

Business Requirements of Knowledge Management Ontology to Support a Software Deployment Process

Annette Lerine Steenkamp
steenkamp@ltu.edu
Lawrence Technological University
Southfield, MI 48075-1058 USA

Jehad Sabri Alomari
ahal8@yahoo.com
Lawrence Technological University
Southfield, MI 48075-1058 USA

Abstract

The software deployment environment is dynamic and complex. Changes to the deployment environment occur continuously, especially in a virtual environment. Managing the complexity of the software deployment process requires careful preparation of relevant knowledge. Business requirements have to be fulfilled and managed before starting the deployment process. Efficient and successful software deployment depends a lot on the knowledge acquired from the software environment and business requirements (technical and non-technical) in order to reduce deployment time and achieve successful deployment. Developing and utilizing an Ontology-based Knowledge Discovery and Management System (OKDMS), based on discovery and mapping of relevant data and information, has lead to a sound understanding of the complexities of software deployment and potential support that may be provided by utilizing KM.

Keyword: Software Deployment, Knowledge Engineering, Ontology development, Knowledge Management

1. INTRODUCTION

Software is increasingly supporting the daily activities of technologically advanced societies to achieve their challenging goals. As a result software deployment has become a daily task in many organizations. Software deployment is part of software engineering. It is performed during the software deployment phase of the software development life cycle. The software deployment environment is dynamic and complex, since changes to the deployment

environment occur continuously, especially in virtual environments. It is a complex process which includes all the activities from the end of development to delivery and execution of applications of the end-user system.

The primary purpose of this research is to gain a sound understanding of the complexities of software deployment, and the potential support that may be provided by utilizing knowledge management (KM), as well as presenting a conceptual solution of the Ontology-based Knowledge Discovery and Management System (OKDMS) which is

capable of automating the management process of software deployment. The research of the software engineering community has focused in great part on software development. The literature review that was conducted has revealed that software deployment warrants rigorous research, and is in fact a new research field. The software deployment process starts from the point where software is available to be installed in the target environment to the point where the software can be utilized. The post-development phases such as deployment and maintenance are becoming much more important than before because of the rapid and dynamic changes in the software environment, the impact of the run-time environment, and cost of ownership and return-on-investment considerations.

Software deployment knowledge is a valuable asset within any organization that should be utilized and managed to make the deployment process less complex and less time consuming. Building a knowledge management system that addresses the complexity of the process requires two types of knowledge: Knowledge about domain facts such as objects, relations, and so on (domain knowledge); knowledge about achieving goals using domain knowledge (problem-solving knowledge) (Chandrasekaran et al, 1998). The ontology building process is often a collaborative effort between the participants in order to share their knowledge and common understanding of the ontology representation (Das et al, 2001).

This knowledge asset concerns people, relationships, skills, ideas, and experiences, and interacts with available information (Briggs 2000). Ontology is defined by Gruber (1993) as an explicit specification of a conceptualization. Ontology can capture the relationship between concepts including tacit and explicit knowledge in the model. The integration of knowledge from different sources requires a new look at KM concepts (Fischer and Ostwald, 2001). With ontology, we can define KM concepts to support software deployment interoperation.

This paper focuses on the business perspective of KM ontology to support software deployment. Section 2 outlines the importance and benefits of enabling ontology to manage the software deployment process. Section 3 focuses on

the discovery and representation of data and information to create knowledge. It also presents an Ontology-based conceptual solution with focus on deployment environment discovery and preparation of the business requirements. Section 4 presents an ontology-based conceptual solution to manage business requirements of the software deployment process. Section 5 summarizes the future work of our ongoing research. Conclusions drawn based on this research project are discussed in Section 6, as well as some ideas for the remaining part of the project.

2. BENEFITS OF ONTOLOGY TO MANAGE SOFTWARE DEPLOYMENT

A knowledge map is defined as "The intellectual infrastructure for knowledge management initiatives. The basis for it consists of multiple taxonomies for content repositories, dynamic categorization of people, their expertise, and the communities they belong to, and finally a set of taxonomies for the variety of tasks that are performed within and by the company's communities." (Anon, 2003)

Knowledge mapping techniques must be effective, efficient, and relevant part of an integrated approach of mapping knowledge to solve a defined problem. Knowledge mapping techniques such as Information Flow Analysis (Lutters, 2001), Social Network Analysis (Pollock, 2002), and Process Method of Analysis (USAID, 2003) are supported by tools to gather, retrieve and evolution data (Vestal, 2005).

Dealing with the flood of available information by enabling information retrieval techniques alone and assuming that people can understand their information needs is not sufficient (Belkin, 2000). Even if a search for knowledge were successful to solve a problem, some knowledge to understand its potential relevance is still needed (Fischer et al, 1992). Applying ontology based techniques to software deployment has the following benefits:

- Enable Information understanding, knowledge reuse, and knowledge sharing between domain experts and other participants in the software deployment process.
- Support standardization of basic application and related domain knowledge and components.

- Support information infrastructure management and knowledge discovery of the software deployment process.
- Support business requirements related to the deployment process.
- Improves software delivery time, increase return on investment to help organizations overcome their business challenges and successfully achieve deployment goals.

3. KNOWLEDGE DISCOVERY AND REPRESENTATION FOR SOFTWARE DEPLOYMENT

The "Knowledge age" has seen a rapid increase of the amount of data and information in various forms. This is also the case for software engineering, and specifically software deployment. This research proposes that the discovery and mapping of knowledge from this surfeit of information can offer great value in the process of software deployment. The key to successful management of this volume of resources is to adopt a strategy of discovering and mapping the important knowledge resources (tacit and explicit). More specifically it is proposed that the discovery and mapping of business requirements (technical and non technical) during the software deployment process will achieve faster turn-around time, improve quality of delivery, greater customer satisfaction, and advance competitive advantages. This approach should reduce many redundant data retrieval daily activities, such as scripts and queries that are generated and stored.

As stated in Nonaka and Takeuchi (1995), the process of knowledge creation focuses on knowledge conversion. The dynamic relations between tacit and explicit knowledge at the ontological dimension of knowledge levels (individual, group and organization) are impacted by developing the social structure. Based on these insights, this research adopted an ontology approach that is supported by the organization's social structure. The ontological approach increased the understanding of the social structure and improved the knowledge conversion process. The definitions of KM presented as ontology in Appendix 1 were derived from the perspective of software engineering professionals. Data, information, knowledge, and insight are representations of the level

of understanding. Data is obtained from facts that have no meaning or context. There are two types of data namely quantitative and qualitative, and they are always explicitly presented. Information is data that has a context, relationship, and meaning. Knowledge is information transformed from a "Thing" (being) to become more valuable or more effective as a basis for action. Insight (wisdom) is the highest level of understanding and it is mostly tacit knowledge that provides unique principles and solutions. There are domain assets that create knowledge. Organizations build their assets from a process perspective and therefore, process and knowledge have life cycles that can improve the quality and value of knowledge (Steenkamp and Konda, 2003).

Tacit knowledge can be subjective, representing action, experience, reflection, and more. This form of knowledge may be transformed to explicit knowledge to improve a process, procedure, and paradigm, redefine a concept, and so on.

Adopting a methodological framework for ontology engineering provides guidance to build and maintain an ontology that supports business requirements for software deployment. The "Developing Ontology-Grounded Methodological Applications" or DOGMA, is an ontology engineering framework, for retrieving and representing information. It was developed by the Semantic Technology and Applications Research Laboratories (STARLab) at "Vrije Universiteit Brussel" (ref, date). The DOGMA framework is based on building the ontology as a separate domain axiomatization and application axiomatization to address the engineering challenges of reusability versus usability. The DOGMA approach provides the advantage of increased reusability of domain axiomatization and increased usability of the application axiomatization (Jarr and Seersman, 2008).

4. ONTOLOGY-BASED SOLUTION TO MANAGE BUSINESS REQUIREMENT OF SOFTWARE DEPLOYMENT

The OKDMS is capable of automating the management process of software deployment, makes the management of software deployment less complex and easy to understand based on knowledge integration as illustrated in Appendix 2.

- The knowledge integration based ontology takes input from various data sources such as clients' information (name, location, and so on), business requirement data (maintenance time, backup data, and so on), product data (software requirements), and infrastructure data (disk space, memory, IP Address, and so on).
- Utilizing an execution mechanism, the knowledge integration based ontology is mapped to various knowledge sources, such as contact information of domain experts, procedures for the installation process, corporate standards information, software information (release information, update information, and so on).
- The support applications are the means of accessing the knowledge sources. Knowledge sources can be accessed via support applications such as support application for text, documents, and spreadsheets.
- The installation applications are those applications that allow the software deployers to perform the task of software deployment on the installation site and provide the means of performing software deployment activities. For example, a secure Telnet session or other web application deployment tool may be used to perform the deployments steps. Through web access, users can search for a product, discover concepts, acquire knowledge about them, and execute predefined tasks by the execution mechanism.
- The execution mechanism is capable of invoking support applications and installation applications. The ontology and the knowledge sources are stored in the repository making it possible to track all changes and updates to the knowledge integration based ontology.
- The discovered knowledge is derived from data and information from the domain asset. The domain asset consists of data sources and knowledge sources as shown in Figure 1. The retrieval mechanism adopts scripts and queries to search and discover relevant data. Scripts and queries are built, generated, and run by domain applications. The domain applications are also used to document and present explicit and tacit knowledge, such as reports, procedure, and knowledge based (KB) articles. The Executer can run executable files, retrieve, gather, and appropriately arrange data to be inserted as values to the object type

property or data type property of the ontology classes.

The ontology is accessible via a web server for knowledge search, discovery, and presentation. The ontology provides web links to discover procedure documents, software release documents, KBs, and reports in their original format.

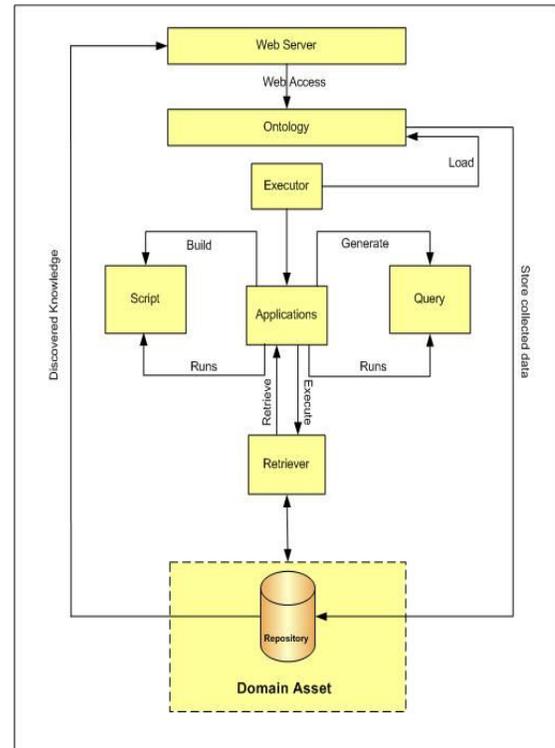


FIGURE 1. HIGH-LEVEL KNOWLEDGE DISCOVERY, RETRIEVAL, AND REPRESENTATION ARCHITECTURE

A knowledge mapping method was used to extract knowledge and identify relationships between them. It was employed to visualize knowledge embedded within the organization. The mapping process by itself often creates new knowledge and intellectual assets by discovering new relations. As stated earlier proactive KM for software deployment requires a dynamic knowledge-intensive discovery process to understand the environment and make better decisions. Tasks include for example reviewing of critical deployment documents, creating a high-level deployment plan, and assigning ownership. The draft conceptual solution includes the processes of deployment environment discovery and preparation, a

knowledge based ontology, web access, an execution mechanism, data and knowledge sources, execution and support applications, and a repository, as illustrated in Appendix 2.

5. FUTURE WORK

A key goal of this research is to developing and utilizing the OKDMS based on discovery and mapping of relevant data and information by developing a semantically interconnected knowledge-based system to manage the software deployment process. Also, we are developing the prototype to automate the management of software deployment process. Software deployment is a knowledge intensive process involving integrating knowledge from a diverse set of resources. The collaborative knowledge-based ontology is centered on supporting decision-making based on business requirements for the software deployment process. The next step of the research is to enable the discovery, mapping and creation of more detailed knowledge about clients' data storage and tracking software deployment issues.

6. CONCLUSIONS

This ongoing research is keenly focused on enabling an Ontology KM System based on discovery and mapping of relevant business deployment resources. At this point the deliverables of the project include the ontology viewpoint of KM (presented in Appendix 1), the high-level Software Knowledge Discovery, Retrieval, and Representation Architectures (presented in Figure 1). Also, the draft conceptual solution for OKDMS is focused on deployment environment discovery as presented in Appendix 2. This conceptual solution is being refined as further insights emerge from the research. The implementation phase will focus on developing data retrieval mechanisms and developing a prototype to upload all data to the ontology model design and automate the management process of software deployment. This implementation will explore several use cases, and provide the capability to evaluate the findings in terms of the conceptual solution, and enable

the researchers to validate the research proposition and research questions.

Insight gained so far is that business requirements are a critical success factor of the software deployment process and require dynamic discovery and mapping of knowledge. It is postulated that the proposed solution will enable knowledge sharing and reuse, enable understanding of information, and improve the overall delivery process.

7. ACKNOWLEDGEMENTS

The authors wish to express their appreciation to the sponsors of this study for providing an opportunity to apply Ontology Engineering and KM discovery and mapping theory to practice and collaborating on this educational project.

8. REFERENCES

- Belkin, J,N.(2000) Helping People Find What They Don't Know," *Comm. of the ACM*, 43(8), pp. 58-61.
- Bellard, F. (2005). Qemu, a fast and portable dynamic translator. In *USENIX 2005 Annual Technical Conference*, Anaheim, CA, USA.
- Briggs, D. (2000) Maximizing the Knowledge Asset Value within the Enterprise. *DM Direct*.
<http://www.dmreview.com/dmdirect/20000421/2146-1.html>.
- Das,A., Wu,W. and McGuinness,D. Industrial strength ontology management. In *Proceedings of the First Semantic Web Working Symposium*. Stanford, USA. 2001.
- Fischer, G., Grudin, J., Lemke, A. C., McCall, R., Ostwald, J., Reeves, B. N., Shipman, F. (1992). Supporting indirect, collaborative design with integrated knowledge-based design environments.
- Fischer, G., Ostwald, J. (2001). Knowledge Management — Problems, Promises, Realities, and Challenges, *IEEE Intelligent Systems*, January/February, pp. 60-72.
- Goldberg, P. R. (1973). Architecture of virtual machines. In *Proceedings of the*

- workshop on virtual computer systems, pages 74–112, Cambridge, Massachusetts, United States. ACM Press.
- Nonaka, I., Takeuchi, H. (1995) *The Knowledge Creating Company*, Oxford University Press, New York.
- Steenkamp, A.L. and Konda, D. (2003) *Information Technology, the Key Enabler for Knowledge Management Information Technology, the Key Enabler for Knowledge Management, a Methodological Approach*, *International Journal of Knowledge, Culture and Change Management*, Vol. 3, Monograph MC03-0070-2003, (Ed. M. Kalantzis and B. Cope),
- Sterling, T. (2001). *An Introduction to PC Clusters for High Performance Computing*, California Institute of Technology and NASA Jet Propulsion Laboratory, USA.
http://www2.sccc.ru/News/Prezent/final-White_paper.pdf
- VMware, Inc. <http://www.vmware.com>.

