Reasoning on High Performance Computing Resources

An Urgent Computing Scenario

Axel Tenschert, Pierre Gilet
Service Management & Business Processes (SANE)
HLRS - High Performance Computing Center Stuttgart
70569 Stuttgart, Germany
E-mail: {tenschert, gilet}@hlrs.de

***Abstract* — The emergent growing amount of available information and data in the last decade has led to very large data stocks that express specific knowledge. This knowledge can be stored in ontologies. Reasoning strategies are then required to deal with a huge amount of data even if the allotted time frame to perform this task is restricted. This work covers the research issue of performing a time-wise restricted reasoning by means of ontologies. The presented approach is suitable for processing a reasoning strategy on high performance computing resources by considering a short time window and offering a solution for a quick allocation of required resources.**

Keywords - Ontology Matching; Reasoning; High Performance Computing; Resource Allocation

I. INTRODUCTION

The work described in this paper presents two research topics that need to be considered for solving the challenge of reasoning in a High Performance Computing (HPC) environment. Both research topics are explained in the next sections.

When thinking of an end user that performs a reasoning task with ontologies that are adequate for his/her needs it has to be taken into account that this end user might not be an expert in HPC infrastructures. This leads to the challenge of supporting the end user in a user friendly and time saving way in order to deal with the fact that time is short. At the present time, the allocation of computing resources at HLRS [1] is performed with a high level of effort and many human interactions involving an IT expert at HLRS that has to perform lots of manual steps. This manual workflow of computing resource allocation is time consuming. And if the end user is not an expert in HPC infrastructures the manual workflow is slowed further down. It is to be noted that HLRS is a federal high performance computing centre providing access to its HPC resources to researchers in Germany and Europe [2] and is currently extending its IT infrastructure through the acquisition of a new Cray XE6 platform in 2011 [3]. The move to production of that new supercomputer will increase the customer base of HLRS, which could comprise members of the traditional HPC community but also of non-HPC communities having only very few knowledge of HPC infrastructures. Hence, the need to support end users belonging to a non HPC community such as the semantic community is growing. To this end, the manual workflow solution is not an adequate option anymore.

Furthermore, a reasoning strategy is required that allows ontology matching over HPC resources. According to Ramesh and Gnanasekaran [4], the overall goal of an ontology matching is a merge of ontologies in order to create a new single terminology that can be used for reasoning purposes. The merge is an organization and reusage of concepts found in the source ontologies. This approach is used within this work to perform a matching between a set of ontologies in order to develop a resulting merged ontology. The resulting merged ontology is actually an ontology initially selected among the ontologies given as input that has been assigned the highest level of priority and is therefore called the priority ontology in the next sections. The merge process enriches that priority ontology with additional concepts, thereby leading to the generation of the resulting merged ontology. Also, the matching strategy considers similarities between the matched terms of the ontologies thanks to the use of a similarity value. This approach is also promoted by Pirró and Euzenat [5]. In their work, they describe a whole framework for the use of similarities in semantics.

The aim of this paper is thus to present an approach for designing a workflow going from allocating HPC resources up to performing a reasoning task in a merged ontology by means of the allocated computing resources. This publication is based on the ongoing PhD thesis written by Mr. Tenschert, and details about implementations and a result validation will be presented in future publications.

This paper gives an introduction to the described research field and related problems (I.), presents current research activities and challenges (II.), demonstrates a related use case scenario (III.), proposes a novel workflow for the given problem (IV.) and finally concludes with an outlook at future developments (V.).

II. CURRENT RESEARCH ACTIVITIES AND CHALLENGES

Nowadays, strategies for computing resource allocation and reservation in the HPC domain as well as reasoning strategies are available. However, the requirements and constraints imposed within an HPC environment are quite complex and specific to concrete use case scenarios, and reasoning strategies often need to deal with high amounts of

 ISBN: 978-1-61208-162-5

data also in a scenario where time restriction plays a prominent role.

The reservation and allocation of computing resources has been a research topic for HPC environments as well as for issues dealing with SLA (service level agreement) management and SLA lifecycles. The Grid Resource Allocation Agreement Protocol Working Group (GRAAP-WG) [15] offers solutions addressing this issue via the development of the Web Service Agreement specification (WS-Agreement) [7] allowing the creation of SLAs defining guarantee and service terms for the allocation of resources between two parties such as a service provider and a consumer. However, the use of HPC resources gives rise to the challenge of having to deal with very detailed and specific guarantee and service terms in an SLA. This results from the management of a specific IT infrastructure requiring very precise knowledge about the computing resources. This issue brings about the question of how to create a very specific SLA with specifications relating to HPC.

When thinking about reasoning strategies out of an information data pool - regardless whether the data is stored as text documents, graphics or visualized models - integration of information is of interest. The management of information by means of integration techniques is described by Lembo et al. [9] with a general formula (1).

$$<G, S, M> \quad (1)$$

- G is a global schema expressed in the global language L_G with the alphabet A_G. L_G determines the expressiveness allowed for specifying G;
- S is a set of local schemas modeled in the source language L_S with the alphabet A_S. A_S determines the set of defined constraints. A_S is disjoint from A_G;
- M is the mapping of G and S.

The presented formula for data integration is used for information retrieval approaches consisting in receiving targeted data and enhancing afterwards a source data set with the new available data. Such an approach is presented by Su et al. [10] for identifying an ontology matching strategy that makes use of a target ontology and a source ontology in order to enhance the source ontology.

Further, vector based techniques for dealing with word ambiguity are relevant for reasoning approaches in order to ensure the correct use of a term by validating its meaning. For instance, noteworthy vector based techniques are the latent semantic analysis (LSA) described by Landauer et al. [11] and random indexing described by Karlgren et al. [12], Chatterjee et al. [13] or Sahlgren [14].

Regarding reasoning, one can consider the research field of ontology matching as relevant, too. In this context, Zhang [16] describes the advantages of using ontologies and the semantic web for representing information. Gal et al. [17] discuss the need for ontology matching using matching of concepts with the aim to describe the meaning of data by considering heterogeneous distributed data sources and by also considering uncertainties in ontologies. Additionally, Huang et al. [18] give an overview of the use of ontologies relating to bioinformatics as formal knowledge representation models in order to offer knowledge to an expert. To this end, relevant ontology matching strategies are required for providing an expert with knowledge represented in the ontologies belonging to a various set of ontologies.

Due to the fact that various strategies for ontology matching have become more and more elaborate in the recent years, ontology matching approaches will be considered in this work as well.

III. URGENT REASONING SCENARIO

The use case scenario demonstrating the use of the resource allocation and reasoning workflow (RAAR-WF) is divided into two main parts:

1. Allocation of HPC resources,
2. Reasoning with a priority ontology.

In this scenario, an end user is in the need of receiving information split among various ontologies and the time frame for it is restricted. This means the results of the reasoning process have a validity window. Upon exceeding the time limit, the information becomes not required anymore and therefore has reached its point of validity. For instance, one could consider a biomedical scientist as an end user in need of receiving information (Fig. 1) for a patient or an urgent study before the point of validity is exceeded.

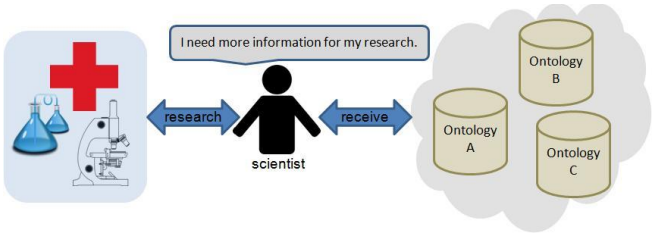

Figure 1: End User from the biomedical Research Area

The following subsections cover respectively the aforementioned two points (allocation of resources, and then reasoning) to ensure a good understanding of the whole scenario. The bringing together of both sections make up the RAAR-WF solution.

A. Allocation of HPC Resources

At present, the allocation of resources in an HPC environment (e.g., at HLRS) requires expert knowledge about IT infrastructures on the part of the end user. For instance, the end user has to know about the different architecture types, the characteristics of compute nodes, the installed software packages and tools, etc. Additionally, he/she must determine what configuration of computing resources best addresses the needs of the use case scenario. The challenge of allocating computing resources has increased due to the fact that within the HPC domain, Grid and Cloud infrastructures have become more and more complex. Considering a non expert end user in need of HPC, Grid or Cloud infrastructures, one can assume that the configuration of the requested computing resources will lead to a high level of effort for the end user and as well for the IT

 ISBN: 978-1-61208-162-5

expert who supports the end user. Besides, this process will be a time consuming one, too.

One can also note that the IT expert who supports the end user during the request for computing resources is also the person in charge for resource allocation. The IT expert performs a validity check of the end user request and then performs the resource allocation and reservation of the computing resources. The more requests are submitted to the IT expert, the more effort and time is obviously required to handle them. Timewise, in the urgent reasoning scenario, there is a high risk of exceeding the point of validity regarding the needed resources because of the general amount of requests for computing resources and the effort needed for supporting an end user who is not an expert in HPC, Grid or Cloud infrastructures.

B. Reasoning with a Priority Ontology

The foundation underlying the urgent reasoning scenario described in this work is a set of ontologies that fit the end users needs. In that scenario, the end user makes a selection of ontologies that most address the scenario requirements, e.g., ontologies about proteins or treatment modality.

One priority ontology needs to be defined first to make a clear distinction between the target and the source ontologies. One source ontology becomes the priority ontology and is then enhanced with the selection of target ontologies. Hence, the aim of the end user in this scenario is to improve one source ontology selected as the priority ontology in order to perform in the end a reasoning task with it. To this end, the end user can only choose one data source, the priority ontology, which will receive the required information. This strategy speeds up the overall reasoning procedure compared to a reasoning strategy that would force the end user to perform a reasoning task on the whole set of identified ontologies appropriate for the use case scenario.

The end user is provided with an automated merge of the selected ontologies that reduces the time for the end user to access the required information. However, one must ensure that the automated matching strategy produces information usable for the end user. To this purpose, a validity check during the matching process is required.

IV. IMPROVEMENTS THROUGH THE RAAR-WF

A. Improving the Allocation of HPC Resources

Thanks to the plugIT [6] approach, the process of validation and allocation of computing resources executed by the IT expert is enhanced in a way that the IT expert has only to approve the recommendations from the plugIT IT Socket. The general idea about the plugIT IT Socket is to support an end user who plays the role of a project applicant by means of the Online Proposal Submission (OPS) application. The OPS application supplies a form that the project applicant has to fill in to request access to computing resources. The project applicant thereby provides all the information required to run an automated assumption process that finds out the best HPC, Grid or Cloud configuration required for the scenario.

After receiving the recommendation from the plugIT IT Socket, the IT expert, who plays the role of the project approver, validates the recommendation and sends a notification with the recommendation back to the project applicant. The recommendation from the plugIT IT Socket is actually an SLA. For plugIT, the schema used as the foundation for the definition of the SLA XML structure is the WS-Agreement specification. However, in order to deal with the specific requirements relating to the HPC domain, additional elements were necessary. Those additional pieces were provided by the so-called WS-Agreement schema for HPC (HPC-WSAG [8]), which extends the WS-Agreement specification with HPC specific items. The recommendation proposed to the project approver is therefore an SLA offer based on the HPC-WSAG schema.

The plugIT IT Socket also requires information about the HPC site and its available infrastructure to make an assumption about the most fitting resource configuration to offer based on the input given by the project applicant. For this purpose, the IT infrastructure, the SLAs and special criteria for the SLAs are all represented as graphical models in an online repository accessible by the plugIT IT Socket. These models are designed by an IT expert, the infrastructure modeler, that has extensive knowledge about the existing HPC, Grid or Cloud environment and useful SLAs applicable to the available computing resources. The benefit of this approach is that the models are easily created by the infrastructure modeler when new computing resources are made available or if the current hardware changes. Thanks to this, the knowledge about the computing infrastructure can be shared among many IT experts. It is mapped in graphical models stored in the online repository. By means of this online repository, the plugIT IT Socket has enough information helping it produce SLA offers automatically.

The ability of the plugIT IT Socket to find out SLA offers based both on models stored in the online repository and on the input of the project applicant comes from the semantic kernel component of the plugIT IT Socket. The semantic kernel transforms the input of the project applicant into an ontology, and then compares it with the models representing the SLAs, SLA criteria and IT infrastructure that have been also transformed into model ontologies (MOs). Further, a domain ontology for HPC environments is used to support the transformation into MOs and the comparison between ontologies. This means that the functionality of the semantic kernel is twofold:

1. Transformation of the project applicant's input and models into ontologies and MOs,
2. Comparison of ontologies.

The picture of the resource allocation by means of the plugIT IT Socket is presented in Fig. 2.

 ISBN: 978-1-61208-162-5

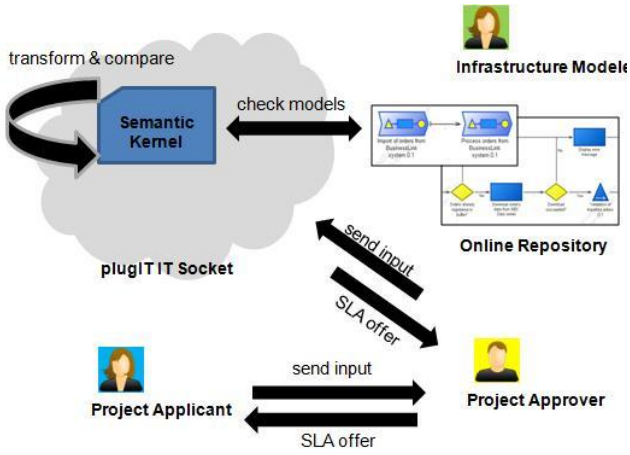

Figure 2: Resource Allocation Overview

The recommendation of SLAs to the project applicant via the plugIT IT Socket automates the process of resource allocation and makes it more efficient compared to the traditional manual workflow for resource allocation of HPC resources.

Additionally, concepts for the emerging cloud computing technology have been considered because of the fact that this new method for usage of distributed computing resources requires as well a clear strategy for resource allocation. The use of a cloud based approach provides the possibility to allocate needed computing resources for large scale data sets within a cloud testbed. One of the goals of the BonFIRE project is to develop a multi-site Cloud prototype. Within this scope, a cloud testbed is set up for research activities in the framework of the EU founded BonFIRE project [19]. The testbed will allow a large scale testing of research activities. This will be beneficial to the described work, especially considering the matching of ontologies made of very large data sets and matching procedure requiring vast amounts of computing power. The ongoing research and the results of this project, especially regarding the use of the cloud testbed, will influence the effective resource allocation as well.

B. Improving the Reasoning with a Priority Ontology

The use of a priority ontology enables reasoning on only one ontology that contains all the relevant information from a previously performed selection of ontologies. However, a strategy for matching the ontologies of the input set and further performing a validity check of the matching results is elaborated in this work with the aim to develop an ontology matching application. To perform adequate matching, a similarity value is created that expresses the level of accordance between matched entities. The matching workflow is performed in two major steps, the preparation and the execution, that are in turn subdivided.

- Preparation of similarity matching:
  a. Identification of relevant ontologies,
  b. Selection of relevant entities,
  c. Definition of the search space.
- Execution of similarity matching:
  a. Generation of the similarity value;
  b. Interpretation of the generated similarity value.

During the preparation phase, adequate ontologies are identified by the end user in order to create the set of target ontologies to be compared with the priority ontology. For this step the expertise of the end user is needed to decide which ontology to select. Then, the selection of the entities takes place to prepare the matching of the priority ontology entities to those of the target ontologies. For each entity of the priority ontology one matching iteration is performed. This means that the amount of matching iterations grows significantly with every entity of the priority ontology. However, the selection of the entities is based on the end user's expertise. This brings an additional possibility to specify the matching process in a very detailed fashion. The next step is the definition of the search space that establishes the number of neighboring entities that need to be taken into account for the matching of an entity. It is a necessary step in order to match the relations between entities. Due to the fact that the relation of one entity to its neighboring entities might not be similar in different ontologies, it is quite important to define the depth level, i.e., the search space, needed to assess the relations between entities in different ontologies. The deeper the search space is defined, the more numerous matching processes are performed and the higher the cost becomes which is associated with the matching process.

In the next step, the execution phase covers first the generation of the similarity value defining the level of compliance between entities. The similarity value is created out of a set of different values generated by various similarity matching processes using parameters such as the features of the concepts and relations to the neighboring concepts. The number of considered neighboring concepts was defined previously in the search space definition step. The second step of the execution phase is the interpretation of the similarity value. The similarity value is used for merging entities into the priority ontology based on the expressed level of compliance of the matched entities. Therefore, a high similarity value leads to a high probability of similarity between entities. Nevertheless, a validity check of the matching results is still required.

C. Improving the Reasoning with a Validity Check

Vector based techniques provide a solution for comparing the matching results saved in the priority ontology with the contents of a text document related to the topic of the use case scenario. The text document is provided by the end user having expertise in the considered topic. The selected text is the validity check document (VCD) containing text about the topic of the use case relating to the contents of the priority ontology. The selected terms from the priority ontology are concept and feature names. These terms are compared with those found in the VCD. Through random indexing techniques, the occurrence of a term coming from the VCD is calculated to determine how frequent it is found in the priority ontology. The random indexing approach for performing the validity check is divided into phases.

 ISBN: 978-1-61208-162-5

The random indexing solution is based on word space approaches and is therefore applicable to the required validity check with the VCD. Using word spaces means creating a high dimensional vector space for words to further construct a statistical value used for the next vector space. In the urgent reasoning scenario, the words considered for the validity check are the terms generated by the matching approach. These terms are concept and feature names. Regarding the vector space constructed by means of the previous statistical value, this strategy works with the assumption that if a set of words continuously appears in a text within the same context, then the meaning of the words will remain the same. It becomes thus possible to validate the terms of the priority ontology with those of the VCD and check if they can even be found in the VCD. However, in order to make an assumption about the context of the terms, the validity check examines the related terms in the VCD and in the priority ontology as well with the aim to compare those related terms. The analysis of the term relations in the VCD is done by examining the occurrence of words in the same sentence while the same analysis for the priority ontology is done by examining the relations between the concepts. A matrix containing the occurrence of terms in the priority ontology and in the VCD can be then created. This matrix is used to validate if the relations in the priority ontology appear in the VCD as well. Nevertheless, word space approaches face the challenge of scalability and efficiency. The use of HPC resources addresses this challenge, but a more fine grained approach for dealing with this issue in order to reduce the amount of required computing resources is still recommended. To this end, the simple vector based word space approach is enhanced through the use of the vector based random indexing approach that creates models such as those produced by latent semantic analysis (LSA) approaches. Following this approach, an extensive co-occurrence matrix is created first and then a reduction of the co-occurrence matrix is performed which limits the size of said matrix. Within the reduction phase, vectors of terms put in a specific context that occur multiple times are aggregated to accumulated context vectors. This way, the random indexing approach reduces the amount of required computing resources to perform the validity check in the given period of time.

Still, vector based techniques run the risk of producing unusable results when the text documents for comparison do not fit well the specific scenario needs. This leads to the question of whether the use of a vector based approach for a validity check as described previously is really usable without requiring a high level of effort from an expert who needs to do a very precise selection of text documents fitting the use case scenario. At the time of writing this paper, the vector based approaches are evaluated as well as the strategy of performing a validity check by means of a comparison ontology. The comparison ontology is matched with the priority ontology with the aim to make a proof of confidence regarding the updated priority ontology.

D. The RAAR-WF

The RAAR-WF comprises the allocation of best fitting HPC resources and the reasoning with the priority ontology including the validity check. The whole process going from allocating the required computing resource to getting back the urgently needed information through reasoning on one priority ontology is performed by considering a restricted time frame. The RAAR-WF is represented in Fig.3. As shown there, three points of human intervention can be identified in the workflow during the following phases:

1. Request for computing resources,
2. Configuration of the matching process and definition of the needed ontology set,
3. Reasoning with the created priority ontology.

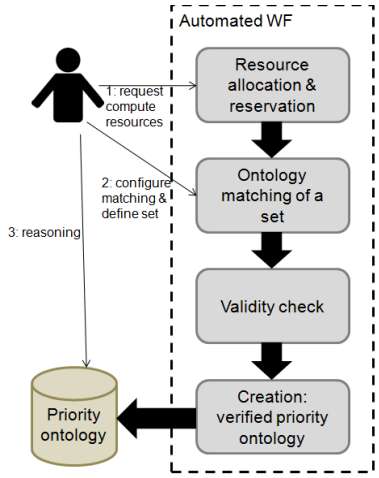

Figure 3: RAAR-WF

Beside the intervention of the end user, the most demanding tasks in terms of effort remain those performed in the automated part of the workflow. This includes the following steps:

1. Resource allocation and reservation: thanks to the use of the plugIT IT Socket, the resource allocation and reservation are provided in an automated fashion by the OPS application;
2. Ontology matching of a set: the selection of ontologies for the creation of the set is made by the end user, however the matching of the ontologies of the set is done automatically;
3. Validity check: the validity check guarantees the quality of the matching results thanks to an additional comparison of the matching results;
4. Creation of the verified ontology: after the validity check, the priority ontology is finally created based on the matching results and with the validity check taken into account.

The last step of the whole RAAR-WF workflow is the reasoning via the use of the priority ontology. The time constraint caused by the urgent computing scenario is also considered through the use of HPC, Grid or Cloud resources and a highly effective ontology matching method and

 ISBN: 978-1-61208-162-5

validation of matching results aiming at creating a priority ontology for reasoning.

V. CONCLUSIONS AND OUTLOOK

The described RAAR-WF includes a smart solution for automated HPC resource allocation involving graphical modelling and semantic processing that transforms models into ontologies and then compares the generated ontologies. The only steps to be taken over by human beings for the resource allocation and reservation are the creation of the necessary models and SLAs. The creation or update of models is only required if changes in the computing infrastructure have been made, such as the acquisition of a new cluster. The modelling effort is predictable and quite easy to make. It becomes thus possible to allocate and reserve HPC, Grid or Cloud resources within a short period of time. This complies with urgent computing cases such as the urgent reasoning scenario.

Furthermore, the reasoning strategy outlined in this document is performed on reserved computing resources which provide adequate computational power, and it makes use of a highly efficient ontology matching approach. The splitting of the ontology matching approach into a preparation phase and an execution phase offers a reliable matching solution whose output is checked by the validity check in order to guarantee reliable matching results. Also, the use of similarities increases further the reliability of the matching results. Since the result of the complete RAAR-WF is one single priority ontology, the task of the end user regarding the reasoning part is simplified because he/she has to consider only one ontology instead of having to cross check a set of many ontologies.

Regarding future developments, the validity check of the matching results offers the opportunity for further research dealing with the evaluation of various vector based word space approaches and diverse ontology matching strategies. The outcome of the work aiming at finding out which strategy has the highest probability of producing reliable matching results depends on the selected strategy as well as on the specific requirements and configurations of the use case scenario. Furthermore, the already mentioned BonFIRE project provides a cloud testbed for research activities. Therefore, the results obtained by that project will influence the resource allocation of computing resources in the HPC domain for this work. In the future, a cloud-like environment will be considered to allocate computing resources for the proposed ontology matching strategy.

ACKNOWLEDGMENT

This work has been supported by the plugIT project [6] and has been partly funded by the European Commission's ICT activity of the 7th Framework Programme under contract number 231430. This paper expresses the opinions of the authors and not necessarily those of the European Commission. The European Commission is not liable for any use that may be made of the information contained in this paper.

The BonFIRE project has received research funding from the European Commission under the Information Communication Technologies Programme (ICT), contract number 257386. The project has a consortium of more than 13 partners from industry and academia as well as non-profit organizations.

REFERENCES

[1] High Performance Computing Center Stuttgart (HLRS), web site: http://www.hlrs.de/ (last accessed: July 22, 2011)

[2] HPC Europa, web site: http://www.hpc-europa.org/ (last access: July 22, 2011)

[3] Cray Wins Supercomputer Contract From the University of Stuttgart Valued at More Than $60 Million, http://investors.cray.com/phoenix.zhtml?c=98390&p=irol-newsArticle&ID=1486975&highlight (last accessed: July 22, 2011)

[4] C. Ramesh and A. Gnanasekaran, "Methodology Based Survey on Ontology Management", International Journal of Computer Sciences & Engineering Survey (IJCSES), vol. 1, no. 1, 2010

[5] G. Pirró and J. Euzenat, "A Semantic Similarity Framework Exploiting Multiple Parts-of Speech", Proceedings OTM, INRIA Grenoble Rhône-Alpes & LIG, 2010

[6] plugIT project, web site: http://plug-it.org (last accessed: July 22, 2011)

[7] Web Service Agreement Specification (WS-Agreement): http://www.ogf.org/documents/GFD.107.pdf (last accessed: July 22, 2011)

[8] B. Koller, Enhanced SLA Management in the High Performance Computing Domain, PhD Thesis, 2010

[9] D. Lembo, M. Lenzerini, and R. Rosati, Review on models and systems for information integration, Universita di Roma La Sapienza, 2002

[10] X. Su and J. A. Gulla, An information retrieval approach to ontology mapping. Data & Knowledge Engineering, vol. 58, pp. 47-69, 2006

[11] T. K. Landauer, P. W. Foltz, and D. Laham, An Introduction to Latent Semantic Analysis, Discourse Processes, vol. 25, pp. 259-284, 1998

[12] J. Karlgren and M. Sahlgren, "From Words to Understanding", in Y. Uesaka, P. Kanerva, and H. Asoh, Foundations of Real-World Intelligence, pp. 294-308, 2001

[13] N. Chatterjee and S. Mohan, Discovering Word Senses from Text Using Random Indexing, Proceedings CICLing, 2008

[14] M. Sahlgren, An Introduction to Random Indexing. Proceedings 7th TKE, 2005

[15] The GRAAP Working Group, web site: http://forge.gridforum.org/projects/graap-wg (last accessed: July 22, 2011)

[16] J. Zhang, Ontology and the Semantic Web, Proceedings North American Symposium on Knowledge Organization, vol. 1, 2007

[17] A. Gal and P. Shvaiko, Advances in Web Semantics I, Lecture Notes in Computer Science, vol. 4891/2009, pp. 176-198, 2009

[18] J. Huang, D. Dou, L. He, J. Dang, and P. Hayes, Ontology-based knowledge discovery and sharing in bioinformatics and medical informatics: a brief survey, Proceedings 7th Conference on Fuzzy Systems and Knowledge Discovery, pp. 2203 – 2208, 2010

[19] BonFIRE project, web site: http://www.bonfire-project.eu/ (last accessed: July 22, 2011)

 ISBN: 978-1-61208-162-5