology, interaction and protocols enabling agent to reach agreement (Tamma et al, 2002). The common purpose of our ontology is a support of different negotiation types with multiple users at the same time in a VE. Besides the ability to support different auction types, auction properties and negotiation tactics, it can also handle different products/services, their properties and users. Our ontology has its roots in (Vetter, 2006).

It enables that every user has the possibility to start its own negotiation with an individual configurable Negotiation Agent (NA) which can then handle multiple negotiations. Firstly, the user specifies a *Good* that he wants to get or offer and over the Contact Agent (CA), if any kind of negotiation is required, creates the NA loading to it *Auctioneer* behaviors. This Auctioneer will lead the auction itself.

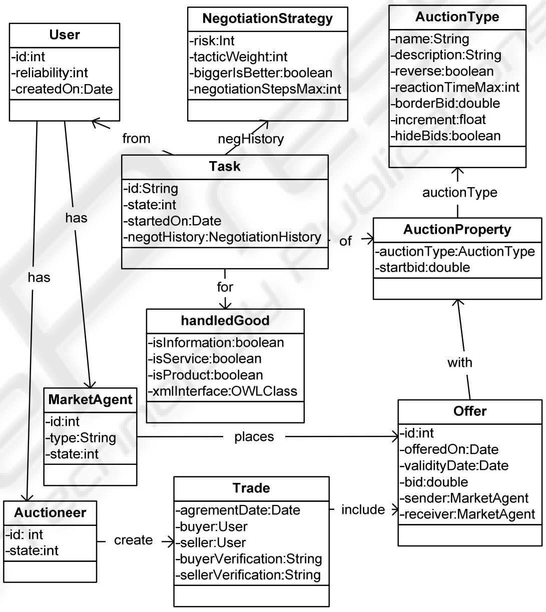

Figure 7: Negotiation Ontology.

Every new user who joins the auction has a configured *MarketAgent* that has participant role and related behaviors. These MarketAgents are sending offers to the Auctioneer. The Auctioneer takes these bids and compares them. Then it sends messages back to all MarketAgents. These messages contain information about the state of the auction (highest bid). Now the MarketAgents know if they want place another bid or to left the auction. When the time is up (or a maximum of negotiation steps is reached), the Auctioneer takes the best bid and creates a Trade object. This Trade object contains all information about the seller, the buyer and the auction itself. In the case that requested service or information doesn't require auction, the CA map these, using the related parts of negotiation ontology concept, in internal company ontology on its own and starts appropriate behaviours.

5 SYSTEM IMPLEMENTATION

As mentioned in the previous sections, the concept is based on distributed multi agent architecture, which is currently implemented at Automation and Control Institute. As Framework for the agent implementation JADE (JADE, 2009) is used. The Java Agent Development Environment provides Platforms for each Company and the ACL-Message System to exchange Messages according to the FIPA-Standard (FIPA, 2009). To connect the different platform, the already implemented and tested DF-Federation for Ontology-based resource allocation (Koppensteiner, 2008b) is used. To provide the JADE-agents with an ontology the system architecture is based on (Merdan, 2009). The different company ontologies, to handle the internal representation of company products and tasks, where therefore modelled in the ontology design tool Protégé (Protégé, 2009). All these agents within the VE are also equipped with an negotiation ontology to share a common understanding.

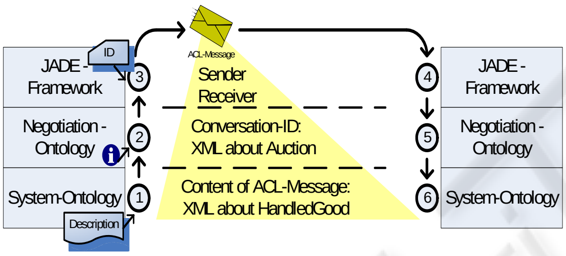

Figure 8: System Implementation.

As the Figure 8 shows, the proposed systems could be divided into three layers. On the bottom is the company's system ontology. The next layer combines this system ontology with the proposed negotiation ontology and the final layer is the JADE-Framework itself. To show the idea behind this architecture, the example of a message exchange is explained. It doesn't have any relevance, whether the message is to start an auction, place a bid or request information, the procedure is each time the same.

(1) If an agent has something to communicate, it derives a description of its demand from its own system ontology and generates an XML representation of it. This XML is stored later in the content field of the ACL-Message.

(2) Related to the type of the communication act the conversation-id of the ACL-Message is created from the negotiation ontology and formatted also in XML. In case of an auction the whole auction - information is captured in this representation.

(3) Afterwards, the message is equipped with the sender information, the receivers address and all other necessary information according to the FIPA standard. In case that the agent doesn't have any information about possible agents which offer particular service the DF-Agent of the JADE-Platform is used to find all reliable agents within the DF-Federation

(4) The receiver gets the message over the JADE-runtime and starts its behavior that maps the message to its ontology.

(5) After an agent has mapped the message, it checks the conversation-id to assign the message to the right context of its negotiation behavior.

(6) Finally, it extracts the information given in the content field of the message and stores it in his knowledge base. Consequently, it acts based on the new circumstances, e.g. place a bid or request more information.

6 CONCLUSIONS

The virtual enterprises paradigm is seen as promising approach that can help companies to face the current dynamic market trends and conditions. However, the limited knowledge and information exchange between involved partners is significant drawback that prevents their wider establishment. We present the ontology-based concept combined with multi-agent approach that enables easier flow of information and knowledge. For the development and implementation of the concept we used four uses cases that are connected through overlapping (negotiation) ontology. We proved the feasibility of the presented approach. Having multi-agent architecture as a basis of our approach, our future work will be concerned with further development and tuning of defined agent behaviors. Furthermore, another part of our research is going to focus on ontology merging and mapping, which is complementary to our concept.

ACKNOWLEDGEMENTS

The authors would like to acknowledge the financial support by the Sparkling Science program, an initiative of the Austrian Federal Ministry of Science and Research. We also want to thank all students involved in the FUNSET Science Project at Vienna Institute of Technology (TGM), department for Information-Technology – especially the students Reinhard Grabler (WareLoXX), Michael Peitl (KABA), Simon Strobl (Negotiation), Nguyen Than Tung (LiStoSys) and Mirza Zenkic (MASPE) for their contribution in ontologies development.

REFERENCES

Aerts A.T.M., Szirbik N.B., and Goossenaerts J.B.M., “A flexible, agent-based ICT architecture for virtual enterprises,” *Computers in Industry*, vol. 49, Dec. 2002, pp. 311-327.

Camarinha-Matos L.M., “Virtual organizations in manufacturing: trends and challenges,” *Proceedings of the 12th International Conference on Flexible Automation and Intelligent Manufacturing*, Dresden, Germany, 2002.

FIPA, Foundation for intelligent physical agents, Communicative Act Library Spec., http://www.fipa.org/specs/fipa00037 and ACL Message structure Specification, http://www.fipa.org/specs/fipa00061, accessed Mai 2009

Gruber T.R., “A translation Approach to Portable Ontology specifications,” *Knowl. Acquis*., vol. 5, 1993, pp. 199-220.

IEC TC65, IEC 61512-1: Batch Control – Part 1: Models and terminology, International Electrotechnical Comission (IEC), 1997.

JADE – Java Agent Development Framework, http://jade.tilab.com/, accesed April 2009

Jennings N. and Bussmann S., “Agent-based control systems: Why are they suited to engineering complex systems?” *Control Systems Magazine, IEEE*, vol. 23, 2003, pp. 61-73.

Koppensteiner G., Merdan M., Hegny I., and Weidenhausen G., “A change-direction-algorithm for distributed multi-agent transport systems,” *Mechatronics and Automation, 2008. ICMA 2008. IEEE International Conference on*, 2008, pp. 1030-1034.

Koppensteiner G., Merdan M., Zoitl A., and Favre-Bulle B., “Ontology-based resource allocation in distributed systems using director facilitator agents,” *Industrial Electronics, 2008. ISIE 2008. IEEE International Symposium on*, 2008b, pp. 1721-1726.

Ktenidis P. and Paraskevopoulos A., “Information and communication technology (ICT) infrastructure for small and medium enterprises (SMEs) networks: achievements and work-in-progress from three European projects,” *Research Issues on Data Engineering: Information Technology for Virtual Enterprises, 1999. RIDE-VE '99. Proceedings*., Ninth International Workshop on, 1999, pp. 44-51.

Kuikka S., A batch process management framework: Domain-specific, design pattern and software component based approach, PhD thesis. Helsinki University of Technology, 1999.

Kulvatunyou B., Cho H., and Son Y.J., “A semantic web service framework to support intelligent distributed manufacturing,” *Int. J. Know.-Based Intell. Eng. Syst*., vol. 9, 2005, pp. 107-127.

Marik V. and McFarlane D., “Industrial Adoption of Agent-Based Technologies,” *IEEE Intelligent Systems*, vol. 20, 2005, pp. 27-35.

Merdan M., “Knowledge-based Multi-Agent Architecture Applied in the Assembly Domain,” PhD Thesis, Vienna University of Technology, 2009.

Merdan M., Koppensteiner G., Hegny I., and Favre-Bulle B., “Application of an ontology in a transport domain,” *IEEE International Conference on Industrial Technology, 2008*. ICIT 2008, 2008, pp. 1-6.

Parunak H.V.D., “Applications of distributed artificial intelligence in industry,” *Foundations of distributed artificial intelligence*, John Wiley \& Sons, Inc., 1996, pp. 139-164.

Protégé, Stanford Medical Informatics, Stanford University, http://protege.stanford.com, accesed Mai 2009

Roche C., Fitouri S., Glardon R., and Pouly M., “The potential of multi-agent systems in virtual manufacturing enterprises,” *Database and Expert Systems Applications, 1998. Proceedings*. Ninth International Workshop on, 1998, pp. 913-918.

Rouwenhorst B., Reuter B., Stockrahm V., van Houtum G.J., Mantel R.J., and Zijm W.H.M., “Warehouse design and control: Framework and literature review,” *European Journal of Operational Research*, vol. 122, May. 2000, pp. 515-533.

Shen W. and Norrie D. H., “Agent-based Systems for Intelligent Manufacturing : A State of-the-Art Survey,” *Knowledge and Information Systems, an International Journal*, vol. 1(2), 1999, pp. 129-156.

Silva N. and Rocha J., “Ontology mapping for interoperability in semantic web,” *Proceedings of the IADIS International Conference WWW/Internet*, 2003.

Stuckenschmidt H. and Van Harmelen F., *Information sharing on the semantic web*, Springer, 2005.

Sünder C., Zoitl A., and Dutzler C., “Functional structure-based modelling of automation systems,” in *Int. J. Manufacturing Research*, vol. 1, iss. 4, pp. 405-420, 2006.

Tamma V., Phelps S., Dickinson I., and Wooldridge M., “Ontologies for supporting negotiation in e-commerce,” *Engineering Applications of Artificial Intelligence*, vol. 18, Mar. 2005, pp. 223-236.

Tamma V., Wooldridge M., and Dickinson I., “An ontology for automated negotiation,” *Proceedings of the Workshop on Ontologies in Agent Systems*, 2002.

Van den Berg J.P., “A literature survey on planning and control of warehousing systems,” *IIE Transactions*, vol. 31, 1999, pp. 751-762.

Vetter M., “Ein Multiagentensystem zur Verhandlungsautomatisierung in elektronischen Märkten,” PhD Thesis, Stuttgart University , 2006.