A Grid Infrastructure for Knowledge-based applications in Open and Dynamic Computing Environments

Edgar Jembere, Sibusiso S. Xulu and Mathew O. Adigun
Department of Computer Science
University of Zululand, P. O. Box X1001, KwaDlangezwa, 3886
Tel: +27 35 9026366, Fax: +27 35 9026569
email: ejembere@gmail.com, {ssxulu, madigun}@pan.uzulu.ac.za

Abstract- The drive towards net-centric applications that enable ubiquitous computing has led to the need for computing systems that operate and interact at knowledge level. In order for this vision to be realised, knowledge need to be modelled as a resource that is made available to computing systems on a global scale. To address this issue, we envisage a grid infrastructure that allows computing systems to interact at semantic level, and have capabilities to share, learn and evolve knowledge representations. Some solutions that seek to achieve this goal have been proposed. These include techniques emanating from the semantic web, semantic grid and knowledge grid efforts. However, none among these solutions have specifically addressed how uncertainty, which characterise such environments and evolution of the knowledge representations, can be addressed. In this paper we propose a Grid Infrastructure for Open and Dynamic Knowledge-Based Applications (GIODKA). GIODKA builds upon probabilistic knowledge representation to handle uncertainty and Belief Change Analysis (BCA) techniques to handle evolution of knowledge representations. In this paper, we discuss the design consideration and architecture of GIODKA. We also present the implementation details of the system. Finally, we discuss how the system can be used to support knowledge-based personalisation in Service Oriented Computing SOC environments.

Index Terms—Knowledge grid, Ontology Evolution, First Order Bayesian Logic, Semantic Web.

I. Introduction

The semantic web vision aims to drive the web to its full potential by making its content machine understandable. This should result in a smarter and more intelligent web, thereby enabling more efficient~~,~~ information retrieval, information integrations and more sensible interactions between human agents and computing systems. Grids are basically concerned with the design and development of the architecture of future internet. According to Goble and De Roure's [1] vision, this should support "on-demand virtual organisations" for coordinated resource sharing and problem sharing at a global scale. We argue that the grid provides the platform for the realisation of the semantic web vision. In this case, knowledge should be modelled as first class citizens among the resources available in the grid. This vision, which some have dubbed the World Wide Wisdom Web (WWWW), goes beyond the semantic grid vision. The Semantic Grid advocates for the description or annotation of the grid resources and services with well defined semantics [2]. The WWWW vision furthers this by availing the well defined semantics in the semantic grid as grid resources. Due to the openness and dynamicity of grids/web environment providing knowledge as a resource presents new challenges. Key challenges among these include uncertainty, inconsistency, complexity and dynamicity of the knowledge and its representations [3]. To address these challenges, we adopted the concept of Probabilistic ontologies [4], based on First Order Bayesian Logic (FOBL) to handle uncertainty. Techniques from Belief Change Analysis (BCA) [5] were adopted to handle and formalise ontology evolution issues. However, in this paper we focus on the design and implementation of a grid infrastructure that supports our solution. The Infrastructure serves as an enabler of knowledge-based net-centric applications.

There are basically three ways in which uncertainty management can be incorporated into ontologies and these are (i) uncertainty modelled in Data[6], (ii), directly extending ontologies by a probabilistic model [7], [8], [9] , and (iii) transformation of ontologies into a structure that handles probabilities [4], [10]. We found the transformation of ontologies into FOBL structures to be the best knowledge representation in open and dynamic computing environments [3]. This is due to the fact that Bayesian conditioning supports rational and objective knowledge evolution.

This paper is organised as follows: Section II discusses some related works. We focus on the works that seek to provide platform for knowledge sharing in open and dynamic computing environment. In Section III, we present the rationale and design requirements of the Grid Infrastructure for Open and Dynamic Knowledge-Based Applications (GIODKA) architecture. Section IV presents the GIODKA architecture. Implementation details of the architecture are given in Section V. In Section VI, we discuss personalisation in grid environments as a use case of the GIODKA architecture. Section VII concludes the paper.

II. Related Work

Figure 1 depicts the evolution of WWWWs or Knowledge grids as a confluence of techniques from knowledge engineering, Knowledge Discovery and Grid Computing. The highest level of the Semantic Web vision seeks to create an environment where computers and human beings can interact seamlessly. This is also the goal of WWWW. In order for this vision to be fulfilled, the grid complemented with the current trends in web technologies is viewed as the key enabler. Research efforts towards WWWW vision can be categorised into three Categories and these are: Semantic Grids [2], Data mining Grids [11], [12], [13], and Knowledge grids [14].

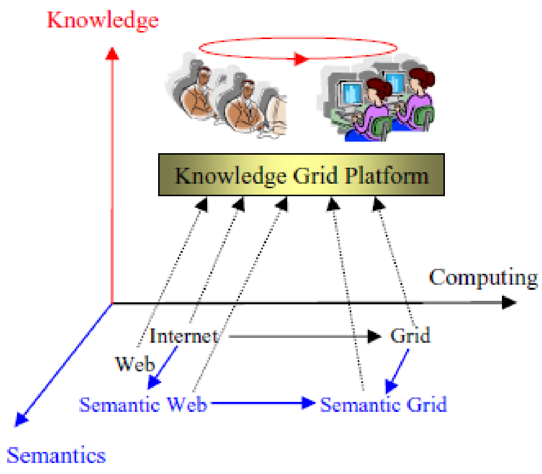

Figure 1:The evolution of WWWW (Adopted from [14])

De Roure et al [2] views the semantic grid as an open system, with a high degree of automation that supports flexible collaboration and computation on a global scale. The aim is to bridge the gap between the grid application development endeavours and the vision of e-science. This research endeavour [2] conceptualises the Grid computing infrastructure as consisting of three layers: the data/computation, information and the knowledge layer. The knowledge layer is concerned with how knowledge is acquired, used, retrieved, published and maintained in order for it to be used to achieve a particular goal. The belief is that the full potential of an intelligent grid can only be realised by exploiting the functionality and the capabilities provided by the knowledge layer services. This is on the premise that it is in the knowledge layer that the reasoning to seamlessly automate a number of actions and interaction takes place.

Research efforts towards the data mining grids provide high level knowledge discovery services as grid services. Over the past few years there has been quite a number of implementations of the data mining grids (e.g. [11], [12], and [13]). Most of them aim to provide scientists with tools to create and manage complex knowledge discovery applications composed as workflows that integrate data sets, mining tools, and computing and storage resources. The main issues addressed in data mining grids are:

- Synthesizing useful and usable knowledge from data.
- Leveraging the Grid infrastructure to perform sophisticated data-intensive large scale computation

Key features of these grid-based data mining systems include, but are not limited to: flexibility, extensibility, scalability, efficiency, conceptual simplicity, and ease of use. Though some of the infrastructural services and the algorithms proposed can be adopted in what we seek to achieve, none of them seeks to achieve automatic knowledge discovery and the use of the discovered knowledge to build intelligence in the infrastructure.

A treatise of the Knowledge grid vision is given in the works from the Knowledge Grid Research Centre (KGRC) [15]. Zhuge defines the knowledge grid as *"a sustainable human machine interconnection environment that enables people or agents to effectively generate, capture, publish, share, manage and promote knowledge, to process any type of resource through machines, and to transform resources from one form to another"*[14]. The Knowledge grid concept defined above is very wide. It includes epistemology (the nature, scope and source of knowledge) and ontology. It exploits social, biological and economic principles and adopts next-generation web technologies. Apart from the extensive coverage of the knowledge grid concepts and development methodology, the KGRC research efforts use Semantic Link Networks (SLN) for its knowledge representation. SLNs can not inherently model uncertainty. Uncertainty management is achieved by extension of the SLNs with probabilistic weights on the axioms in the SLN [16]. Hence our work is, to the best of our knowledge, the first research effort towards a knowledge grid that uses FOBL ontologies for knowledge representation.

III. The GIODKA Architecture

A. GIODKA Design Requirements

Different from the other efforts towards the WWWW discussed above, the GIODKA is specifically meant to provide a platform for knowledge-based application, with knowledge modelled as a first class resource in the environment. To achieve this end, we considered the following design requirements:

i. **Semantic Interaction Environment**: In order to fully support knowledge based application, the components of the infrastructure need to interact with one another semantically, in order to enhance inter-component interoperability. Semantic grid technologies such as metadata and ontologies can be exploited as the basis for resource description and discovery, but due to uncertainty, incompleteness and inconsistencies that characterise the environment [4], [6], there is a need for a knowledge representation that handles uncertainty in a principled way.

ii. **Automatic distributed knowledge discovery, search, learning and evolution**: The knowledge-based application need to search and discover relevant knowledge~~,~~ which might be stored in different nodes in the grid. Based on the context of the instantiated objects, the current state of the knowledge is inferred. Owing to the fact that knowledge, society and systems are the functions of time [4], there is a need for evolving the knowledge representation as the computing environment changes with time.

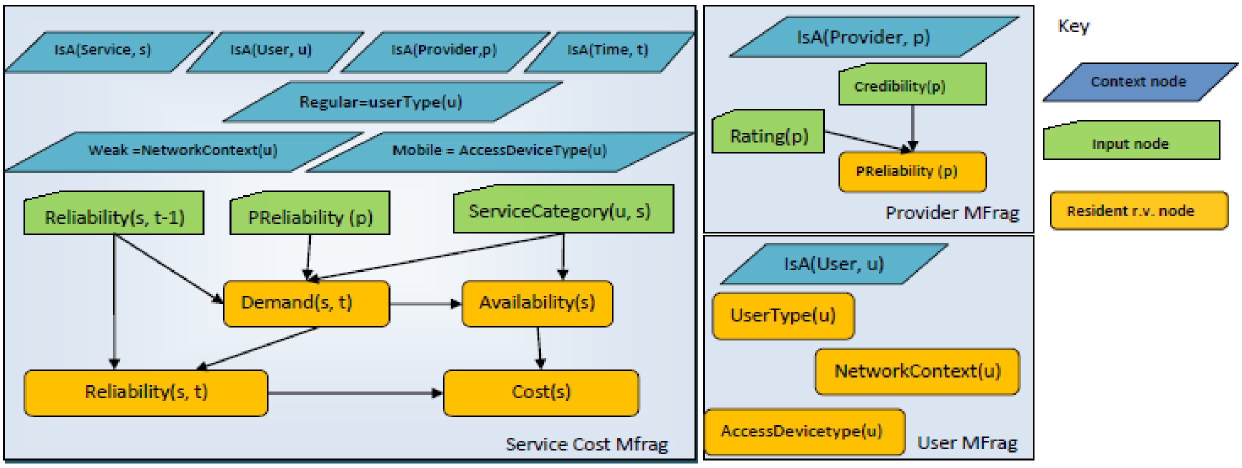

Figure 2: An Example MTheory in MEBN logic

iii. **Uniform resource organisation**: the resources in the WWWW are not limited to programs and computational resources. They also include knowledge, and knowledge management services. To ensure flexible management of these resources a mechanism for uniform organisation is required. Taking inspiration from [17], we adopt the Resource Space Model defined by Zhuge [18]. This model defines a Resource Space Model as a three-dimensional space, R = (Category, Level, and Location) where Category denotes the category of resources, Level denotes at which level the resource is applied (infrastructure, domain, and application) and Location gives the URL of the resources. Each entry in the resource space should be defined as duple (ID, SD) where ID is a unique identifier of an entity and SD is the Semantic Description annotated by ontologies.

iv. **Modular organisation of the software component**: The components in the infrastructure should be Web Services Resource framework (WSRF) compliant. This implies that each component will be implemented as an infrastructural service that is autonomous.

B. Knowledge Representation

Uncertainty, inconsistency and complexity are typical characteristics of open and dynamic computing environments which cannot be avoided [4], [6], [7]. They need to be captured at knowledge representation level. Classical ontology modelling approaches do no cater for these characteristics. This calls for development of mechanisms that enable next generation computing environments to account for these characteristics in representing and reasoning upon the knowledge in their ontological Knowledge Bases (KBs). For this purpose we adopt Multi-Entity Bayesian Network (MEBN) logic [19]. MEBN is a First Order Bayesian logic language, for knowledge representation. It views a domain as a set of related random variables defined over a set of entities in the domain. Knowledge is expressed using some constructs known as MFrags, each of which represents a joint distribution of a group of related random variables. A collection of MFrags that satisfy some consistency constraints, ensuring the existence of a unique joint distribution over all the random variables in a given domain, known as an MTheory, captures the knowledge about a domain. For details on MEBN logic the reader is referred to [19]. Figure 2 shows a simple MTheory example for representing knowledge about a service cost. The knowledge represented in Figure 2 communicates that given that the user is a non-privileged user (regular user), under weak network connectivity and that the service is being accessed by a mobile device, the cost of a service depends on its availability and reliability (N.B. in some other contexts the service cost might only depend on its demand, demand and availability, or demand and reliability, etc). A Probability distribution over each and every resident random variable given its parents is represented as a conditional probability table (CPT). In Figure 2, the CPTs were left out in order not the clutter the diagram.

Apart from handling uncertainty in principled way MEBN logic provides a natural way of handling context-awareness, which is very important for reducing complexity of knowledge inferences in complex domain and for increasing recall and precision of knowledge searches. The fact that the knowledge representation language is based on Bayesian logic also provides MEBN with the ability of updating or revising the knowledge representations as observational evidence accrues. In the next section we discuss how this can be achieved in a rational and objective way adhering to Belief Change Analysis principles.

C. Knowledge Evolution

From the definition of ontology evolution given in [20]:*"Ontology evolution is the process of modifying ontology in response to a change in the domain (first kind of change) or its conceptualisation (second type of change)"*, it can be deduced that ontology evolution involves both ontology update (first kind of change) and revision (second kind of change). Thus, in ontology evolution, like in belief change, it can be argued that ontology change is necessitated by two factors, namely: (i) the domain may have changed, and (ii) the system's knowledge about the domain may be simply mistaken or incomplete. In belief change literature, changes necessitated by changes in the world are addressed by Belief Update while Belief Revision addresses changes necessitated by mistaken or incomplete knowledge about the world.

Belief revision is used to effect belief changes in static domains. In such domains, belief changes are only necessitated by the fact that the system's beliefs about the world are mistaken or incomplete since there will be no changes in the domain. AGM theory [21] is the flagship implementation of belief revision. AGM takes a coherence view to knowledge change, which advocates that credibility of an axiom depends on how coherent it is with other axioms in the KB. The premise behind the coherence view is, if a belief revision operation calls for some beliefs to be retracted (in order to keep the KB consistent after the belief change operation), the relative entrenchment of a belief depends on how coherent the belief is with other beliefs in the belief set.

Belief update is used for belief change in dynamic domain. The work by Katsuno and Mendelzon [22], known as the KM theory is the most popular theory for belief update. The KM theory uses the event model to capture knowledge changes in a dynamic world. The KM theory takes observation of a new axiom as evidence that there have been a change in the domain. The assumption here is the existence of new axioms is the least of what could have possibly changed in the domain. In order to capture what have transpired the KM theory models the most plausible transition of a given world, *w*, into a world, *v*, that satisfies the observation of *A*. A set of pre-orders that are reflexive and transitive are defined over a set of worlds. A pre-order $u \leq_w v$ implies that *u* is at least as plausible a relative change relative to *w* as *v*. In KM, unlike in AGM, once an inconsistency is introduced in the KB there is no way to eliminate it using the update operator because update assumes that the world has changed. This implies that the update operator does not allow an observation to force revision of the system's beliefs about the state of the domain prior to the observation. In short the KM theory can be considered to be concerned with states of a changing world at given time stamps and the transition between these states necessitated by an event model, where as the AGM is concerned with belief states and their possible revision necessitated by the epistemic entrenchment of axioms in the belief set. A lot of effort has gone into generalising the KM theory to cater for revision (e.g. [23]). While such approaches can cater for both belief revision and update, the resultant KB is simply a belief set with no updated epistemic states. The resultant belief state provides no guidance for changes in belief due to subsequent changes. This makes such approaches less promising candidates for automatic ontology evolution, in cases where there is no quantitative measure of plausibility. A quantitative version of this approach serves the needs of what we seek to achieve.

Generally, belief change is guided by the following six (6) principles [5] (i) principle of minimal change, (ii) principle of consistency maintenance, (iii) principle of fairness, (iv) principle of adequacy of representation, (v) principle of primacy of new information (vi) principle of irrelevance of syntax. Our focus will be on the first three principles, minimal change, consistency maintenance, and fairness. We consider how these principles can be used to ensure rational ontology evolution, with both ontology update and revision, in FOBL ontologies. These principles then led to the design of the ontology evolution algorithms.

MEBN logic defines two types of knowledge constructs it represents and these are: the Generative Knowledge and Findings knowledge. A Generative MTheory summarises statistical regularities that characterise the domain. This knowledge consists of the knowledge on the dependencies between the variables (Structural knowledge) in the domain and the associated statistical regularities (parametric Knowledge). The Findings knowledge is the knowledge about the instances of the entities in the domain. Findings provide a mechanism for incorporating observations into MTheories. From the foregoing discussion it can be seen that ontology evolution is defined only in generative MTheories, both for structural and parametric knowledge with the findings providing the observations needed to evolve the knowledge representations.

For evolving structural knowledge we use a search and score Bayesian approach to structure learning. The structure learning takes a prior network, which represent the current state of knowledge, and the observations as input. It then searches through for neighbouring networks, using some mutation operators. The mutation operators used are *edge deletion*, *edge addition* and *edge reversal*. The search criterion returns network structure with a local optimal score. Moving one step at a time from a prior network ensures that we will have minimal change. Discovery of a locally optimal solution ensures fairness (repeatability of the solution regardless of the search path taken from the prior network). A cycle checker is used to ensure that the optimal network structure found has no cycles, thereby maintaining consistency

Parametric learning in FOBL is still a challenge, because it requires instantiation of the instance of the entities in a given domain. In our case we will instantiate all the observed entities to get the data to be used in the learning process. We adapted the generalised belief update approach [23] to its quantitative counterpart and the following formulas were derived to cater for both revision and update of generative parameters:

$$P_{t+1}(x_i) = \sum_{e \in E} \sum_{j=1}^{n-1} P_t(x_j) P(e/x_j) P(x_i/x_j, e) \qquad (1)$$

$$P^r_{t+1}(x_j) = \frac{P(o/x_j) P_{t+1}(x_j)}{\sum_{j=0}^{n-1} P(o/x_j) P_{t+1}(x_j)} \qquad (2)$$

where x_i is the state of a variable at time $t+1$, x_j is the state of the variable at time t in *Equation (1)* and any possible state of the variable in *Equation (2)*, *e* is any possible event in a given state, *o* is an observation. *Equation (1)* is used for update and *Equation (2)* is used to effect revision.The ontology change is a two step process: first it updates the distribution to form $P_{t+1}(x_i)$ using *Equation (1)*, and second it conditions this distribution on the observation o to obtain $P^{r}_{t+1}(x_j)$. If the domain is static, the only possible event is the null event (the event that nothing happens/changes), *n*, which is sure in static domain, that is $P(n/x_j) = 1$. The probability of the variable moving from one state to another given a null event has occurred is zero for all $x_i \neq x_j$ and 1 for $x_i = x_j$. Substituting these probabilities in *Equation (1)* gives $P_{t+1}(x_j) = P_t(x_j)$ in static domains. Thus the algorithm only effects *Equation (2)* in static domains. In dynamic domains, the algorithm first do epistemic update of the parameters using the event model and then revise the parameters on the basis of the observations.

IV. Design of the Architecture

The GIODKA architecture builds on the existing grid platforms compliant to the Web Services Resource Framework (WSRF). The architecture aims to provide layer of Knowledge Management services, leveraging upon the semantic grid layer services. Figure 3 presents the Service Oriented architecture of the GIODKA, with four Layers; the grid resource Layer, basic Grid layer, Semantic grid layer and the Knowledge grid layer.

The basic grid Layer serves as the infrastructure for searching, discovering, invocation and management of the Grid resources. The grid resources include hardware, files and directories, data, ontologies and Knowledge management tools. The knowledge resources are stored as ontologies based on both First Order logic (OWL) and First Order Bayesian Logic (Pr-OWL). The Knowledge management tools include the algorithms for rational and objective evolution FOBL ontologies, we discussed in Section III, and knowledge inferences algorithms.

The Semantic grid layer provides the services to annotate grid resources with explicit semantics such as meta-data access, semantic service discovery and ontology matching services. The Knowledge grid services are the key contribution of this work. They include services for performing inferences, and updating/revising the Knowledge bases. We categorise these services into two; basic Knowledge Grid services and Knowledge Management services. The basic Knowledge Grid services enable the Knowledge Management services to access resources that are needed for ontology evolution and knowledge inferences. These services include the Knowledge Access Service, which enables Knowledge Management services to access the KBs to be used. The Data Access Service uses the semantic grid information service and the Data Management services of the basic grid to access the data needed by the Knowledge Managements services. The Algorithm Access Service enables the Knowledge Management services to access the algorithms they need to learn new knowledge, revise or update existing knowledge bases, or perform knowledge inferences. The Knowledge Management services implements the process that enable knowledge evolution and inferences.

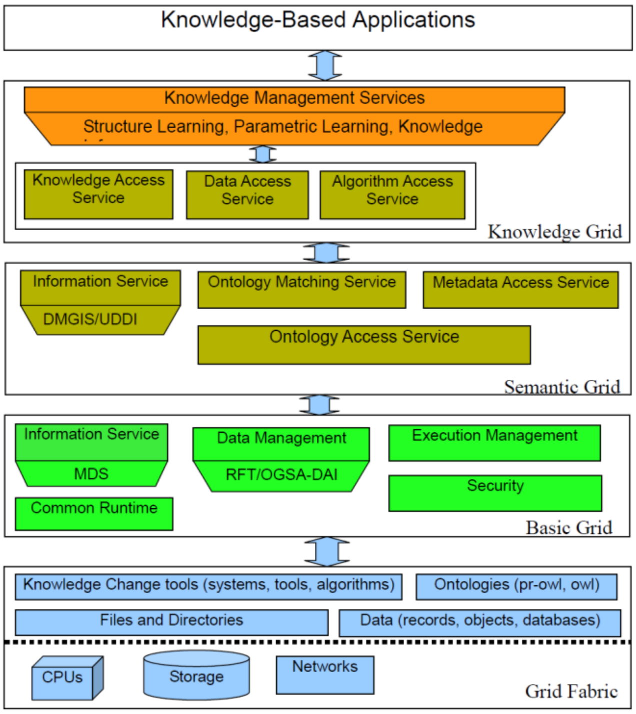

Figure 3: The GIODKA Architecture

V. Implementation

Implementation of the GIODKA architecture leverages on the existing frameworks for supporting Grid environments. Owing to fact that the implementation of GIODKA needs to be WSRF compliant, we decided to use the Globus Toolkit 4 (GT4) [24] as our basic grid layer. The GT4 offers all the infrastructural services in this layer. Following the mechanism for uniform resource management proposed by Zhuge [18] and used in KIGBRS [16], the metadata of the resources are represented in the form of an XML file (based on the Resource Space Model discussed in Section III). The discovery of resources is performed by resolving this XML document after querying the information service through the Metadata Access Service. The grid resources are made available to the grid through the Globus's Monitoring and Discovery System (MDS) service. Each node in the Grid publishes the resources it has with the MDS. The MDS will then facilitate the discovery of the published resources. The semantic grid services and the basic knowledge grid services are performed by various grid services which are all registered with the MDS.

Since MEBN [18] is adopted as the FOBL for knowledge representation, unbbayes [25] was the default API for implementing Knowledge Management Services. This also resulted in Pr-OWL[4] being used as the Probabilistic ontology language. Powerloom is used for the knowledge base since unbbayes uses powerloom as its default knowledge base. For Inferences we service-enabled the Laskey inference algorithm implemented in unbbayes. For Structure learning the Bayesian Network Inference with Java Objects (BANJO) API [26] was used. This is because (i) it allows prior structure knowledge to be used in the structure learning process, (ii) it can learn Dynamic Bayesian structures, which are a necessity for our ontology evolution solution to cater for ontology update, and (iii) provides the mutation operators we require in the structure leaning process to ensure minimal change. We service-enabled the BANJO structure learning algorithms for the algorithms to be available as grid services. For parametric learning, the algorithm we discussed in Section III. C was used.

VI. USE CASE

We use personalisation in service oriented environment to demonstrate applicability of GIODKA. We considered a next generation grid environment, which based on a user request for a composite application the grid infrastructure can automatically discover, select and bind to all the necessary grid services that make up an application that satisfies the user's requirements and preferences, and optimises the user's utility. Such a task will require that the system have the user knowledge, grid services knowledge, domain knowledge, and personalisation function knowledge [17]. This knowledge as well as various tools and software needed to meet the user's request will be scattered in different nodes in the grid/web. This implies that the knowledge has to be discovered, fused and used in the composition of the application to best serve the user. But because of the uncertainty, complexity and vastness of the environment context information is supposed to be used to narrow down the knowledge space, so as to get the knowledge with increased certainty. The resulting knowledge is then used by the application to best serve the user.

VII. CONCLUSION

In this paper we have presented the GIODKA architecture for supporting knowledge-based application in open and dynamic computing environment. The architecture provides a semantic interaction environment for software components in the environment and caters for uncertainty and dynamicity in the environment by adopting a probabilistic knowledge representation and proposing an ontology evolution solution. Implementation of the infrastructure leverages on the techniques emanating from solutions towards the semantic grid. A use case of personalisation in SOA environment is given to demonstrate the applicability of the GIODKA architecture.

Edgar Jembere received his MSc degree in 2008 from the University of Zululand and is presently studying towards his PhD degree at the same institution. His research interests include Knowledge Engineering, Data mining, Grid computing and Service Oriented Computing Technologies.

References

1. Goble, C. A., and De Roure, D., The Grid: an Application of the Semantic Web. ACM SIGMOD Record 31(4): 65-70 (2002).
2. De Roure D., Jennings N. R., and Shadbolt N., The Semantic Grid: A Future e-Science Infrastructure., *Grid Computing: Making the Global Infrastructure a Reality*, F. Berman, A.J.G. Hey, and G. Fox, eds., John Wiley & Sons, 2003, pp. 437–470.(2003)
3. Jembere E., Xulu S. S., Adigun M.O., Automatic Ontology Evolution in Open and Dynamic Computing Environments; J.-S. Pan, S.-M. Chen, and N.T. Nguyen (Eds.): ICCCI 2010, Part III, LNAI 6423, pp. 122–132, Springer, Heidelberg (2010)
4. Costa, P.C.G.D., Bayesian Semantics for the Semantic Web. Doctoral Dissertation. Department of Systems Engineering and Operations Research, George Mason University: Fairfax, VA, USA (2005)
5. Flouris, G., On Belief Change and Ontology Evolution, PhD thesis, University of Crete, Greece (2006), unpublished.
6. Haase P.Völker J., Ontology Learning and Reasoning - Dealing with Uncertainty and Inconsistency. ISWC-URSW 2005: 45-55, (2005)
7. Scharrenbach T., Bernstein A., "On the Evolution of Ontologies using Probabilistic Description Logics", Proceedings of the First ESWC Workshop on Inductive Reasoning and Machine Learning on the Semantic Web, Editor(s): Claudia d'Amato, Nicola Fanizzi, Marko Grobelnik, Agnieszka Lawrynowicz, Vojtech Svátek, June 2009, CEUR Workshop Proceedings (2009)
8. Lukasiewicz T.: Probabilistic Default Reasoning with Conditional Constraints. Ann. Math. Artif. Intell. 34(1-3): 35-88 (2002).
9. Domingos P., Kok S., Lowd D., Poon H., Richardson M., Singla P.,: Markov Logic. Probabilistic Inductive Logic Programming 2008, 92-117 (2008)
10. Yang Y, Calmet J: OntoBayes: An Ontology-Driven Uncertainty Model. CIMCA/IAWTIC 2005: 457-463 (2005)
11. Cannataro M., Knowledge-based Services for the Next Generation Grid, *In Proceedings of the 13th IEEE International Workshops on Enabling Technologies: Infrastructure for Collaborative Enterprises (WET ICE'04)* (2004)
12. Brezany P., Janciak I. Tjoa A. M., GridMiner: A fundamental infrastructure for building intelligent grid systems, *In proceedings of the 2005 IEEE/WIC/ACM International Conference on Web Intelligence, WI'05*, (2005).
13. Congiusta A., Talia D., Trunfio P., Distributed data mining services leveraging WSRF, *Future Generation Computer Systems*, 23(1), pp 34–41 (2007).
14. Zhuge H., *The Knowledge Grid*, World Scientific Publishing Co., Singapore, (2004).
15. http://www.knowledgegrid.net/(Last accessed on 04 August 2011).
16. H.Zhuge., Autonomous semantic link networking model for the Knowledge Grid, *Concurrency and Computation: Practice and Experience*, 7(19) (2007)1065-1085.
17. Liu P., Nie G., Chen D., and fu Z., The knowledge grid based intelligent electronic commerce recommender systems, In Proceedings of IEEE International Conference on Service-Oriented Computing and Applications, Newport Beach, 2007, 223-232
18. Zhuge H., And Xing Y., Integrity Theory for Resource Space Model and Its Application, *in proc. Of the 6th International Conference on Web-Age Information Management (WAIM2005)*, Hangzhou, China, Oct.11-13, 2005., LNCS 3739, pp.8-24
19. Laskey, K. B., MEBN: A Logic for Open-World Probabilistic Reasoning. The Volnegau School of Information Technology and Engineering. George Mason University, Fairfax, VA, USA (2005)
20. Mazzieri M., Dragoni A. F., Ontology Revision as Non-Prioritized Belief Revision. ESOE 2007: 58-69 (2007)
21. Alchourron C., Gärdenfors P, and Makinson D. On the Logic of Theory Change: Partial Meet Contraction and Revision Functions. Journal of Symbolic Logic, 50:510-530, (1985)
22. Katsuno H., Mendelzon A. O., On the Difference between Updating a Knowledge Base and Revising It. KR 1991: 387-394 (1991)
23. Boutilier C., A Unified Model of Qualitative Belief Change: A Dynamical Systems Perspective. Artif. Intell. 98(1-2): 281-316 (1998).
24. http://globus.org/toolkit/downloads/4.2.1/(Last accessed on 7 February 2011)
25. http://unbbayes.sourceforge.net/ (Last accessed 7 February 2011)
26. http://www.cs.duke.edu/~amink/software/banjo/ (Last accessed on 07 February 2011)