Walery ROGOZA, Michał ZABŁOCKI

Westpomeranian Technological University in Szczecin

Grid computing and Cloud computing in scope of JADE and OWL based Semantic Agents – A Survey

***Abstract**. This work is an overview of the state of art of distributed computing in scope of the new trends in artificial computational intelligence. Grid computing as main ingredient of cloud computing is most vulnerable to the positive effect of connection with multi-agent systems. The Web Semantic Language (OWL) can help to provide structured semantic description of the existential environment for agent and support them in processing knowledge which gives agent better ability to reasoning and acting in intelligent manner.*

***Streszczenie.** Niniejsza praca jest próbą zobrazowania stanu techniki obliczeń rozproszonych pod kątem nowych trendów w dziedzinie sztucznej inteligencji. Grid jako główny składnik Chmury obliczeniowej jest najbardziej podatny na pozytywny efekt połączenia z systemami wieloagentowymi. Język opisu Semantycznej Sieci (OWL) może przyczynić się do zapewnienia strukturyzowanego opisu środowiska egzystencjalnego dla agentów, przez co wesprzeć ich w efektywniejszym rozumowaniu i działaniu w inteligentny sposób. (**Tworzenia Semantycznych Agentów w zakresie przetwarzania w Grid i w Chmurze – Przegląd**).*

Keywords: Grid computing, Cloud computing, The Web Semantic Language, Java Agent Development Framework.
Słowa kluczowe: Grid, Chmura obliczeniowa, OWL, JADE.

doi:10.12915/pe.2014.02.25

Introduction

Nowadays, where cloud computing become very popular, it's good to have a look on how intelligent agents using OWL semantic language influenced on distributed computing. A grid computing as main ingredient of cloud computing is most vulnerable to the positive effect of connection with multi-agent systems (MAS) [1]. MAS, as a system composed of intelligent agents (which solves humans problems) that are able to exist in distributed heterogeneous informational environment. They need to have a common standard mechanism to represent and reasoning about possessed knowledge.

The Web Ontology Language (OWL) [2] is an ontology language that allows represent knowledge considered as states, rules or predicate logic [3]. It is strongly connected with the idea of Semantic Web [4]. Ontologies are ideal for machine processing and thus increase the interoperability and machine understandability.

JADE is probably the most widespread agent system in today use. This FIPA compliant agent platform designed to be a middleware for developing multi-agent applications based on the software agent paradigm. JADE is an open source platform, specifically designed for creating an applications in heterogeneous and uncertain environments. JADE is not only suitable framework in the context of research driven applications but as well in commercial application.

Grid systems (GRID) and MAS used to be considered as two different kinds of distributed systems intended for solving two different types of computational problems. Just recently came to recognize the benefit of the combination of these two different systems. The GRID provide an computational environment for agents, and MAS enriched GRID with intelligent behaviors.

It seems to MAS found the biggest application in Cloud computing. This commercial model of distributed computing can derive the most from the potential of MAS. Intelligent agents can support Clouds users in automatic SLA negotiations, resource management and incident detection.

Grid computing

The grid computing is a form of data processing via a computer network. The main concept is that all computer's resources like processing power, memory and data storage are shared with the every computer in the system. The main goal is to force all computers to work for the benefit of one system. In the grid all connected computer's are seen as one powerful processing center. This special kind of distributed computing forcing the computer scientists, programmers and engineers to creating, establishing and implementing dedicated standards and protocols.

The main issue that touch the grid computing is a coordination of shared resources, in means of direct access to the computer, software, data, and other resources to solve a problem (task). This coordination includes range declaration of shared resources, access roles, and the circumstances in which sharing occurs. [5] The problem rises when the system has to work in dynamic and heterogeneous environment. That is why a key to building a system compliant with this requirements is standardization, which allows to support interoperability, portability and reusability of components and systems. The service-oriented architecture, the Open Grid Services Architecture (OGSA ™) is addressed to fulfill this needs “by a set of core capabilities and behaviors that address key concerns in Grid systems”[1].

Cloud computing

For Cloud computing is a model of information processing based on services provided by (external or internal) provider. As a services we should understand an on-demand network access to shared pool of configurable computing resources. The access should be possibly ubiquitous and convenient. The resources should be provided as rapidly as is it possible, released without any unnecessary configuration and with minimal provider interaction. [7]

There are three delivery models: Software as a Service (SaaS), Platform as a Service (PaaS) and Infrastructure as a Service (IaaS). Each model differ in range of providing computing resources. The choice is dependent on the customer needs. In the SaaS model provider deliver solely an application for the customer, who can not control an operating system, hardware or network infrastructure in which the application works. PaaS model provides a environment for customer software applications. The provided environment in some cases can by configurable, but client can not control operating system, hardware or network infrastructure. Finally, the IaaS model gives to customer control over processing power, storage, networking components or middleware. Customer decides

[1] [6], page 4.

which operating system he will use, how much storage capacity will need, what applications will deploy and what networking components (i.e. firewalls and load balancers) will turn on.

The customer can decide between four types of deployments models: public, private, community and hybrid. The public cloud system provide services that are available to all via the Internet. Usually the services are free or offered on a pay-per-use model. In difference to public cloud, the private one can be reserved for dedicated group of users and there is no network bandwidth restrictions or any others imposed by provider. In the community cloud an access is mostly available for users who belongs to a community. The hybrid cloud is a fusion of a public and private cloud. The customer use private cloud for processes that are security-critical, but others are managed in public cloud.

With the subject of the cloud computing is related a lot of important issues. An organizations concerned about the future of this idea was gathered and discussed main problems. In the results was released a paper [8], which describes crucial customers expectations for cloud infrastructures and formulates basic principles that must be followed by the cloud providers to satisfy the customers.

JADE

Java Agent DEvelopment Framework is a software framework for development a multi-agent systems. Its fully implemented in Java language as a fully compliant with FIPA Specifications[2] middle-ware. JADE provides a set of graphical tools to support programmers in debugging and deployment phases. The middle-ware can by fully distributed across the computer (or mobile devices) network. JADE is an open source project started, maintained and distributed by Telecom Italia, the copyright holder. [12]

An intelligent software agent is a unit, which state is considered as set of mental components such as beliefs, capabilities, choices and commitments. It is located in some environment and is capable of autonomous behavior in this environment in order to reach programmed goals. [13][14] In accordance with the agent-oriented programming (AOP) an agent can not provide an interface to pass the control over themselves to another agent. The agent has his own execution thread that controls his own life cycle - his actions are fully autonomous. The basic form of communication between agents is asynchronous message passing. It is a sole channel of interaction and manner of attempt exert influence on each other. However, a receiver decide if ignore or cooperate with a sender of the message.

On the basis of this requirements JADE support the following features: a) fully distributed environment as a existential platform for the agents, where every agent has his own thread and can communicate with others agents in transparent way, b) very effective asynchronous message transport protocol that provides location transparency, c) implementation of white and yellow pages provides easy search mechanisms for agents and their services, d) easy, but still effective agent life cycle management watches over uniqueness of agent ID, e) support to agent mobility provides mechanism of agent code transfer (with storing the agent state) to other platform, f) flexible core that allows programmers for adding new features.

OWL

The intelligent behavior of the computer program would require an ability to choose a proper algorithm to a given problem. The algorithm is a description of the procedural knowledge of the program and the problem is declarative knowledge of the domain of application. Writing an intelligent program it should be a possibility to express this knowledge in the same formalism, then both would be equally accessible. OWL allows to embed in the program both description of a application environment, and the knowledge of how to deal with the problems of that environment. [9]

The OWL Web Ontology Language is a intelligent knowledge representation language designed to describe ontology. Ontology models a domain of knowledge using a set of representational primitives to defines the terms.[10] OWL is a part of the W3C Semantic Web [4] project, which the main goal is to support automated access to information. Ontology allows to describe information by the metadata which makes them meaningful for the computer during processing. Ontologies are intended to share specific domain information between diverse devices. OWL describes classes (general things), the relationships that can exist among them and the properties (or attributes) that those things may have. It greatly supports automated reasoning which is a background for intelligent behavior of applications. [11]

The semantic of OWL is based on RDF/XML serialization. XML Schema language is used in OWL, it is restricting the XML document structure and extends it with datatypes. To express facts OWL uses the Resource Description Framework (RDF). RDF is a simple semantic data-model for objects and relations between them, that can be represented in an XML syntax. Analogously to XML Schema, RDF Schema (RDFS) defines classes, properties and their hierarchies for RDF data-models. Additionally OWL extends RDFS and allows to define more complex classes. [11]

Nowadays the combination of OWL and SAM became a standard. The application of OWL-based solutions provides many features including the autonomous semantic-based content discovery [32] and Web services discovery [33].

MAS-Grid with OWL ontology support

Multi-agent systems and Grid systems (GRID) are, on first sight, two different approaches to distributed computing. The Grid systems used to be concerned as a systems infrastructures (tools and applications) for reliable and secure computer resource sharing in dynamic and distributed computer network in order to solving high demand computational problems (for example Grand Challenge problems [15]). In the same time MAS were explicit in the direction of solving problems that require autonomous and intelligent actions in flexible and uncertain environment. Recently engineers began to notice the benefits of combination of those two systems.

The one of the firsts proposal was to consider agent as a service which increase service autonomy. Thous agents comes together to form a new virtual organization. This approach assumes a cooperation of initially distinct agent-services to form new service entity. [16] The first attempts of integration JADE MAS with GRID was concentrated on creation of improved agents. Those agents possessed ability to create and compose role-based tasks and exchange them between themselves. [17]

After MAS technology adopted the idea of building semantic agents the natural course of things was idea of reusing semantic web solutions in GRID technology [18] and ultimately propose the Semantic Grid as a combination

[2] Foundation for Intelligent Physical Agents (FIPA) is an organization that propagates technology for heterogeneous and intelligent agents. A list of specifications that they released are a set of standards that describes interoperation of heterogeneous agents and the services that they can offer. See http://www.fipa.org/.

of GRID, MAS and OWL-based ontology. The realization of this idea gives better integration of MAS and GRID technologies and resulting in rise of autonomy of the services. [19]

Nowadays MAS and GRID seek to full integration with SOA and that why are starting to be perceived as a part of Service-oriented Computing. Clément Jonquet in his thesis [20] brings an idea of Dynamic Service Generation (DSG). He proposes a concept of a new approach to human–service interaction in reference to SOA. During this interaction seen as conversation between user and service provider while the needs of the user will be clarifying, the services will be constructed on the fly in reference to user needs.

A group publication [21] gives an overview on available at the moment implemented components that provides transparent bidirectional channel of interaction between web service and agent. The publication also compares proposed solutions and pointing on the good and bad sites of them.

The concept of the semantic web brought a new need for approach to automated composition of atomic Web services into more complex Web processes. In papers [23][24] authors proposes, in accordance with the standards of SOA, a centralized approach based on UDDI Agent. On the contrary in work [22] we can meet with more MAS standards approach. The authors proposes an agent-based middleware implementing Web services dynamic composition and agent reasoning model based on Peer-to-Peer network.

Multi-agent Cloud Systems

As the GRIR is considered as a main part of Cloud computing system, it is obvious that MAS also finds application in it. Clouds systems are providing them services through a service-oriented interface. In paper [25] there are discussed analogies, differences and potential synergies between Clouds and multi-agent systems. The authors are pointing that Cloud computing can supply a very powerful, reliable, predictable and scalable existential environment for software agents. On the other hand, agents can support the Clouds systems in implementation of intelligent solutions. Such solutions would make Clouds system more adaptive, flexible, and autonomous in resource management. An autonomous agents can support Cloud systems in in the interaction with users and makes those systems more efficient in allocating processes and data storage.

In large-scale data centers, agent components can perform data-mining tasks like searching, filtering, but also queering and updating the massive volumes of data. Cloud agents working on our and provider behalf, can monitor services, deploy processor-to-application assignment strategies, and manage energy-efficient use of Cloud computing infrastructures. [25]

In SaaS model, agent can optimize the usage of applications provided as services and manage the hardware and software infrastructure taking care of its efficient utilization and finally can support the provider in providing proper QoS. In PaaS model, agent can assist to software developers in deployment of their applications in Cloud. [25]

In paper [26] author uses MAS to support negotiations of service-level agreements (SLAs) and in [34] is described a model to monitor and control multi level SLAs. The negotiations between resource provider and customer is potentially best field for multi-agent technology implementation. Since customer requirements can vary over time, intelligent agents can support autonomous resource mapping and dealing with changing customers demands.

The paper [27] presents a Cloud service discovery system (CSDS) to support customers in finding a Cloud services over the Internet. The system using the Cloud ontology is looking for similar services. The system is based on popular search engines (like Google, Bing, Yahoo), is composed of three agents (Filtering Agent, Query Processing Agent and Cloud Service Reasoning Agent) and dedicated Cloud ontology. It generates queries for default search engine, filters and evaluates search results.

The idea of providing easy access to the information about available Cloud providers are carried on in [28]. In this paper authors presents an agent-based Cloud service composition that works in multi-Cloud environments. The author's Cloud service composition has two dimension (vertical and horizontal) and two modalities (one-time and persistent). It is composed of agents that represents Cloud participants and agents that wraps and controls the Cloud resources. Thanks to agents abilities to self-organization, integrated into their behaviors, the continuously changing consumer requirements may be fully supported. The experiment results presented in [28] prove the high performance of proposed solution.

In paper [30] authors are introducing the cloud incident detection system Security Audit as a Service (SAaaS). Since there is no flexible enough intrusion detection mechanisms to cope with cloud specific security issues. Cloud infrastructure can changing constantly, that is why the security audits are very desired. They proposed building intelligent autonomous software agents, that are aware of underlying business driven intercommunications of cloud services, in order to support cross customer event monitoring within a cloud infrastructure.

All this research seeks to improve state of art of cloud computing. All work are targeted (even though not directly) to fulfill the issues of Open Cloud Manifesto [8]. Cloudware project [29] is an attempt of creating Open-Source architecture based on the best solutions. It is based on MAS to support the Cloud system in agent-based monitoring, SLAs negotiations and ontology.

Conclusion and Perspectives

In the light of works presented above about combining GRID with MAS the following conclusions can be formed.

The integration between Web services and FIPA compliant agents (agents) has to be operationally transparent to invoking both entities. A mechanism (gateway) should seamlessly transform the message, issue it and transcribe any response before returning it to the invoking entity. The whole process should be fully automated without manual intervention or configuration. It is desired that the above features were encapsulated and integrated in the gateway and require no additional external resources.

The integration of agents and Web Services need to provide an ability to finding the location, communication method, and services offered by these entities to enable an interoperability between them. To ensure that agents discover Web Services (and vice versa), the following requirements should be fulfilled: a) Translation of WSDL Web Service descriptions into FIPA Agent Service descriptions (and vice versa) should be possible. b) Agents should be able to discover Web Services published via UDDI. c) Web Services Client should be able to discover agent service descriptions for those agents that are designed to interoperate with Web Services.

Thanks to the integration a Web services (combined with agents) can became self-aware, can gain awareness of existence of other Web services and their capabilities as a services that they provides. Web services cooperating with agents can provide higher-level and more comprehensive services. It is important to provide semantic description of all relevant aspects of the services. Using Web Service Modeling Ontology (WSMO) [31] we can facilitate the automation of discovering, combining and invoking electronic services over the Web. Based on the multiagent framework it is possible to decompose a problem on many tasks that can be performed by atomic Web service.

Starting from agent-based SLAs negotiations, discovery and integration of cloud services, through agent-based security modules to agent-based intelligent and autonomic cloud system where agents proved their invaluable contribution to improvement of cloud services. But there are still giant area of agents application in cloud systems, for example load-balancing, task allocation, automated service deployment.

OWL thanks to the W3C promotion becomes a standard widely used in systems based on ontology. A lot of new projects connected with Semantic Web are formed. There is no doubt that it is main trend that leads to better machine reasoning.

AOP and MAS systems are the future of software engineering. Classical object-oriented programming is not fully compliant with new programming trends, viz human-computer interaction. New user interfaces are based on conversations in order to better understanding user need by service. Agent-oriented paradigm can unreservedly fulfill this new requirements.

REFERENCES

[1] Foster I., Jennings N.R., Kesselman C., Brain meets brawn: why Grid and agents need each other, *3rd International Joint Conference on Autonomous Agents and Multiagent Systems*, AAMAS'04. 1 (2004), 8–15

[2] Smith M. K., Welty C., McGuinness D. L., OWL Web Ontology Language Overview, *W3c recommendation*, World Wide Web Consortium (2004)

[3] Laclavik M., Babik M., Balogh Z., Hluchy L., AgentOWL: Semantic Knowledge Model and Agent Architecture, *Computing and Informatics*. Volume 25., (2006), no. 5, 419-437

[4] Berners-Lee T., Hendler J., Lassila O., The semantic web, *Scientific American*, 284 (2001), No. 5, 34-43

[5] Foster I., Kesselman C. and Tuecke S, The Anatomy of the Grid: Enabling Scalable Virtual Organizations, *International Journal of Supercomputer Applications*, 15 (2001), No. 3, 200-222

[6] Foster I., Kishimoto H., Savva A., Berry D., Djaoui A., Grimshaw A., Horn B., Maciel F., Siebenlist F., Subramaniam R., Treadwell J., Von Reich J., The Open Grid Services Architecture, *Open Grid Services Architecture WG, Global Grid Forum*, 1.5 (2006)

[7] Cloud Computing Use Case Discussion Group, Cloud Computing Use Cases. Version 4.0, (2010), Url: www.opencloudmanifesto.org

[8] Open Cloud Manifesto, (2009), Url: http://www.opencloudmanifesto.org

[9] Szolovits P., Hawkinson L., Martin W.A., An Overview of OWL, A Language for Knowledge Representation, *MIT/LCS/TM-86, MIT Laboratory for Computer Science*, (1977). http://people.csail.mit.edu/psz/ftp/OWL_overview1.html

[10] Gruber T., Ontology, *Encyclopedia of Database Systems*, Springer-Verlag, 2009, http://tomgruber.org/writing/ontology-definition-2007.htm

[11] McGuinness D. L., van Harmelen F., OWL Web Ontology Language, *W3C Recommendation* (2004), http://www.w3.org/TR/owl-features/

[12] Java Agent DEvelopment Framework Home page. URL: http://jade.tilab.com. January 2013

[13] Shoham Y., Agent-Oriented Programming, *Artificial Intelligence*, 60 (1993), 51 – 92

[14] Jennings N. R., Wooldridge M., Applying Agent Technology, *Journal of Applied Artificial Intelligence*, 9/6 (1995), 357-370

[15] Task Force Report Grand Challenges, nsf.gov. (2007), URL: http://www.nsf.gov/od/oci/taskforces/TaskForceReport_GrandChallenges.pdf

[16] Foster I., Jennings N. R., Kesselman C., Brain meets brawn: why Grid and agents need each other, *3rd International Joint Conference on Autonomous Agents and Multiagent Systems*, AAMAS'04, 1 (2004), 8–15

[17] Poggi A., Tomaiuolo M., Turci P., Extending JADE for Agent Grid Applications, *Proceedings of the 13th IEEE International Workshops on Enabling Technologies: Infrastructure for Collaborative Enterprises* (2004)

[18] Jabisetti N., Lee Y., OWL-S-based autonomic services for grid computing. *Proceedings of the IEEE International Conference on Web services, (2005),* 825-826

[19] Pasha M., Ur Rehman S., Ali A., Ahmad H. F., Suguri H., Semantic Grid: Interoperability between OWL and FIPA SL, *Lecture Notes in Computer Science*, 4088 (2006), 714-720

[20] Jonquet C., Dynamic Service Generation: Agent interactions for service exchange on the Grid, PhD thesis, (2006), University Montpellier 2, Montpellier, France

[21] Greenwood D., Lyell M., Mallya A., Suguri H., The IEEE FIPA Approach to Integrating Software Agents and Web Services. AAMAS'07, (2007)

[22] Liu S., Kungas P., Matskin M., Agent-Based Web Service Composition with JADE and JXTA, *Proceedings of the 2006 International Conference on Semantic Web & Web Services*, (2006), 110-116

[23] Zhang L., Rao G., Proactive semantic web service system based on intelligent UDDI agent, Control and Decision Conference, (2008), 1354 – 1357

[24] Kumar S., Mishra R. B., Multi-Agent Based Semantic Web Service Composition Models, Circuits,Communications and System (PACCS), 2010 Second Pacific-Asia Conference, 2 (2010), 67 - 70

[25] Talia D., Cloud Computing and Software Agents: Towards Cloud Intelligent Services, *WOA*, (2011), 2-6

[26] Sim K.M., Towards Complex Negotiation for Cloud Economy, *Proceedings of the 5th International Conference on Advances in Grid and Pervasive Computing,* (2010), 395-406

[27] Han T., Sim K. M., An Agent-Based Cloud Service Discovery System that Consults a Cloud Ontology, *Intelligent Control and Computer Engineering*, 70 (2011) 203-216

[28] Gutierrez-Garcia J. O., Sim K. M., Agent-based Cloud service composition. *Applied Intelligence*, (2012)

[29] Petcu D., Frincu M., Craciun C., Panica S., Neagul M., Macariu G., Towards Open-Source Cloudware, *Proceedings of the 2011 Fourth IEEE International Conference on Utility and Cloud Computing (UCC '11),* (2011), 330-331

[30] Doelitzscher F., Reich C., Knahl M.,H., Passfall A., Clarke N. L., An agent based business aware incident detection system for cloud environments, *Journal of Cloud Computing: Advances, Systems and Applications*, (2012), 1-9

[31] Roman D., Lausen H., Keller U. (eds.), Web Service Modeling Ontology (WSMO), (2005), http://www.w3.org/Submission/WSMO

[32] Fornara N., Jezic G., Kusec M., Lovrek I., Podobnik V., Trzec K., Semantics in Multi-Agent Systems, Chapter 7. S. Ossowski (ed.), *Agreement Technologies, Law, Governance and Technology Series*, Springer Science+Business Media Dordrecht, 8 (2013), 115-136

[33] Rohallah B., Ramdane M., Zaidi S., Agents and owl-s based semantic web service discovery with user preference support, *International Journal of Web & Semantic Technology (IJWesT)*, 4 (2013), No. 2, 57-75

[34] Singh S., Arora M., Monitoring and Controlling Multi Level SLA in Cloud Environment using Agent, *International Journal of Advanced Research in Computer Science and Software Engineering*, 3 (2013), No. 7, 1006 - 1010

Authors: Prof. Valeriy Rogoza, Head of Department of Computer Simulation and Information Protection, E-mail: wrogoza@wi.zut.edu.pl; Ph.D. Student, Michał Zabłocki, E-mail: mzablocki@wi.zut.edu.pl; Department of Information Technologies, Westpomeranian Technological University in Szczecin, ul. Żołnierska 52, 71-210 Szczecin, Poland