Proceedings of the 17th World Congress
The International Federation of Automatic Control
Seoul, Korea, July 6-11, 2008

Semantics in Industrial Distributed Systems

Marek Obitko[*,], Pavel Vrba[*], Vladimír Mařík[*,**], Miloslav Radakovič[*,**]**

**Rockwell Automation Research Center, Prague, Czech Republic*
***Department of Cybernetics, Czech Technical University in Prague, Czech Republic*
{mobitko,pvrba,vmarik,mradakovic}@ra.rockwell.com

Abstract: Industrial distributed systems aim at robust and flexible control of industrial processes, for which the traditional centralized approaches are not sufficient. The general problem of distributed systems is that they are still tightly coupled from the point of view of system integration and are still far from openness that would enable cooperation at a larger scale without any human intervention. To achieve better operation and integration in an open reconfigurable environment, explicit semantics is needed to capture the meaning during communication. Relevant research area is the field of ontologies and semantic web. We show how semantics can be employed in industrial systems, in particular in distributed agent-based systems, and especially using semantic web research. We review current state of the art and, based on our own experience, we discuss potentials and challenges as well as differences and similarities of applications of semantics and ontologies in industrial systems when compared to WWW oriented research.

1. INTRODUCTION

Industrial systems need to be distributed in order to achieve robust and flexible control of complex processes. In many scenarios, traditional centralized approaches to control, planning, and scheduling or supply chain management are not sufficient to cope with higher requirements on complexity and robustness.

The general problem of highly distributed systems, such as for example internet, service oriented architectures (SOA) or multi-agents systems (MAS), is that they are still more tightly than loosely coupled from the point of view of automated gathering and integration of data, information and knowledge. Although the software and hardware interface standards enable unforeseen coupling during runtime, such systems – especially in industrial domain – are still far from being open. The openness would enable cooperation at a larger scale without any human intervention. The reason is that the systems themselves are programmed with the focus on performing particular tasks rather than on interoperability. This is understandable, but in order to achieve better integration from the shop floor level up to the level of virtual enterprises, these systems have to use explicit semantics to describe the interpretation of the data they provide.

In this paper, we compare the internet and World Wide Web (WWW) technologies with the relevant information technologies used in the industrial manufacturing systems. We pay special attention to the semantic web research and its application to industrial distributed systems and show how semantics can be applied to such systems. We also focus on the multi-agent and holonic techniques that provide suitable paradigm for such systems as well as clear modeling framework for introducing semantics.

The paper is organized as follows: we briefly review the research in semantics, ontologies and semantic web. Then we show how distributed industrial systems are influenced by internet and WWW technologies. Benefits of using semantics in industrial systems are discussed. After that, current work in using the semantics is described, first in holonic and agent-based distributed industrial systems and then in general approaches applicable for the industrial domain. We present features common to manufacturing systems with explicit semantics. We conclude with the discussion of differences and similarities of the usage of the semantics in industrial domain versus other relevant research, such as semantic web.

2. SEMANTICS AND ONTOLOGIES

To allow communication among heterogeneous systems, there must be an agreement on both the syntax and the semantics of the exchanged messages or of the other type of interaction. The syntax defines the structure of a language, i.e., a grammar typically in a form of rules that govern the structure of sentences. Semantics refers to the aspects of meaning as expressed in the language, i.e., the sense of language elements and their combination, including the relation of these elements to the real world.

The semantics is often captured by an ontology. The term ontology as clarified in (Guarino and Giaretta, 1995) comes from philosophy, where it refers to the study of being or existence, and attempts to describe categories and relationships of all existing things. In engineering applications, this is reduced to model a part of a selected domain – a model that is processable by a machine and appropriate for a specific application. The ontology in this context is often defined as a formal explicit specification of a shared conceptualization. A conceptualization is a shared view of how to represent the world. Ontology is then formal

978-1-1234-7890-2/08/$20.00 © 2008 IFAC **10.3182/20080706-5-KR-1001.3182**

description of this view and as a software engineering artifact it usually consists of concepts, relations and restrictions. The knowledge base containing the particular instances of ontology concepts then captures the state of a selected part of reality. For example, material handling ontology says there are conveyor belts and diverters that can be connected together and a knowledge base describing particular plant captures state of affairs, such as that the diverter A is connected to the conveyor belt B.

2.1 Semantic Web

Semantic Web aims to provide a common framework that would allow data to be shared and reused. It is an extension of the World Wide Web in which the content is described in a form suitable for software agents, so that they would be able to find, integrate and process information more precisely. For this purpose, machine processable semantics is needed. The core semantic web technologies are Resource Description Format (RDF) and Web Ontology Language (OWL). Although these technologies were primarily designed for the web, they are suitable for other applications as well.

The RDF is a standard for expressing structured data in a form of simple statements (Manola and Miller, 2004). Each statement is expressed as a triple subject-predicate-object and each participant of the triple is a web resource (identified by URI). In the place of object, a literal such as string or number can be used. The triples, which can be gathered from distributed sources on the web, are linked together to a searchable graph. For searching within an RDF graph, the Simple Protocol and RDF Query Language (SPARQL) is available to specify required graph pattern for a query.

For vocabulary definition of RDF statements, the Web Ontology Language (OWL) was created (Dean and Schreiber, 2004). OWL serves as a standardized means for describing ontologies, and is in fact description logic with the syntax of RDF. Description logics describe concepts and roles (called classes and properties in the semantic web). A description logic system is composed of a T-box (terminology) and an A-box (assertions). The T-box describes ontology, i.e., knowledge that is not changing, while the A-box describes knowledge base, i.e., knowledge that is continually updated. OWL is based on the logic that allows sound and complete reasoning in practically usable time. OWL ontology describes taxonomy of classes and properties together with constraints of the use of these classes and properties (T-box). In this way, OWL forms a vocabulary describing the world state (A-box).

2.2 Semantic Web Services

Web Services are in fact Web Application programming interface (API) that can be accessed over the internet. Web service specifications are standards for syntactic interoperability and include protocols for calling a service and returning the result, for describing a service interface, and for publishing and discovering metadata about services.

Semantic Web Services are semantic enrichments of the Web Services, like Semantic Web is an enrichment of the web. The semantic enrichment has a goal of allowing automated discovery, composition and execution of services. Special ontologies were developed for this purpose: OWL-S (The OWL Services Coalition, 2004) and WSMO (Lausen *et al.*, 2005). For example, OWL-S at its top level describes a service using ServiceProfile (what the service does), ServiceModel (how it works) and ServiceGrounding (how to access it). Such a description is necessary for influencing the broader SOA vision to allow truly open architecture that would be able to integrate various new services. In the internet environment such a vision is often referred to as Internet of Services (Schroth and Janner, 2007).

3. INTERNET AND WWW IN CONTROL SYSTEMS

Any larger industrial control system consists of distributed components. The internet and WWW technologies influenced the industrial domain in many ways.

At the level of network protocols, technologies like DeviceNet suitable for device-level communication or ControlNet ensuring deterministic communication have been used for communication. Although these networks have still their advantages for these particular applications, EtherNet/IP standard that is built on the standard Internet TCP/IP protocol is becoming popular for integrating systems together. However, there are still concerns for specialized applications, where, for example, determinism is required.

The communication over Ethernet is suitable for direct integration within Supervisory Control And Data Acquisition (SCADA) systems. An important part of SCADA is a Human-Machine Interface (HMI) that provides appropriate visualization for humans and allows manual control of a system. The Rockwell Automation’s package RSView Enterprise Supervisory Edition (SE) is a distributed HMI that supports multi-server and multi-client applications. The architecture of this package was inspired by WWW – there are servers that store HMI project components such as graphic displays and clients that access and show the displays. Each HMI project is identified in a form resembling URL that is identifying WWW resources.

The RSView SE Client is custom software, however, it is also possible to access HMI via standard web browser, for example with the RSView32 WebServer that provides a look into graphic displays, tags and alarms. For diagnostics at the lower level, a web server is running in a standard EtherNet/IP module and provides access to a chassis with controllers. This web server provides simple diagnostics of the network and the modules in chassis. If there is a need for detailed diagnostics, there is a specialized module that enables users to view plant-floor data via web browser, receive email alarm notifications, and transmit information to other applications using XML data, without the need for special HMI software.

As we can see, the internet and WWW technologies are important in the larger control systems particularly when connecting them with human user interfaces or with systems ensuring higher level integration, such as Manufacturing

Execution Systems (MES) or Enterprise Resource Planning (ERP) systems. These technologies are also influencing the architecture of the industrial systems. The semantic web research has considerable impact on the manufacturing domain as well, as we show in the rest of this paper.

4. SEMANTICS IN INDUSTRIAL SYSTEMS

In this section we discuss the potentials of using semantics in industrial distributed systems with special attention paid to the semantic search issues and the semantic enrichment of holonic and multi-agent industrial control systems. The obvious advantage of using ontologies is that the assumptions about the manufacturing domain become explicit. This facilitates communication in a distributed system because the communication vocabulary becomes unambiguous. Also, the formal ontology allows reasoning over acquired knowledge.

4.1 Semantic Search

One of the core applications of the semantic web is a semantic search, i.e., search within semantically enriched data. The design, operation and maintenance of a manufacturing system is very knowledge intensive task and involves handling of information stored in different forms – for example function blocks or ladder diagrams describing the real-time control system, SCADA/HMI views, collected historical data, etc.

It is often not easy to search within such information space even using plain text search. However, when the information is accessible in the semantic web form, it is possible to make queries beyond the classical keyword search. We have investigated the use of semantic web technologies for the semantic search within various information sources of an assembly line. Our conclusion is that the RDF/OWL form of data is appropriate for storing the information and for querying (Obitko, 2007). The SPARQL language is capable to express structural queries that are of practical interest. Example of such a query is to find projects where specified ladder code instructions with a specified variable occur in one rung. In the prototype prepared by Rockwell Automation the data extracted from the manufacturing system are annotated automatically depending on their context, so the process of indexing for structural queries is fully automatic.

In addition, the implicit information (such as the `part-of` relation) can be made explicit in the ontology describing the manufacturing system. The query engine can employ an OWL reasoner to include this information into query results, so that for example query results containing the `part-of` relation contain transitive closure of this relation. An important advantage of using RDF is that all the distributed information can be merged into single RDF graph like on the internet. This allows asking for connections of information from different sources, such as in the query to find HMI views that have pushbuttons connected to a ladder code project that is used in a specified area.

4.2 Semantics Utilization for Reasoning

Let us illustrate the usage of reasoning in an ontology expressed using OWL on an example from a transportation domain (Obitko and Mařík, 2003), where transportation devices are described using properties like `targetNode`, `defaultCost`, and `connectedTo`. The transportation edge properties – `targetNode` and `defaultCost` – are of type `owl:FunctionalProperty`, which means that there exists at most one value for that property. Let us suppose that there are several different kinds of identifications of a particular transportation node – one is by its purchase identification number from a commercial department and another one is the address used among transportation agents. When an agent gets information from two sources that these two transportation nodes are target nodes of the same transportation edge, it can derive from the functional property type that these two nodes are the same individuals that just have different identifications for different communities. To show another illustration, let us suppose that `targetNode` is `rdfs:subPropertyOf` of `connectedTo` and the `connectedTo` property is `owl:SymetricProperty`. Then, when it is known that a transportation node is a target node for a transportation edge, it can be derived that not only the node is connected to the edge, but also that the edge is connected to the node.

This kind of reasoning is available just by using the specified ontology. If this ontology is shared among agents, these agents are able to find the same consequences from the information that they exchange about the environment.

5. SEMANTICS IN HOLONIC AND AGENT-BASED INDUSTRIAL CONTROL SYSTEMS

Holonic and multi-agent systems have been widely recognized as enabling technologies for designing and implementing next-generation of distributed and intelligent industrial automation systems (Bussmann *et al.*, 2004). These systems are characterized by high complexity and requirements for dynamic reconfiguration capabilities to fulfill demands for mass customization, yet low-volume orders with reduced time-to-market. Self-diagnostics and robustness that allow efficient continuing in operation even if the part of the system is down are other important properties.

The trend of applications of multi-agent systems is apparent at all levels of the manufacturing business. At the lowest, real-time control level, so called holons or holonic agents are usually tightly linked with the real time control programs (implemented in IEC 61131-3 or IEC 61499 standards) through which they can directly observe and actuate the physical manufacturing equipment (Vrba, 2006). Intelligent agents are also used for production planning and scheduling tasks both on the workshop and factory levels. More generic visions of intensive cooperation among enterprises connected via communication networks have led to the ideas of virtual enterprises (Camarinha-Matos, 2002).

The common principles in industrial deployment of the agent technology is distribution of decision-making and control processes among a community of autonomously acting and mutually cooperating units – agents. At the shop floor level, for instance, an agent represents and independently controls particular physical equipment, like a CNC machine, conveyor belt or docking station. The substantial characteristic is the cooperation among the agents as they pursue either their individual goals or the common goals of the overall control system. The inter-agent interactions vary from simple information exchanges, for example about the state of processing as the product moves from one machine to another one, through requests to perform a particular operation, for example requesting an AGV to transport a product to a particular work station, to complex negotiations based on contract-net protocol or different auction mechanisms.

As the information representation and exchange is the essence of such systems, the need for explicitly defined and shared ontologies becomes apparent. The exploitation of semantics and ontologies in the area of agent-based industrial systems seems to be very intensive these days, which was not the case even a few years ago. The researches apparently realized that the syntactical interoperability, predominantly ensured by the XML-based messaging, will not be sufficient to keep the pace with the trend towards semantically interoperable knowledge based systems. Thus, the use of semantic web technologies has accelerated significantly in the agent research community over the past few years.

5.1 Domain-Specific Ontologies for Agent-Based Manufacturing Control Systems

The number of reports of deployments of ontologies in agent-based manufacturing systems increases. Usually, a domain-specific ontology covering a subset of the manufacturing area, for instance assembly, is developed and utilized for the purposes of the particular agent-based control application.

In (Cândido and Barata, 2007) the ontology for shop floor assembly is described. Two basic categories of concepts are proposed: *modules* and *skills*. Modules represent physical processing units or their aggregation: the *workcell* concept is defined as composition of *workstations* (typical `composedOf` relation is used), where workstation is a composition of *units*. The transforming unit (and its subclasses pick&place unit or milling machine), a flow unit and a verification unit are examples of units (`isA` relation is used to capture the inheritance). Skills represent abilities to perform manufacturing actions, as for instance *MoveLinear*, where complex skills are represented as a composition of basic skills. The manufacturing resource agent, when instantiated, searches the ontology for skills it supports using its serial number and type of equipment and then registers these skills in the Directory Facilitator (DF) agent, which maintains a list of services provided by individual agents. The coalition leader agent gathers coalition members' basic skills and searches the ontology to find out what complex skills could be supported by the coalition. If there are some found, those complex skills are also registered in the DF (with the description of how the coalitions are formed without identifying its members). Each skills description is supplemented with the message template for corresponding inter-agent negotiations. The proposed solution has been deployed in the NovaFlex shop floor environment installed at the Intelligent Robotic Center at UNINOVA.

An OWL-based ontology developed for agent-based reconfiguration purposes is reported in (Al-Safi and Vyatkin, 2007). The ontology application is illustrated on a small laboratory manufacturing environment consisting of two machines used for processing and handling work pieces equipped with different mechatronic devices such as rotating indexing table, plunger, drill, picker, etc. The basic ontology concepts, similarly to previous case just using different terms, are *material resource* and *operation*. The resources are *machine* and *tool* with corresponding subclasses like *handling machine* and *processing machine* as well as *rotary indexing table*, *drill*, *kicker*, etc. The operations are subdivided into *manufacturing operation* and *logistic operation* with further classification on *sorting*, *hole testing*, *drilling* and *picking*, *kicking* and *rotating* respectively. References between machine and operation concepts express the facts that machine *enables realization of* an operation and similarly. These general concepts from the ontology are then instantiated to capture the real environment, such as the particular machines and their relations. Such a dynamic part of the ontology is also expressed in OWL allowing the agents to reason about the available machines and operations, however still in the semantic context.

Magenta Technology company provides another example of exploration of ontologies in agent-based applications. In (Rzevski *et al.*, 2007) reporting on a set of multi-agent tools, the details of an Ontology Management Toolset is given. This tool enables developers to create and edit the static aspects of ontology as well as the dynamic aspects, here called *scenes*. The ontology developed by this toolset for supply chain and logistic planning is then presented, as also discussed in (Andreev *et al.*, 2007). The examples of concepts are for instance *factory*, *cross-dock*, *truck*, etc. and relations like *is booked for a demand*. Although it is not explicitly mentioned in the last two cited papers, the Magenta's multi-agent engine provides a mechanism of updating the agent's behavior (i.e., program code) dynamically as the ontology is being extended. The corresponding piece of code providing an agent with an algorithmical description of its behavior associated with a particular new ontology concept is sent to the agent so that it can subsequently execute the code to react appropriately.

The proposal of an ontology for organizational model of general holonic systems deployment is presented in (Cossentino *et al.*, 2007). The context of an organization is described in terms of *project management*, *manager*, *employee* and the roles such as *supervise* and *assigns*. Other general concepts for the agency and holonic domain are defined like *agent*, *agent role*, *holon*, *holon role*, etc.

5.2 General-Purpose Ontologies for Manufacturing Domain

We have documented that although there are many efforts towards designing ontologies for manufacturing domain, different researchers use slightly different vocabularies, describe their domains (often overlapping) from different view points and in majority of cases focus just on a specific problem domain. There are also activities towards generic ontologies for the manufacturing industry domain.

Very promising standardization effort seems to be concentrated around the OOONEIDA consortium focusing on creation of technological infrastructure for a new, open-knowledge economy for automation components and automated industrial products (Vyatkin *et al.*, 2005). The aim is to provide the framework for both the hardware and the software interoperability at all levels of the automation components market, i.e., from device vendors and machine vendors to system integrators up to the industrial enterprises. The set of searchable repositories of so called Automation Objects is envisioned, where each player deposits its encapsulated intellectual property along with appropriate semantic information to facilitate searching by intelligent repository agents. The use of semantic languages for knowledge repositories (mainly OWL) is promoted.

A complementary work to OOONEIDA initiative focuses on application of semantic web technologies to semantically enrich the description of automation objects (Lopez and Lastra, 2006). Two separate ontologies for mechatronic devices reference model (covering both the hardware and the software features) and the IEC 61499 reference model respectively are proposed and merged into an ontology for Automation Objects reference model (proposed by IEC-TC65 group). At the lowest level, the basic concepts of function blocks, events, I/Os, etc. are provided while at the device/machine level, concepts covering function block applications, resources, etc. are outlined. Two examples of Conveyor and Lifter automation object semantic description are sketched. As argued in (Lastra and Delamer, 2006), the semantic web services are generally suitable for rapidly reconfigurable factory automation systems.

A substantial standardization effort related to manufacturing is devoted by NIST. An example is the shop data model that is described in the form UML diagrams and XML serialization examples (McLean *et al.*, 2005). The model includes description of organization, bill of materials, process plans, resources, schedules, etc. Although it is not a formal ontology in the sense described earlier in this paper, such standards are important as a base for ontologies that would be widely accepted. Another example of NIST activities is the Process Specification Language PSL (Grüninger and Kopena, 2005) that is a logical theory that covers generic process representation common to manufacturing applications. The PSL ontology contains axioms grouped to theories describing aspects such as complex activities and can serve as a solid base or upper ontology for representing processes.

Another contribution in this area is the MASON ontology (MAnufacturing's Semantic ONtology) described in (Lemaignan *et al.*, 2006). The aim is to develop an upper ontology that would allow specific ontologies to be fluently integrated on the same common cognitive architecture. The OWL-based ontology describes the taxonomy of concepts such as entities, operations and resources and describes also properties such as relating tools to operations. For example, it defines *Milling* as a subclass of *Cutting*, or defines property *requiresTool* with the domain *ManufacturingOperation* and a range *Tool*. It is reported that currently the ontology, which is available on-line, constitutes of more than 270 base concepts and 50 properties. Moreover, a mapper has been developed between OWL ontologies and the internal ontology model used by the popular JADE agent platform. Although some of the constructs in the ontology seem to be application specific (for example restricting previous operation in the definition of operation concepts), this work can be seen as an important step towards formalizing the manufacturing domain.

When building the general-purpose manufacturing ontologies it is obviously necessary to have solid basis in form of well developed foundational (upper) ontologies. In that case, the ontology can employ for example spatial or time theories in a meaningful manner. Another advantage is that the integration with ontologies both inside and outside the manufacturing domain becomes easier. It is not easy to employ foundational ontologies since they are often created in very expressive languages without taking care of computability. However, it is always possible to export the result to a formalism that is computationally tractable. The formalization of ADACOR ontology (ADAptive holonic COntrol aRchitecture for distributed manufacturing systems) using the DOLCE methodology (Descriptive Ontology for Linguistic and Cognitive Engineering) is outlined in (Borgo and Leitão, 2004). ADACOR is originally described using UML diagrams and natural language descriptions, while DOLCE uses first order modal logic and aims at capturing the ontological categories underlying natural language and human commonsense, such as physical or abstract objects, events and qualities. The alignment of ADACOR to DOLCE yields well formalized and well founded ontology. The ontology described in ISO 15926 "Industrial automation systems and integration" also uses well founded principles of temporal and spatial representation of objects in a form of four dimensional approach to simplify reasoning in the process engineering domain (Batres *et al.*, 2005).

5.3 Interoperability of manufacturing ontologies

A widely accepted, consistent and comprehensive general-purpose upper level ontology for manufacturing domain and supplementary coherent set of standardized domain-specific ontologies is needed to ensure reusability and interoperability. In the manufacturing domain, standardization plays more important role than for a general web search, however, when describing the same thing from different point of views, such as the view of accounting department and of assembly line designer, there is a need to integrate these different views. In other words, the ontologies in open distributed systems are inherently different, and there is a need to exchange information in this environment.

Similar threats of ontology standardization are pointed out by (Lastra and Delamer, 2006). As illustrated in (Obitko and Mařík, 2005), this problem can be solved by translation of messages between ontologies – one agent prepares a message in its ontology, and the message is then translated into the ontology used by the receiving agent while preserving the meaning of the message. The details on the translation using semantic web technologies and OWL reasoning are described using transportation domain examples in (Obitko, 2007).

5.4 Common Properties of Ontologies Deployment in Agent-based Manufacturing Systems

Let us summarize and discuss the typical attributes of the applications of semantics in the agent-based distributed industrial control systems.

Static & dynamic aspects of semantics – as discussed earlier, OWL provides instruments to express static, invariant concepts or facts (T-box) as well as the dynamic model of the real world (A-box) created as particular instances of T-box terms. The *machine* provides an *operation* is the example of the former one while *machine M25* provides *drilling* is the example of the latter case. It is arguable how to call these two aspects – the former one could be simply referred to as *ontology* and the latter one as *knowledge base* or *scene*. While *ontology* is a semantic network of classes (concepts), their relations and attributes, the s*cene* constructed using concepts from ontology, represents the current situation in the corresponding part of the real world (Andreev *et al.*, 2007). While the ontology is designed a priori, the knowledge base is built by the agents dynamically as they perceive the real world by means of sensors, communication (sharing their knowledge bases) or possibly through user interaction (in case of combined human-machine systems).

Self awareness and localization – essential part of the agent's knowledge base is the information related to its localization in the real-world together with the perception of its presence and the results of actuation in the real world. Sensing, tracking and tracing technologies will play an increasingly important role in providing the real-world acting agents most accurate information about the surrounding environment. We can see a significant potential in the RFID (Radio Frequency Identification) technology as shown in a pilot application where the RFID technology is integrated with agents used for manufacturing control (Vrba *et al.*, 2006).

Reactivity – imagine a situation when an agent notes a particular event in the real world, for instance detects a failure of the controlled machine. It creates a corresponding fact consistent with the ontology (describing relation between a *machine* and *failure* concepts) and stores this information into its knowledge base. The agent's inference engine can then possibly deduce other new facts, but this still does not directly lead to reaction. But often, particularly in agents acting in real world, some action or reaction needs to be taken by the agent – for instance actuating (stop the drive) or informing other related agents. So the meaning of the particular concept from the ontology is not only knowledge-based but also "algorithm-based". The ontology should provide the agent also with the explicitly defined rules (or directly a program code) to be executed by the agent to react appropriately. Magenta agent runtime environment (Rzevski *et al.*, 2007) provides such feature – program code is sent to the agent to modify or extend its behavior. In the effort to deploy ontologies in the MAST system (Vrba, 2006), we have recently also implemented such a feature. Again, in the area of semantic web technologies, various rule languages are being developed. Using these languages allows direct exploitation of the relevant RDF-based tools and also easier interoperability outside of the manufacturing domain.

Ontology-based service matchmaking – one of the basic concepts of multi-agent systems is the registration of agents' skills and services in the Directory Facilitator which is then used by the agents to search for particular service providers. The information registered in DF is however in majority of agent platforms available today (JADE or Cougaar) in a very simple form. It usually contains just the type of the service (for instance *Drill*) but it cannot be further parameterized (*Diameter 10-100 mm*, *Hole depth 5-20 mm*). Obviously, to fully explore the potential of semantics in agent-based systems, ontologies must be deployed for service registration and lookup through DF as well. Within the registration the agent will send the corresponding part of its ontology (services it offers) to DF. DF can be then asked for finding particular service providers using more complex queries ("find all machines that can drill a hole of 50 mm diameter and 15 mm depth"). The result sent back by DF to the requester (also in form of ontology) might be with convenience supplemented by the message template and protocol to be used in the corresponding inter-agent negotiations, as discussed in (Cândido and Barata, 2007). Services provided by agents can be described using OWL-S in a similar way as semantic web services. Matchmaking of services can be then made using OWL reasoning.

Orchestration of manufacturing processes – service integration and composition becomes very attractive topic in the SOA domain. In (Hahn and Fisher, 2007) a solution based on multi-agent and holonic techniques is proposed. Community of interacting holons, representing service providers and requestors, can be nested so that the complex service requested is automatically orchestrated as a composition of basic services. An important function of a reconfigurable distributed manufacturing system is the distribution of tasks over multiple agents or holons. This goes beyond the simple service matchmaking – the whole process must be composed, executed and problems occurring during the runtime must be resolved. For that, manufacturing processes should be also specified in ontologies (Lastra and Delamer, 2006). We envision the ontology-based recipes compiled as a sequence of elementary operations described in a suitable ontology to allow automatic discovery of equipment that can perform requested operations.

Interoperability – property needed within a manufacturing system as well as with other systems running in a company. Translation between ontologies is a way of integrating systems that use different ontologies. The architecture of integrating systems has to be considered as well – the low level control devices would be hardly able to do such translation themselves, and so they need to ask a special service to provide translation for them or the translation has to be made automatically in the message transportation layer (Obitko, 2006).

6. SEMANTICS IN INDUSTRIAL SYSTEMS VERSUS WWW ORIENTED RESEARCH – SIMILARITIES AND DIFFERENCES

From the overview in the previous sections we can see that the usage of semantics has a great potential for industrial systems, especially for the distributed and reconfigurable ones. The use of semantic web technologies and the use of the research results in ontologies and multi-agent systems in general make the adoption of semantics easier. Based on our experience with the prototypes described earlier, we can identify areas where the semantic web problems are very similar to the problems in industrial systems; however, we can also notice some important differences.

One of the major problems of the semantic web research is the disambiguation of terms used in the textual web pages. The problem is to identify the context of a query to return relevant results, to identify what was meant by words used in the query. The situation in industrial domain is different. An assembly line is usually designed and described even before the first testing is started. Also, only what the engineers decide to connect to a line is really connected and provides data – there is no such freedom as on the web where anyone may add and claim anything. This means that the logical models of data are more under control and that there is no need for devices to cheat as the web pages cheat about their content on the web to get better ranks in search engines. The terminological disambiguation does not play any role, at least until the system is connected to human interface systems.

On the semantic web, it is generally not a problem if the result contains something else or does not contain what was wanted, especially because the query may be approximate. On the other hand, in industrial systems we want to get exact results to exact queries. Based on our experience with structural search, it is beneficial to add semantics to data using ontologies defined for specific projects. These ontologies are then used to construct exact structural queries that return results with exact precision and recall. The challenge is to provide simple user interfaces for construction of complex queries, instead of requiring users to learn special query languages such as SPARQL. Such an interface would be usable also for semantic web, however, the situation is easier for the industrial domain, since the number of ontologies and their size is smaller than on the semantic web.

Standardization of ontologies is viable on a specific part of the manufacturing domain rather than on the whole semantic web, which can contain anything. We believe that the standardization is possible to some degree, but also that the ontologies should be well founded on the principles such as DOLCE methodology and on the proper representation of spatial and temporal properties. These representation principles are currently researched outside the semantic web community, and usually complex logic is needed that is not computationally tractable. However, ontologies in these logics can be reduced to for example OWL notation and they still retain the advantages of proper design. Standardization of ontologies is possible among different vendors; however, there will still be a need for interoperability with unanticipated ontologies – the number of ontologies may be low in an assembly line, however, when systems are connected to MES systems or to other companies to form virtual enterprises, the problem of interoperability will arise. Well founded ontologies and ontologies founded on common upper ontologies will be easier to integrate without significant human intervention. Proper ontology design and interoperability could be achieved more likely in the manufacturing industry than on the general semantic web.

The important difference to the semantic web is that we are dealing with physical components. Based on our experience with the deployment of the multi-agent solutions, there are differences to pure software information agents. The components and processes that are to be connected together are not just computers that exchange bits in electronic way; they are physical devices that exist in a physical environment and are bounded by their physical properties. This means that the mistakes in communication or interpretation may be very costly – because of the potential damage of equipment, unnecessary material consumption, or because of delays in delivery of the manufactured product. That is why exact computationally tractable semantics is needed so that proper results can be guaranteed. This is different from text search, but similar threats exist in e-commerce or other business applications. Again, we are able to achieve better standardization and to form well founded ontologies, so the task is in this regard easier than in the case of the web.

An important part of a reconfigurable and fault tolerant industrial system is the ability to dynamically discover, compose and execute services. The research of semantic web services is applicable to such industrial systems; however, there is again the need of proper matchmaking and execution. The execution of a composed plan is more challenging than on the semantic web, as it is more likely that additional constraints or faults will occur and these problems will have to be resolved during runtime automatically.

Despite the differences, we can see both the world of the industrial systems and the world of the semantic web converging. In the beginning of the semantic web research, the most popularized way of adding semantics was to embed annotation to arbitrary web pages. This way is gaining popularity again as proper standards start to appear. However until now, most of the current semantic web is generated and not annotated manually. This is not surprising, since large part of the web is generated from databases – it is easier to generate the information and context than to embed it manually to static web pages. This is similar to our applications – we generate the annotations and the semantics of knowledge bases of agents or the semantics of messages to ensure that they will be consistent with our ontologies and that the applications of semantics will have desirable results.

7. CONCLUSION

As we have shown, the manufacturing domain is being influenced by the internet and WWW technologies, and especially the usage of semantic web technologies makes the adoption of semantics and ontologies easier. For distributed reconfigurable industrial systems the vision of the internet of services is an important inspiration, and despite some differences, much of the research is directly applicable to manufacturing domain. We have shown how semantics can be exploited from the low level control interoperability through shop floor integration up to the connection to SCADA/MES systems and later to the level of truly computer driven virtual enterprises.

REFERENCES

Al-Safi, Y. and V. Vyatkin (2007). An Ontology-Based Reconfiguration Agent for Intelligent Mechatronic Systems. In: *HoloMAS 2007*, LNAI 4659, pp. 114-126. Springer Berlin-Heidelberg.

Andreev, V., G. Rzevski, P. Skobelev and P. Shveykin (2007). Adaptive Planning for Supply Chain Networks. In: *HoloMAS 2007*, LNAI 4659, pp. 215-224. Springer Berlin-Heidelberg.

Batres, R., M. West, D. Leal, D. Priced, and Y. Nakaa (2005). An upper ontology based on ISO 15926. *Computers & Chemical Engineering*, **31**(5-6):519-534

Borgo, S. and P. Leitão (2004). The role of foundational ontologies in manufacturing domain applications. In: *On the Move to Meaningful Internet Systems 2004: CoopIS, DOA, and ODBASE*, LNCS 3290, pp. 670–688. Springer Berlin-Heidelberg.

Bussmann, S., N.R. Jennings and M. Wooldridge (2004). *Multiagent Systems for Manufacturing Control: A design Methodology*. Springer Berlin-Heidelberg.

Camarinha-Matos, L.M. (2002). Multi-Agent Systems In Virtual Enterprises. In: *Proceedings of International Conference on AI, Simulation and Planning in High Autonomy Systems*. pp. 27-36. SCS, Lisbon, Portugal.

Cândido, G. and J. Barata (2007). A Mutliagent Control System for Shop Floor Assembly. In: *HoloMAS 2007*, LNAI 4659, pp. 293-302. Springer Berlin-Heidelberg.

Cossentino, M., N. Gaud, S. Galland, V. Hilaire and A. Koukam (2007). A Holonic Metamodel for Agent-Oriented Analysis and Design. In: *HoloMAS 2007*, LNAI 4659, pp. 237-246, Springer Berlin-Heidelberg.

Dean, M. and G. Schreiber (2004). OWL Web Ontology Language reference. http://www.w3.org/TR/owl-ref/ (Accessed 25 September 2007)

Grüninger, M. and J. B. Kopena (2005), Planning and the Process Specification Language, In: *Proceedings of WS2 ICAPS 2005*, pp. 22-29.

Guarino, N. and P. Giaretta (1995). Ontologies and Knowledge Bases – Towards a Terminological Clarification. In: *Towards Very Large Knowledge Bases*. IOS Press, Amsterdam.

Hahn, Ch. and K. Fischer (2007). Service Composition in Holonic Multiagent Systems: Model-Driven Choreography and Orchestration. In: *HoloMAS 2007*, LNAI 4659, pp. 47-58. Springer Berlin-Heidelberg.

Lastra, J.L.M. and I.M. Delamer (2006). Semantic web services in factory automation: fundamental insights and research roadmap. *IEEE Transactions on Industrial Informatics*, **2**(1):1–11.

Lausen, H., A. Polleres and D. Roman (2005). Web Service Modeling Ontology WSMO. http://www.w3.org/Submission/WSMO/ (Accessed 25 September 2007)

Lemaignan, S., A. Siadat, J.-Y. Dantan and A. Semenenko (2006). MASON: A proposal for an ontology of manufacturing domain. In: *IEEE Workshop on Distributed Intelligent Systems (DIS)*, pp. 195–200. IEEE Computer Society Press.

Lopez, O. and J.L.M. Lastra (2006). Using Semantic Web Technologies to Describe Automation Objects. *Int. J. Manufacturing Research*, **1**(4):482-503.

Manola, F. and E. Miller (2004). RDF primer. http://www.w3.org/TR/rdf-primer/ (Accessed 25 September 2007)

McLean, C., Y. T Lee, G. Shao and F. Riddick (2005), Shop Data Model and Interface Specification, NISTIR 7198.

Obitko, M. and V. Mařík (2003). Adding OWL semantics to ontologies used in multi-agent systems for manufacturing. In: *HoloMAS 2003*, LNAI 2744, pp. 189–200. Springer Berlin-Heidelberg.

Obitko, M. and V. Mařík (2005). Integrating Transportation Ontologies Using Semantic Web Languages. In: *HoloMAS 2005*, LNAI 3593, pp. 189–200. Springer Berlin-Heidelberg.

Obitko, M. and V. Mařík (2006). Transparent Ontological Integration of Multi-Agent Systems. In: *IEEE International Conference on Systems, Man, and Cybernetics*, pp 2488-2492. IEEE SMC.

Obitko, M. (2007). *Translations between Ontologies in Multi-Agent Systems*. PhD thesis, Czech Technical University, Prague.

Rzevski, G., P. Skobelev and V. Andreev (2007). MagentaToolkit: A Set of Multi-agent Tools for Developing Adaptive Real-Time Applications. In: *HoloMAS 2007*, LNAI 4659, pp. 303-313. Springer Berlin-Heidelberg.

Schroth, C. and T. Janner (2007). Web 2.0 and SOA: Converging concepts enabling the internet of services. *IEEE IT Professional*, **9**(3):36–41.

The OWL Services Coalition (2004). OWL-S: Semantic markup for web services. http://www.daml.org/services/owl-s/1.0/owl-s.html (Accessed 25 September 2007)

Vrba, P. (2006). Simulation in agent-based control systems: MAST case study. *Int. J. Manufacturing Technology and Management*, **8**(1/2/3):175–187.

Vrba, P., F. Macůrek and V. Mařík (2006). Using Radio Frequency Identification In Agent-Based Control Systems For Industrial Applications. In: *Information Systems, Control and Interoperability – Preprints vol. I.*. pp. 459-464. IFAC.

Vyatkin, V., J. Christensen, J.L.M. Lastra and F. Auinger (2005). OOONEIDA: An Open, Object-Oriented Knowledge Economy for Intelligent Industrial Automation. *IEEE Transactions on Industrial Informatics*, **1**(1):4-17.