SMARTRESOURCE TEKES PROJECT – ANNUAL REPORT ”NETWORKING STAGE” (2006)

TEKES

SmartResource Project

Year 2006

Annual Report

“Networking Stage”

Industrial Ontologies Group

Agora Center, University of Jyväskylä

JSP Jyväskylä Science Park · metso automation · University of Jyväskylä · TeliaSonera · TietoEnator Building the Information Society · ABB

March, 2007
Jyväskylä, Finland

Table of Contents

Project's webpage: *http://www.cs.jyu.fi/ai/OntoGroup/SmartResource_details.htm*

Group's website: *http://www.cs.jyu.fi/ai/OntoGroup/*

Head of SmartResource Industrial Consortium (Steering Committee Head) **Dr. Jouni Pyötsiä**, Metso Automation Oy.

Jouni.Pyotsia@metso.com , Tel.: 040-548-3544

SmartResource Contact Person **Prof. Timo Tiihonen**, Vice-Rector, University of Jyväskylä

tiihonen@it.jyu.fi , Tel.: 014-260-2741

SmartResource Project Leader **Prof. Vagan Terziyan**, Agora Center, University of Jyväskylä

vagan@it.jyu.fi , Tel.: 014-260-4618

1 Introduction

Project Motivation

With the development of technologies, very fast creation and communication of information/knowledge has become possible. Automated knowledge accumulation and sharing is becoming the most profitable kind of business for modern, *knowledge-based*, companies. Such industries are looking for fast and global solutions related to Knowledge Management, Enterprise Application Integration, Electronic Commerce, Asset Management, etc. Various industrial standards, which have been created and implemented by different industrial consortiums, appear to be insufficient for growing interoperability demands.

One of the domains, where knowledge accumulation and its timely delivery are crucial, is industrial maintenance[1]. Development of a global environment, which would support automation of knowledge management for industrial maintenance, is a very profit-promising and challenging task. The latter is what the Smart Resource project aims at.

Our intention is to provide tools and solutions to make heterogeneous industrial resources (files, documents, services, devices, processes, systems, human experts, etc.) web-accessible, proactive and cooperative in a sense that they will be able to analyze their state independently from other systems or to order such analysis from remote experts or Web-services to be aware of own condition and to plan behavior towards effective and predictive maintenance.

Project Approach and Goal

The contribution of this ongoing SmartResource project (2004-2006) together with strong research effort includes prototype implementation of *distributed Semantic Web enabled maintenance management environment* with complex interactions of components, which are devices, humans (experts, operators) and remote diagnostic web-services. The environment will provide automatic discovery, integration, condition monitoring, remote diagnostics, cooperative and learning capabilities of the heterogeneous resources to deal with maintenance problems. Maintenance (software) agents will be added to industrial devices, which are assumed to be interconnected in a decentralized Peer-to-Peer network and which can integrate diagnostic services in order to increase the maintenance performance for each individual device. In the project, the maintenance case is expected to demonstrate the benefits and possibilities of new resource management framework and Semantic Web technology in general for Finnish industry.

Thus, project approach harnesses the potential of emerging progressive technologies – Semantic Web, Agent Technology, Machine Learning, Web Services and Peer-to-Peer – in addressing its very challenging goals.

Project Stages

Project research and development activities are divided into three yearly stages: Adaptation Stage (2004), Proactivity Stage (2005) and Networking Stage (2006). Each year of the project delivers more enhanced version of architectural design and prototype implementation for the maintenance environment.

---

[1] Metso Automation's customer magazine, (2003) *Automation*, *1*, 7-9.

Adaptation Stage defines Semantic Web-based framework for unification of maintenance data and interoperability in maintenance system. Its research and development tasks include development of generic semantic adapter mechanism (General Adaptation Framework) and supporting ontology (Resource State/Condition Description Framework) for different types of industrial resources: devices, software components (services) and humans (operators or experts). The key technology, which is utilized during the Adaptation Stage is *Semantic Web*. The latter is a relatively new initiative within W3C standardization effort to enable machine interpretable metadata in the Web. It provides standards and tools to enable explicit semantics of various Web resources based on semantic annotations and ontologies. Integration in general is considered nowadays as a "killer application" of Semantic Web technology, which particularly can be interpreted as heterogeneous data integration, Enterprise Application Integration and Web-service integration among other interpretations.

Proactivity Stage focuses on an architectural design of agent-based resource management framework and on enabling a meaningful resource interaction. Its research and development tasks include adding software agents (Maintenance Agents) to the industrial resources, enabling their proactive behavior. For this purpose, Resource Goal/Behavior Description Framework has to be designed, which will be the basis for making resource's individual behavioral model. The model is assumed to be processed and executed by the RGBDF engine used by the Maintenance Agents. Agent-based approach for management of various complex processes in the decentralized environments is being adopted and popularized currently in many industrial applications. Presentation of the resources as agents in the multi-agent system and use of technologies and standards developed by the Agent research community is a prospective way of industrial systems development. Creation of framework for enabling resources' proactive behavior and such agent features as self-interestedness, goal-oriented behavior, ability to reason about itself and its environment and to communicate with other agents, can bring a value to the next-generation industrial systems.

The objective of the **Networking Stage** comprises complex behavior/interaction scenarios of Smart Resources (agent-augmented Device, Expert and Service) in the global decentralized networked environment. The scenarios assume agent-based interoperation of multiple devices, multiple services and multiple experts, which allows discovery of necessary experts in Peer-to-Peer network, using their experiences to learn remote diagnostics Web-services, making online diagnostics of devices by integrating diagnoses from several services, learning models for a device diagnostics based on online data from several distributed samples of similar device, etc. Emerging Peer-to-Peer technology and similar network architectures suite well the increasingly decentralized nature of modern companies and their industrial and business processes, whether it is a single enterprise or a group of companies. The set of advantageous features of the Peer-to-Peer model includes decentralization, scalability and fault-tolerance along with low administration expenses. Client/server architectures with centralized management policy increasingly fail with big amounts of nodes, because of their complexity and extremely high demands on computing resources. Distributed content management systems address the need to access content wherever it resides, produce content while maintaining control over it, and collaborate efficiently by sharing data real-time within a distributed network of stakeholders.

2 Project Background Concept: a Global Understanding Environment

Global Understanding Environment (GUN)[2] is a concept used to name a Web-based resource "welfare" environment, which provides a global system for automated "care" over (industrial) Web-resources with the help of heterogeneous, proactive, intelligent and interoperable Web-services. The main players in GUN are the following resources: service consumers (or components of service consumers), service providers (or components of service providers), decision-makers (or components of decision makers). All these resources can be artificial (tangible or intangible) or natural (human or other). It is supposed that the "service consumers" will be able: (a) to proactively monitor own state over time and changing context; (b) to discover appropriate "decision makers" and order from them remote diagnostics of the own condition, and then the "decision makers" will automatically decide, which maintenance ("treatment") services are applied to that condition; (c) to discover appropriate "service providers" and order from them the required maintenance. Main layers of the GUN architecture are shown in Figure1.

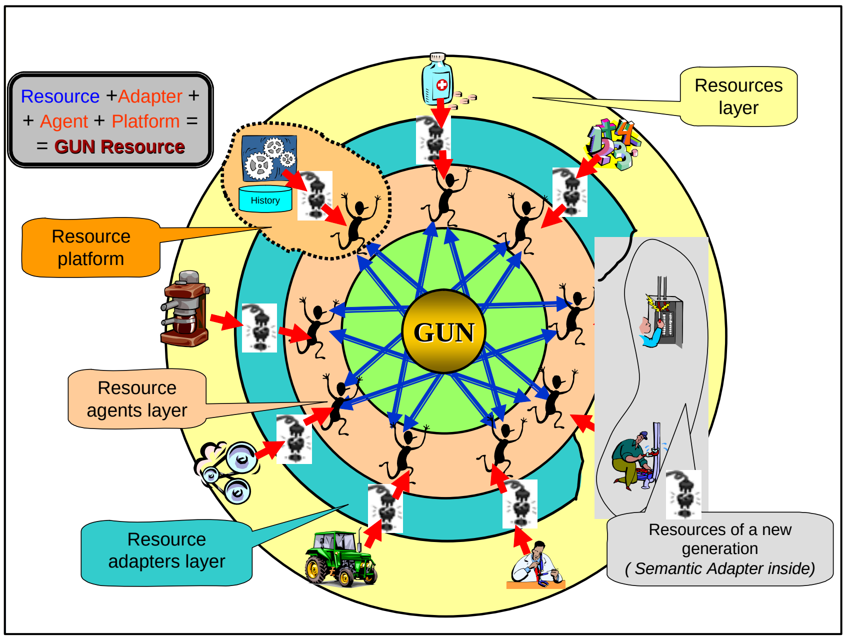

Figure 1 - Layers of the GUN architecture

Industrial resources (e.g. devices, experts, software components, etc.) can be linked to the Semantic Web-based environment via adapters (or interfaces), which include (if necessary) sensors with digital output, data structuring (e.g. XML) and semantic adapter components (XML to Semantic Web). Agents are assumed to be assigned to each resource and are able to monitor semantically reach data coming from the adapter about states of the resource, decide if more deep diagnostics of the state is needed, discover other agents in the environment, which represent "decision makers" and exchange information (agent-to-agent communication with semantically

[2] Terziyan V., Semantic Web Services for Smart Devices in a "Global Understanding Environment", In: R. Meersman and Z. Tari (eds.), *On the Move to Meaningful Internet Systems 2003*, LNCS, Vol. 2889, Springer-Verlag, 2003, pp.279-291.

enriched content language) to get diagnoses and decide if a maintenance is needed. It is assumed that "decision making" Web-services will be implemented based on various machine learning algorithms and will be able to learn based on samples of data taken from various "service consumers" and labeled by experts. Use of agent technologies within GUN framework allows mobility of service components between various platforms, decentralized service discovery, FIPA communication protocols utilization, and MAS-like integration/composition of services.

3 Project Results (Year 2006, Networking Stage)

General Networking Framework (GNF) – Deliverable 3.1

The General Networking Framework (GNF) considers an opportunity of ontological modeling of business processes as integration of component behavioral models of various business actors (agents representing smart resources in the web) in such a way that this integration will constitute the behavioral model of an agent responsible for the "alliance" of the components. This means that such "corporate" agent will monitor behaviors of the proactive components against the constraints provided by the integration scenario. Such model is naturally recursive and this means that the corporate agent can be a component in a more complex business process and will be monitored itself by an agent from the more higher level of hierarchy. Hierarchy of agents can be considered as possible mapping from the part-of ontological hierarchy of the domain resources.

The above motivates the main research objective of SmartResource project in 2006: "Design of a General Networking Framework as a platform for integration individual behaviors of proactive smart resources into a business process with opportunity to manage the reliability of components by certification, personal trust evaluations and exchange".

Accordingly first axiom (see Figure 2) of the Global Understanding eNvironment, Process – is similar resource to other resources in GUN (Device, Service and Human/Expert), but does not belong to the world of physical resources. As all GUN resources, Process has own properties that describe Process's state, history, sub processes and belongingness to upper-process (super-process). Thus, following principles of GUN resource, each Process is enhanced with an Agent that serves Process as a resource and actually realizes it as a behavior engine. Each process is a sequence of the actions (rgbdfs:Execution) that results in achievement of the final goal. So, each Agent per se is a process. In this case Agent Behavior plays role of a sequence of the actions and final result is represented by Agent Goal.

Axiom 1: ***Each resource in dynamic Industrial World is a process and each process in this world is a resource.***

Axiom 2: ***Hierarchy of subordination among resource agents in GUN corresponds to the "part-of" hierarchy of the Industrial World resources.***

Figure 2

Each GUN resource can theoretically be involved to several processes, appropriate commitments and activities, which can be either supplementary or contradictory. This means that the resource is part of several more complex resources and its role within each of the resource might be different.

There are some models of upper-process organization. But before we will talk about these models, we should state some definition. Let us consider executable module as an atomic non configurable actions. Thus, the choreography of a subject resource by its Agent via action performing is a non configurable atomic leaf-process. In this case, Agents behave accordingly to certain plan – planned set of behaviours. But, such simple processes can be organized in alliances – Process. The main function of a Process-Agent is the orchestration of a set of sub processes. Following this approach, architectures of arbitrary nested processes can be built, where leaf-processes are physical world Resource-Agents (Device-Agent, Service-Agent and Human/Expert-Agent).

One aim of Process (upper-process) creation is to organize cooperative work of sub processes for improving their individual performance. Each Agent should be supplied with a behaviour-planer module that generates plan for behaviour performance without any conflicts. And in this particular case, Process-Agent should utilize behaviour-planer to build plan of sub processes cooperative work and set constraints on their own plans. Another aim Process creation is to utilize other processes to reach another separate, lat us say - group-goal. In this case, achievement of the sub processes' goals depends on commitments and contracts between all parties. Thus, Agent-owner of this group-goal plays two roles: role of the sub process as another sub processes in this Process (with one difference – it has just goal and does not have atomic behaviour) and role of Process-Agent that performs orchestration of the sub processes. If we separate these two roles, we come to first model where we have blank sub process (has just goal and does not have any atomic behaviour) among sub processes, but achievement of this group-goal takes biggest priority. Figure 3 shows us generalized model when Process-Agent replans sub processes behaviours accordingly to sub processes goals achievement priorities.

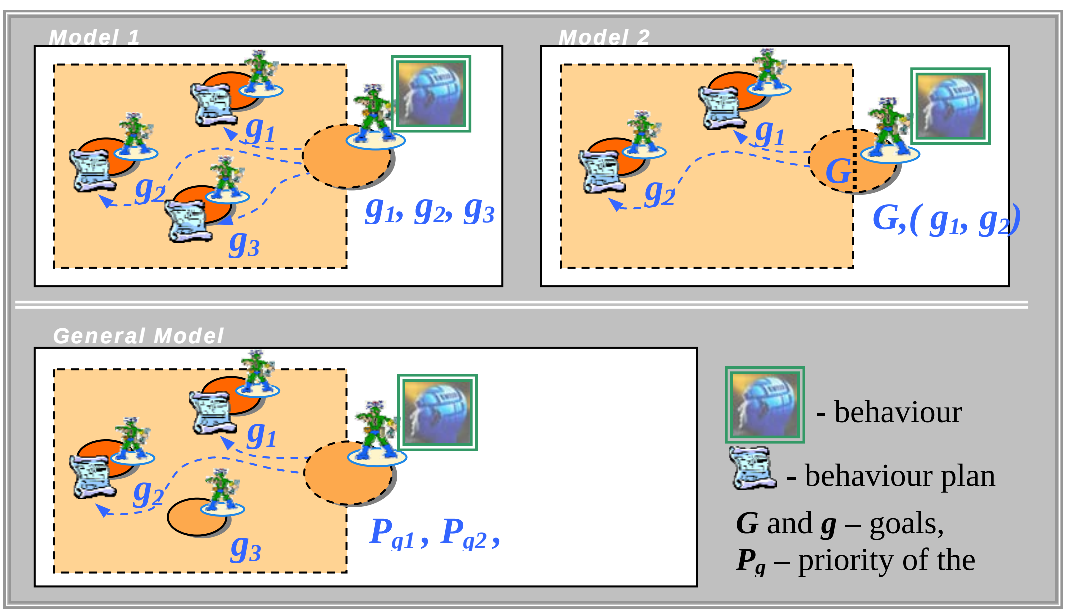

Figure 3

Nobody can guarantee stability of an environmental data if data space is shared among several Processes. It brings a need to replan the behaviour depending on the changes. The optimal way to reduce amount of replans is to collect all Processes that share same data space under one upper-process, if it is possible.

Generally, all the behaviours are represented by the set of rules that operate with the classes of resources (not the concrete instances). But during the behaviour processing by Behavior Engine all the rules are bounded with concrete instances. After such bindings we may have the conflict situations. If two processes use different instance spaces (spaces of facts, desires and etc.), then no conflicts may happen. But, if they share the same instance space, they can block others process performance by changing the shared information space. Actually while those Resource-Agents are living separately (resources are not members of some biggest process), no one cares about this conflicts of performance and they are concentrated just on achievement of the own goals. But when those two processes are members of another bigger upper-process, the duty of the Process-Agent is to resolve the conflicts via setting the constraints for behaviours of its members to reach the own goal and goals of the members (if it has been mentioned in a contract of the process). Initial behaviour of Process-Agent contains such set of actions as:

- Collection of all the behaviors of process members and convert them to the set of rules;
- Applying an algorithm to build a sequence of actions (performance plan) for optimal achievement of a final goal and intermediate goals (if necessary) based on behavior-rules of sub processes;
- Setting the constraints on behaviours of the members for conflict situations (when several rules may be applied, but result the different states – Environment State). In another words, we have a need to define and provide the meta-behavior-rules for the sub processes.

Such constraints (for process behaviour-rules) change behaviour of the Resource-Agent and restrict the degrees of an Agent freedom. Actually with its degree of freedom sub process sacrifices to upper-process when becomes a part of it. It is not necessary, that it negatively affects sub process's goal achievement, but often the opposite – it can result to speedup of the goal achievement.

Each industrial resource can theoretically be involved to several processes, appropriate commitments and activities, which can be either supplementary or contradictory. This means that the resource is part of several more complex resources and its role within each of the resource might be different. Modeling such resources with GUN can be provided by appropriate resource agent, which can make clones of it and distribute all necessary roles among them. Each industrial resource, which joins some commitment, will behave according to restrictions the rules of that commitment require. The more commitments individual resource takes, the more restriction will be put on its behavior. The main feature of the General Networking Framework is smart way of managing commitments (processes and contracts) of any proactive world resource (SmartResource) to enable cooperative behavior of it towards reaching also group goals together with the individual ones. Taking into account that world of industrial products and processes has multilevel hierarchy (based on *part_of* relation), we can say that it results to a hierarchical structure of GUN agents, which are meant to monitor appropriate world components in a cooperative manner.

A basic unit of monitoring in a power network is a *feeder*, which is a section of the power line including all the poles, conductors, insulators, etc. The start and the end point of a feeder are *substations*, whose task is to transform the electric power e.g. from high-voltage to medium-voltage or from medium-voltage to low-voltage. In addition to the transformer, any substation naturally includes the devices monitoring and protecting both the incoming and the outgoing feeders. Such *protection relays* automatically monitor the state of the feeder in terms of voltages and currents, are able to disconnect the feeder if a significant *disturbance* is registered, and to automatically re-close the circuit after a specified time (and to break it again if the disturbance persists).

Persistent disturbance is usually a sign of a *fault* in the network, which could be e.g. earth fault (conductor falling of the ground), short-circuit (could be caused e.g. by a tree falling on a line with bare conductors), or open circuit (broken line). Restoration of the network, after a fault occurs, includes *fault detection*, *fault localization* (estimating the geographic location of the fault), and of course fault removal. In meanwhile, network reconfiguration may also be performed, with a goal of e.g. minimizing the number of customers who will suffer outage of power until the fault is removed.

As mentioned, the fault detection is performed by protection relays. The rest is performed in the *operation centers* with participation of human *operators*. In case of a fault, protection relay sends an alarm to the operation center and also sends a dataset with recorded disturbance: several-second history of all the monitored parameters with a high frequency of sampling (0.5 ms or so). A certain operation center controls a sub-network of the integral power network. The operators use systems like ABB Distribution Management System (DMS) to have an integrated graphical view over the sub-network and ABB MicroSCADA for data acquisition from the substations and remote control over the relays, switches, etc. The systems like DMS also include implementations of various algorithms: for fault localization, for calculation of optimal reconfiguration of the network and other.

This deliverable described a case study in the domain of distributed power network maintenance we have been performing in collaboration with ABB Company (Distribution Automation unit). The goal was to study the potential add-value which ABB could receive from introducing Semantic Web technologies and Global Understanding Environment (GUN) framework in particular, into their business. Development of a prototype, for demonstration of the concept purposes, was a part of the study as well. The description of the practical work on adaptation of the general SmartResource platform for the electricity distribution field is a part of this report too.

An important feature of described solutions is that the existing software systems are not supposed to be replaced. Utilization of the SmartResource platform should aim at extending the interactions of existing systems and integration with various external systems, data storages, information services, and algorithms, which can be found in other organizations, or on the Internet (Figure 4).

In process of work, we discovered though several problems in existing ABB software products, hindering such development. A major of them is that DMS system does not provide any application programming interface (API), which would allow accessing its useful functionality by external applications, not only a human operator.

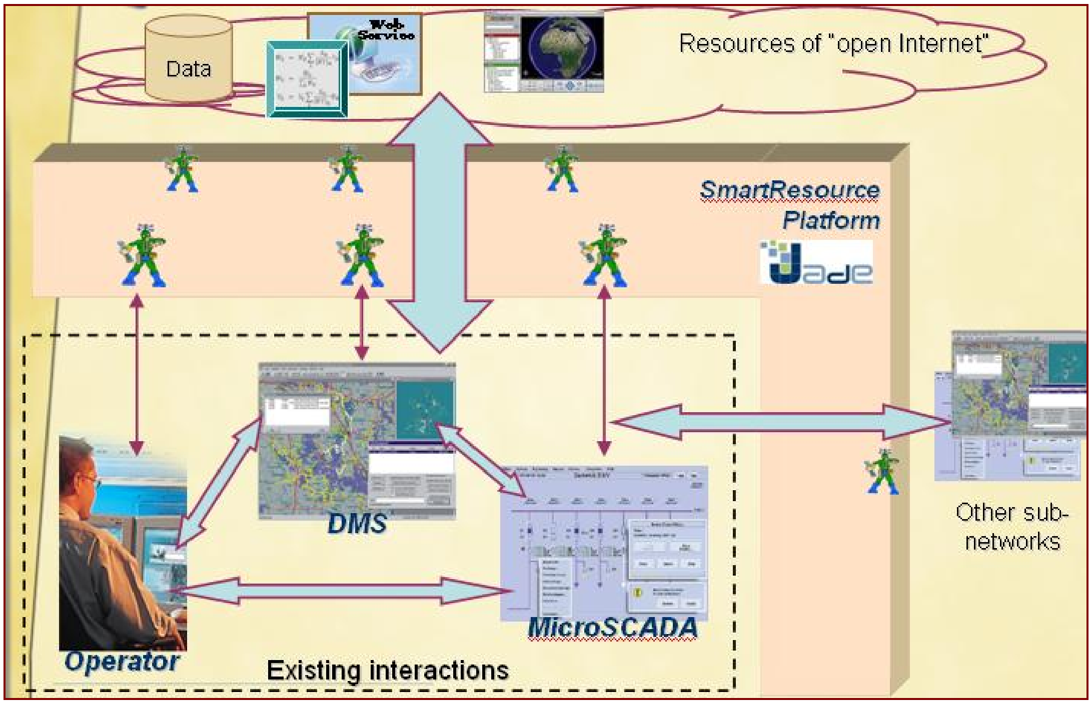

Figure 4 - Extending existing interactions

SmartResource Platform for Web Service Interactions' Semantic Log – Deliverable 3.3

METSO Automation[3] provides maintenance and technical support services to its clients all over the world. There is a VPN connection established between customer sites and METSO Automation maintenance center. Customers send fault messages in a SOAP/XML format to the maintenance center, where message data is analyzed by experts and may or may not be stored in a file system. Fault messages contain data which is potentially useful for fault analysis and predictive maintenance. To make it easily and flexibly processable, the maintenance center has to install a storage system of a new generation, which is easy to maintain, extend and query.

Based on the problem description, we have selected a semantic web-based approach to storage and retrieval of log data. Metso Automation is a big enterprise which may require a more complex integration solution in the future. Having a lot of customers and providing a large variety of services, METSO Automation will benefit from an integral (pseudo-)centralized storage of its business data including but not limited to maintenance activities. We have selected a Semantic Web technology for implementation because it has a set of distinctive features which enforce data representation. Here we can name at least graph-based data representation, simple, but effective class-subclass and property-subproperty relationships, bringing add values, which are hard to implement using RDBMS[4]. These features are naturally supported by most of the semantic storages and may be extended using third-party inference engines.

The system presented here comprises a set of components, which serve as an interface for SOAP messages handling and transformation, interaction with semantic storage and storage browser (see Figure 5).

---

[3] Metso Automation – a leading provider of software and maintenance solutions in paper industry

[4] Relational Database Management System http://en.wikipedia.org/wiki/Relational_database_management_system

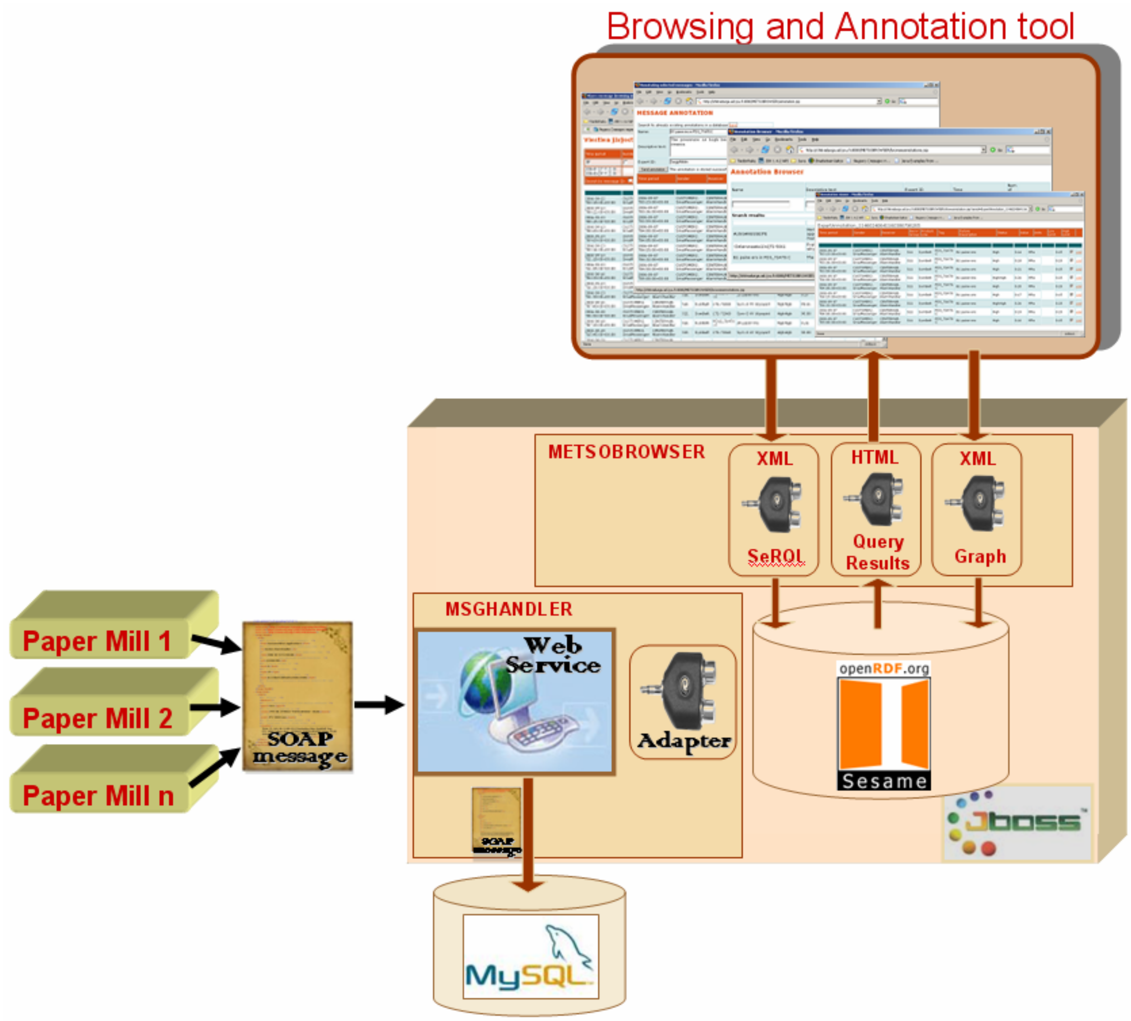

Figure 5 – System Architecture

The system can be divided into two main subcomponents – Message Handler and Message Browser. Message Handler receives and processes SOAP messages from different customers. Message Browser provides functionality for message browsing, filtering and annotation. Both subcomponents run on a JBoss[5] application server and are independent from each other.

The system presented here demonstrates the applicability of Semantic Web technology on a real industrial case. The system performs transformation from XML formats to RDF and provides browsing and annotation facilities for stored data. The system development process has shown that it is possible to combine different Semantic Web tools in a different modeling and execution tasks such as ontology modeling in the Protégé tool and ontology insertion into the Sesame storage. It is pretty simple to add new classes or properties in a Protégé tool and then just copy-paste the updated model in RDF format to Sesame without any restarts. The model is updated in a couple of seconds while the system continues to run. We can say that ontology-based approach is extensible, although we did not test what happens when we make major changes to the ontological model such as changing domain and range of properties or deleting

[5] JBoss - application server http://www.jboss.com/

classes. Adapters are the most sensitive components to changes in the structure of the incoming formats, messages and ontology. In our opinion the adapter transformation function should be tied together with the ontology and react immediately on changes which lead to inconsistency of the data. We see one of the future research challenges in an elaboration of a user interface development process for ontology-based applications. In the implementation of a browsing tool we have extensively used AJAX technology and XML-based messaging. We have realized a need for a script-based engine-like client side visualization library for ontology based applications.

4 3rd Project Year Publications (up to the beginning of year 2007)*

[1] Terziyan V., Katasonov A., Global Understanding Environment: Applying Semantic Web to Industrial Automation, In: J. Cardoso, M. Hepp, M. Lytras (eds.), Real-world Applications of Semantic Web Technology and Ontologies, Springer, 36 pp. (Book chapter, submitted 30 October 2006).

[2] Naumenko A, Srirama S., Terziyan V., Jarke M., Semantic Authorizations of Mobile Web Services, In: Computer Communications Journal, Special Issue on Security on Wireless Ad Hoc Networks, 2006, Elsevier, ISSN: 0140-3664, 28 pp. (submitted 18 December 2006).

[3] Nikitin S., Terziyan V., Pyotsia J., Data Integration Solution for Paper Industry, In: J.P. Müller and K. Mertins (Eds.), In: Proceedings of the 3rd International Conference on Interoperability for Enterprise Software and Applications (IESA-07), March 28-30, 2007, Madeira Island, Portugal, 13 pp. (submitted 2 November 2006).

[4] Terziyan V., Challenges of the "Global Understanding Environment" based on Agent Mobility, In: V. Sugumaran (ed.), Advanced Topics in Intelligent Information Technologies, Vol. 1, Idea Group, 2007, pp.121-152. (Book chapter, to appear).

[5] Ermolayev V., Keberle N., Kononenko O., Terziyan V., Proactively Composing Web Services as Tasks by Semantic Web Agents, In: L.J. Zang (Ed.), Modern Technologies in Web Services Research, Idea Group Inc., 2006, 31 pp., (book chapter, to appear).

[6] Salmenjoki K., Tsaruk Y., Terziyan V., Viitala M., Agent-Based Approach for Electricity Distribution Systems, In: Proceedings of the 9-th International Conference on Enterprise Information Systems, 12-16, June 2007, Funchal, Madeira, Portugal, 8 pp. (to appear).

[7] Naumenko A., Katasonov A., Terziyan V., A Security Framework for Smart Ubiquitous Industrial Resources, In: J.P. Müller and K. Mertins (Eds.), In: Proceedings of the 3rd International Conference on Interoperability for Enterprise Software and Applications (IESA-07), March 28-30, 2007, Madeira Island, Portugal, 13 pp. (to appear).

[8] Khriyenko O., Coordination of the Distributed Proactive Smart Resources, In: Proceedings of the IASTED International Conference on Parallel and Distributed Computing and Networks (PDCN 2007) as part of the 25th IASTED International Multi-Conference on APPLIED INFORMATICS, Innsbruck, Austria, 13-15 February, 2007, 6 pp.

[9] Ermolayev V., Terziyan V., Kaykova O., SW @ UKRAINE, In: M. Lytras (Ed.), Semantic Web Factbook 2005, AIS SIGSEMIS, 2006, ISSN: 1556-2301, pp. 13-17.

* Papers are downloadable from *http://www.cs.jyu.fi/ai/OntoGroup/SmartResource_details.htm*

[10] Kaykova O., Khriyenko O., Kovtun D., Naumenko A., Terziyan V., Zharko A., Challenges of General Adaptation Framework for Industrial Semantic Web, In: A. Sheth and M. Lytras (eds.), Advanced Topics in Semantic Web, Idea Group, Vol. 1, 33 pp..

[11] Khriyenko O., Terziyan V., A Framework for Context-Sensitive Metadata Description, In: International Journal of Metadata, Semantics and Ontologies, ISSN 1744-2621, 11 pp. 2006.

[12] Luostarinen K., Naumenko A., Pulkkinen M., Identity and Access Managementfor Remote Maintenance Services in Business Networks, In: Proceedings of the sixth IFIP conference on e-Commerce, e-Business, and e-Government I3E'2006, October 11-13, 2006, Turku, Finland (in press) (done in cooperation between IOG (SmartResource) and MODPA project).

[13] Naumenko A., Nikitin S., Terziyan V., Service Matching in Agent Systems, In: International Journal of Applied Intelligence, In: M.S. Kwang (Ed.), Special Issue on Agent-Based Grid Computing, Vol. 25, No. 2, 2006, ISSN: 0924-669X, pp. 223-237.

[14] Terziyan V., Bayesian Metanetwork for Context-Sensitive Feature Relevance, In: G. Antoniou et al. (eds.), Proceedings of the 4-th Hellenic Conference on Artificial Intelligence (SETN 2006), Lecture Notes in Artificial Intelligence, Vol. 3955, 2006, pp. 356-366.