Digital Ecosystems and Business Intelligence (DEBI) Institute

Lightweight Community-Driven Approach to Support Ontology Evolution

Ahmed Abdulridha Aseeri

This thesis is presented for the Degree of

Master of Philosophy (Information Systems)

of

Curtin University

November 2011

Declaration

To the best of my knowledge and belief this thesis contains no material previously published by any other person except where due acknowledgment has been made.

This thesis contains no material which has been accepted for the award of any other degree or diploma in any university.

Signature:

Date: 6th Nov 2011

Summary of Thesis

The challenge for businesses today is to operate efficiently and effectively by ensuring the availability of updated information necessary for business activities. At situations where daily operations require collaborative communication among staff working in dispersed locations - characterised by different time zones and working conditions - it is essential that the staff have and share common knowledge, especially since staff members may have diverse understandings and perceptions of an issue.

In order to overcome such challenge in business, using the oil and gas industry as a case study, this research proposes a platform based on a Lightweight Community-driven approach whereby staff are presented with opportunities to raise and discuss a particular issue in a systematic manner. In addition, ontology is used to support the representation of domain knowledge in which evolution is expected as human knowledge is not static. The fundamental principle of the Lightweight Community-driven approach is to involve organisation personnel in an exchange of ideas and opinions that lead to a higher quality of output. Staff are categorised in two groups, the Contributor Group and the Admin Group, each of which has a different role and responsibility in the whole process of communication. Members of the Contributor Group are those who are actively participating in the discussion by raising issues, providing feedback, and voting on an issue. This group is responsible for the quality of the discussion result which includes reliability of the ontology. The Admin Group are tasked with the managerial aspects of discussion to ensure the eligibility of every participant.

Ontology evolution, subsequently, takes place to incorporate the result of the discussion. When there is a need to revise the intended ontology, the current ontology will be improved and archived ontology is created. Otherwise, ontology will remain the same as it is considered relevant by employees of the oil and gas industry. The proposed platform is explained and validated in detail in this thesis. The outcomes of the collective efforts to improve the quality of information within the domain are: a decrease of day-to-day work, cost saving, increased productivity and the availability of a communication forum.

Acknowledgments

I would like to thank my supervisor Dr Pornpit Wongthongtham for her ongoing advice and support, and her keen perception of my strengths and abilities, which enabled me to determine the direction of my research. Also, I am grateful for her encouragement, and her insistence that I work hard in order to reach my maximum potential. I wish to thank my Co-Supervisor, Dr Farookh Hussain, for spending time with me and encouraging me to continue with this research whenever I felt overwhelmed. I am grateful for his advice on how to proceed further on each issue that surfaced during the course of my research.

I thank the examiners for their useful suggestions. I wish to thank Dr John Fielder and Ms Bruna Pomella for their professional proofreading and corrections of the English.

Most importantly, I would like to thank my family who provided me with years of unfailing support throughout the research. I dedicate this work to my mother Ms Esmat Aseeri, sister, Roya Aseeri and my brother Mahdi Aseeri.

I am deeply grateful to my colleagues, especially Novita Ikasari, for her support in providing ideas and suggestions on how to proceed further with some areas of this work. Her advice greatly improved the quality of my research. Finally, I also thank DEBII and Curtin University for providing all the necessary facilities and support enabling me to bring this research to a successful conclusion.

List of Figures

List of Tables

Table of Contents

List of Acronyms

OilEd (Ontology Interchange Language Editor)

OntoEdit (Ontology Editor)

OIL (Ontology Interchange Language)

CHAO (Change Annotation Ontology)

API (Application Programming Interface)

RPC (Remote Procedure Call)

OWL (Ontology Web Language)

HTML (HyperText Markup Language)

RSS (Really Simple Syndication)

RNA (RSS Aggregator Service)

W3C (World Wide Web Consortium)

OGO (Oil and Gas Ontology)

IO G1 (Integrated Operation Generation 1)

IO G2 (Integrated Operation Generation 2)

IO (Integrated Operation)

RDFS (Resource Description Framework Schema)

Chapter 1 - Introduction

1.1 Introduction

Globalisation has affected society and business in various ways and shaped the world as it is today. Since early 1980s, globalisation has contributed and is geared to social, cultural, economic, political and technological aspects of human life (Dreher, Gaston et al. 2008). Globalisation is defined as the "degree of interdependency and interrelatedness among different and geographically dispersed actors" (Archilbugi and Iammarino 2002). This highlights the continuous interactions among people in every facet of life irrespective of their location. In the business context, globalisation has made way for technological advancements and increased global business competitiveness.

In light of this competitiveness and the need to respond to broader demands of world-wide consumers without foregoing substantial competitive advantage, organisations have established offices in different parts of the world. The globalisation process requires teamwork among different levels of management that transcends geographical boundaries. As organisations will employ staff for individual locations, it is assumed these localized employees will have knowledge formed by very specific social, educational and cultural backgrounds. To improve functioning within the organisation as a whole, there is a need to have a shared knowledge representation to ensure that business objectives are attained. To ensure that the exchange of knowledge has occurred smoothly, businesses have applied ontology as a domain knowledge representation for their conceptual building blocks (Stojanovic, Maedche et al. 2002). Ontological processes are those which facilitate organisation of information to improve access and clarity. However, the dynamic nature of human beings means that individual knowledge is changing constantly. Augmented by perpetual interactions, the knowledge represented by ontology needs to evolve accordingly.

The oil and gas industry is a particularly significant example of how ontology in information management can affect performance. In this industry, ontological data hierarchies can be applied to inform several domains of technology including information technologies, communication technologies, and data management. Limited exchange of unsecured data between competitors is also desirable as it helps organisations cultivate a better knowledge of the industry-specific challenges faced by these companies as they pursue petroleum extraction. By using the oil and gas industry as an example of how ontologies can be applied in data and knowledge management both in individual companies and throughout the industry as a whole, the significance of ontological hierarchies and ontological organisation will be made clear. Finally, while the information presented in this research paper is specific to ontological applications found in the oil and gas industry, the importance of these applications can be transferrable to different industries. The principle users of the ontologies described will be participants in the oil and natural gas industry, but information from this research is relevant and appropriate to inform ontological applications in other global ventures.

In this first chapter, an introduction to (i) Ontology definitions, (ii) Ontology Editor and (iii) Ontology Evolution, is presented. The chapter goes on to identify the advantages of ontology evolution, the current lack of a systematic approach for ontology evolution, and explains the motivation for this study. The chapter ends with the objectives of the study and the thesis structure.

1.2 Ontology Definitions

There are many attempts to define ontology which encompasses the philosophical and information science domains (Smith and Welty 2001). The term “ontology” is derived from its usage in philosophy where it means the study of being or existence as well as the basic categories. Therefore, it is now used to refer to what exists in a system model.

Definition 1: One of the earlier efforts to define ontology was made by (Sowa 1983) who claimed that ontology is “a catalogue of everything that makes up that world, how it’s put together and how it works”. Although it captures both worlds, the definition is an imprecise and ambiguous conceptualisation of ontology.

Definition 2: On the other hand, a renowned researcher, Thomas R Gruber, provided a more concrete definition of ontology in 1983. He defines it as a study which explicitly explains concepts and relationships with sets of concepts (e.g. classes, relations, functions) that are used to represent and describe domain knowledge (Gruber 1993). For example, in the oil and gas industry there is an established ontology for Statoil in Norway (Association 2008) a standard library related to an oil and gas domain.

1.3 Ontology Editor

Ontology Editor is an application which was developed to view and edit ontology. In the past few years, many applications have been developed such as OilEd (Bechhofer, Horrocks et al. 2001), OntoEdit (Sure, Erdmann et al. 2002), Protégé (Gennari, Musen et al. 2003) and Web-Protégé (Tudorache, Vendetti et al. 2008). Further details are given below:

- OilEd: OilEd was developed at Manchester University. It is a simple ontology editor that provides further guidance in the development of Ontology Interchange Language **(**OIL)-based ontologies (Bechhofer, Horrocks et al. 2001). It is the one which pioneers ontology editing (Bechhofer, Horrocks et al. 2001).

- OntoEdit: OntoEdit was developed by the Knowledge Management Group at University of Karsnuhe Institute. It provides an ontology development that allows collaboration and inferencing. The method involves three main steps: requirements specification, refinement and evaluation. In the first step, the ontology engineers and domain experts meet and work towards identifying the goal of the ontology, description of the domain, and the availability of references. Design guidelines are also established in this step. Then, the team takes the ontology to the refinement phase. Finally, the ontology requires evaluation according to its requirement specifications by identifying possible errors in the ontology and efficiency for enabling collaborative work (Sure, Erdmann et al. 2002).

- Protégé: Protégé was developed by Mark Musen at Stanford University. It is an ontology editor which has come a long way. Protégé started in 1987 as a small application, which was aimed at building knowledge acquisition tools. Protégé has then been developed further, providing many new features with each version that has been released. Currently, hundreds of individuals and research groups are using Protégé (Gennari, Musen et al. 2003).

- Web-Protégé: Web-Protégé is a Web version of Protégé, also developed at Stanford University. This allows the users who have access to view and edit the ontology from the Internet (Tudorache, Vendetti et al. 2008). However, this research will focus on Web-Protégé because it is readily available online, enabling people in different locations to use it.

1.4 Ontology Evolution

Ontology evolution is the process by which a domain changes, or there is a change in the conceptualisation or specification of the ontology. Traditionally ontology evolution pertains to the changes in domains or conceptualisation (concepts): or changes due to user preferences that might use the data in a way not previously intended or changes in the data itself (instances).

1.4.1. Evolution at a concept level

There are cases where an ontology lacks, or is not informed by, instance data. When instance data is missing, changes must be made to make concepts relatable. Machines must relate to concepts which are “formally defined semantics that machines can interpret” (Klein 2004). A common example of ontology evolution is where two departments consider merging their separate systems. Klein (2004) posits that because ontologies drive search functions and provide the framework for navigating through large collections of data (web sites, text, data) any merging of systems has to include not only commonalities in language, but also in how the system works through the process of retrieving the necessary instance data. This will not be the focus of this study.

1.4.2. Evolution at an instance level

As mentioned above there are some ontologies that don’t have an instance data but for the purpose of this research the paper will focus on those ontology evolutions that include instance data. There are often situations where it is difficult to determine where classes and instances differ purely because the classes themselves have sub classes that could appear to be instances, but in fact are not. An example outlined in Klein (2004) was in the case of a specific book title (the Lonely Planet for Amsterdam) is an instance of a class known as “Travel Guides” if accessed through the book store. But the individual copies of the books – a list of the books under Travel Guides – Lonely Planet for Amsterdam for example, are instances.

1.5 Advantages of Ontology Evolution

When organisations have integrated an ontological hierarchy into their operating processes, it is necessary to update these to improve consistency and accuracy of information. The problem with ontology in an ever-changing business context is the fact that ontology does not evolve at the same pace as the business itself. In a particular sector such as the oil and gas industry, an updated ontology is crucial. Delays or even failure to transfer data and to change the ontology in a timely manner might result in great losses within an industry. An ontology evolution platform has become a vital part of business with advantages such as: (i) error reduction; (ii) cost saving and (iii) users' involvement. It has been postulated that the evolution of the ontological hierarchy might be as significant to the success of the ontology as process of initial ontological establishment (Gartee 2011).

1.5.1 Error Reduction

Due to the daily operational work, an updated ontology is imperative if organisations are to reduce errors within their daily processes, as delayed information leads to uninformed decisions. Any issues with the ontology will be addressed at an early stage, along with any errors that other staff members have if they are inconsistent or if an updated concept needs to be stored in the ontology. The overall productivity of the organisation will improve if the staff's recognition and reporting of errors leads to a data update. For example, information on a sudden temperature change will compel staff in an offshore office to postpone drilling activities. This information is vital to ensure the safety of the offshore staff. Therefore, due to daily critical operational work, it is vital to receive the information instantaneously in order to prevent any errors in decision-making (Thorsen and Rong 2008).

1.5.2 Cost Saving

Businesses need to incorporate changes and new information in their daily business activities to decrease unnecessary costs. This is achieved by regular updates of ontology. For example, there are issues related to certain proportions of oil and water during drilling processes that have to be maintained by the offshore staff to ensure high quality oil. A high percentage of water will impact on the quality of drilled oil. Pre-drilling oil discovery must thus be performed to ensure the extraction of high quality oil. If the offshore staff insists on drilling without the pre-drilling oil discovery and low quality oil is extracted, machines that have been used to drill based on false information will be less useful. Therefore, by allowing all the staff (onshore and offshore) to report any issues within the process, this will provide a huge saving to the oil and gas industry.

1.5.3 Users Involvement

An organisation needs to have a smooth interchange of knowledge among the users. This requires a dynamic collaboration of all participating members or staff in order to come to an agreed updated ontology. On many occasions, the evolved ontology has become obsolete since users have not been involved in the evolution process, resulting in their inability to comprehend the ontology. The manager, for instance, will identify the need to revise ontology, whereas ontology engineers will perform the actual change, simply because the members of the hiring organisation do not have enough expertise to create or revise the ontology. The ontology engineers, on the other hand, do not have the first-hand information optimally needed to change the ontology. Changes to ontologies will emerge through continued practice and application (Gartee, 2011). Classification systems are representative of the current state of the organisation within the broader context of the industry and external forces such as the economy. In ontologies, user involvement and data from external sources can contribute new evidence, and consensus among users can facilitate a specific informed direction for the evolution of the ontology (Gartee, 2011). In this respect, methods of ontology evolution described in this research benefit from having maximum involvement of users throughout the evolution process.

In the development of informed ontological hierarchies, proposed changes to the existing system can be informed by users but must be approved by administrators. Contributors are general, as these are users who participate in the ontology and have first-hand information in how changes could be made to improve the classification system and its practical applications. Administrators are specific, as these are users who are professionally invested in the management and the improvement of the ontology. Involvement is typically designated by positions within the organisation; administrators have access to the infrastructure of the ontology and can affect change, while general users contribute to the change process by engaging in routine use of the system.

There is, however, a caveat to the assumption that ontological evolution will be informed by a wide user base. Many successful globalised organisations have not pursued organisation- or industry-wide expansion of internal ontologies, and have not encouraged all participants within the organisation to contribute to either the construction or the revision of the existing ontological hierarchies. The decision to segregate participants from use or reform of the ontology is a management decision, not a technical decision, and is not germane to this research project.

1.6 Problems in Ontology Evolution

There are two contrasting approaches available in the field of ontology evolution. The first approach involves users in an active role (Klein 2004; Stojanovic 2004; Vrandecic, Pinto et al. 2005; Noy, Chugh et al. 2006) whereby a modification of the existing ontology is a result of collaborative efforts among ontology developers, engineers, editors and experts. On the other hand, the second approach delegates the task of ontology evolution to machines where collaborative efforts are performed in an automated manner (Alani, Harris et al. 2006; Bloehdorn, Haase et al. 2006; Novacek, Laera et al. 2007).

Although human-based ontology evolution is preferred, two issues need to be addressed with regard to expertise and geographic boundaries. The first issue occurs when ontology is modified manually and it requires people with a sound knowledge of ontology to perform the actual modification. This means that an Ontology Engineer, or someone with the same expertise, has to be employed by an organisation. In a real-world situation, it is safe to assume that many organisations find this to be impractical.

The second issue relates to the different locations from which business activities are carried out. Nowadays, many organisations operate in various countries and regions. In many cases, different segments of business are performed in different parts of the world. For example, Australian-based clothing industry Bonds has a head office in New South Wales but has a factory in China where clothing is produced (Sharp and Zappone 2009). Some business sectors even have to operate in dispersed locations as part of their business. One example of such sectors is oil and gas. Business activities within the oil and gas industry are managed by staff that are working from offshore and onshore offices. In addition, these offices are most likely to be dispersed around the globe, creating issues such as delayed transfer of information.

1.7 Motivation of Study on Ontology Evolution

As described earlier, ontology evolution brings monetary as well as non-monetary advantages for businesses. The importance of an updated ontology in business contexts is indisputable especially to particular types of business where activities are managed from dispersed locations. Within this context, the failure to manage these diversities will disadvantage the organisations. Moreover, due to the wide and distributed location of the staff, it is vital to have an updated knowledge captured in ontology to process the work on updated data. The following motivations have initiated the search for possible answers in this study.

1.7.1 Up-to-date Ontology

Ontology is a vital part of the daily activities of business organisations such as the oil and gas industry. Within this particular business sector, ontology needs to be updated all the time in order to increase the productivity and reduce operational errors and maintenance costs, which saves a organisation millions of dollars (Thorsen and Rong 2008). Furthermore, due to the involvement of all staff in updating the ontology as described in the previous section, ontology evolution will occur more frequently. Therefore, the changes in the instance level in oil and gas ontology are an important part of enabling onshore and offshore staff to operate in a more efficient and effective way.

1.7.2 Effective Collaborative Communication

As briefly explained in previous sections, business activities are not bounded by geographical factors. Organisations are known to have offices established in many areas of the world with many branches to operate. This makes communication between staff much harder than if they were working in the same location. Staff who work in different environments will have a different reaction to different issues resulting in different perceptions and knowledge. Furthermore, the information between geographically-dispersed branches is not fully integrated. This leads to a loss of vital information that could be transformed into useful knowledge. In order to prevent this loss, organisations need to enhance the communication between scattered offices. Within this context, a platform will provide further opportunities to enhance the much needed communication in order to develop and change the shared knowledge captured in ontology when required.

1.8 Research Objectives

In the above sections, a description of the need for ontology evolution has been explained. By developing a platform that allows better communication amongst staff, organisations will benefit from reduction in errors, economic benefits that include higher productivity, and users' involvement. Therefore, there are two main research objectives for this study:

Objective 1: To develop an approach that involves users in the ontology evolution process.

Objective 2: To enhance the collaborative communication among different participants who are working from scattered offices which leads to ontology evolution.

The fulfilment of both objectives will ensure a more supported updated ontology, which is understood by the users and increases the utility of the ontology. The proposed approach acknowledges challenges that arise from various geographic locations as well as individual contextual backgrounds. Given these unavoidable factors, a platform for communication is essential to facilitate ontology evolution process. The platform will help the staff to understand and to participate in resolving the issue of outdated ontology. Also, the objectives address the problems and issues associated with geographically-dispersed staff and the best ways by which they can communicate with one another.

1.9 Thesis Structure

The thesis is structured as follows:

Chapter 1 – Introduction which is this chapter.

Chapter 2 – A review of the literature related to ontology evolution, ontology editors s and plug-ins, knowledge management, formal social networking - Wiki and informal social networking is presented. At the end of the chapter, a critical evaluation is undertaken to provide an integrated perspective.

Chapter 3 – The chapter starts by describing the problem overview. Two main problems facing a Community-driven Ontology Evolution are discussed, and this corresponds to the two main research issues. The first research issue is the Lightweight Community Support, which refers to a lack of users' involvement in ontology evolution. The second issue is that there is the need for an ontology evolution platform that allows ontology to evolve. The chapter also presents the requirements for the two proposed solutions to these problems. The oil and gas industry is introduced as a case study, and the chapter concludes with the presentation of the engineering-based research approach.

Chapter 4 – This chapter outlines the proposed solution, beginning with an overview of the Lightweight Community-driven Approach and the three key principles underlying this proposed solution. Then the conceptual framework for Lightweight Community Driven Approach is explained, followed by a discussion on the advantages and disadvantages of the approach. Next, the chapter describes the Lightweight Community-driven Approach as a solution to users' involvement in ontology evolution; here, the various roles and responsibilities of the groups are described in detail. In addition, a ticketing support system and the differences and similarities between community forums and ticketing support systems are presented. Finally, the platform as a solution for ontology evolution is presented, and its three layers are discussed thoroughly. The three layers are: (i) social networking layer, (ii) permission layer, and (iii) ontology evolution layer.

Chapter 5 –This chapter comprises the second key principle of the Lightweight Community Driven Approach, i.e. the implementation of the approach. The developed platform is demonstrated by way of pseudocodes, showing every feature that is available on the platform. The chapter continues with a detailed illustration of the calculation of reputation value and voting points, each with a corresponding example.

Chapter 6 – This chapter describes the proof of concepts of the proposed platform. It starts with platform requirements for the Lightweight Community Support and Ontology Evolution Platform. An illustration of the proposed solution is given. This includes how the platform is populated, modified and deleted. The features discussed include login, issue, project, vote and vote results. The chapter further presents a sample of oil and gas ontology which has been created to demonstrate the evolution process of ontology in the oil and gas domain. This includes classes, properties and individuals.

Chapter 7 – The final chapter presents the issues and corresponding solutions for lightweight community support and an ontology evolution platform. Finally, possible directions for future work are suggested.

1.10 Conclusion

In this chapter, the research problem is introduced. This is followed by an explanation of ontology, ontology editor and ontology evolution. It then identifies the advantage of ontology evolution and acknowledges its related problems. This chapter then explains the motivation for this research and the objectives. In conclusion, the thesis structure is presented. In the next chapter, a review of literature relating to existing ontology editors and plug-ins and two main types of social network is discussed.

1.11 References

Alani, H., S. Harris, et al. (2006). Winnowing ontologies based on application use. Proceedings of 3rd European Semantic Web Conference.

Archilbugi, D. and S. Iammarino (2002). "The globalization of technological innovation: definition and evidence." Review of International Political Economy **9**(1): 98 - 122.

Association, P. C. (2008). "POS Caesar Association." Retrieved March 16, 2010, from http://projects.dnv.com/reference_data/RD4Browser/default.aspx.

Bechhofer, S., I. Horrocks, et al. (2001). OilEd: A Reason-able Ontology Editor for the Semantic Web. KI 2001, Vienna, Austria

Bloehdorn, S., P. Haase, et al. (2006). Ontology Evolution, John Wiley & Sons.

Dreher, A., N. Gaston, et al. (2008). Measuring Globalisation: Gauging its Consequences.

Gartee, R., (2011). Electronic Health Records. American College of Physicians.

Gennari, J. H., M. A. Musen, et al. (2003). "The evolution of Protégé : an environment for knowledge-based systems development." International Journal of Human-Computer Studies **58**(2003): 89 - 123.

Gruber, T. (1993). Toward Principles for the Design of Ontologies Used for Knowledge Sharing. International Workshop on Formal Ontology, Padova, Italy.

Klein, M. (2004). Change management for distributed ontologies. Amsterdam, Vrije Universiteit. **PhD Thesis**.

Novacek, V., L. Laera, et al. (2007). Semi-automatic integration of learned ontologies into a collaborative framework. IWOD.

Noy, N. F., A. Chugh, et al. (2006). A Framework for Ontology Evolution in Collaborative Environments. 5th International Semantic Web Conference 2006, Athens, USA.

Sharp, A. and C. Zappone (2009). "West End factory to close as Bonds goes offshore." Retrieved December, 4 2009, from http://www.brisbanetimes.com.au/articles/2009/02/25/1235237708475.html.

Smith, B. and C. Welty (2001). Ontology: Towards a new synthesis. International Conference on Formal Ontology in Information Systems.

Sowa, J. F. (1983). Conceptual structures: Information processing in mind and machine.

Stojanovic, L. (2004). Methods and Tools for Ontology Evolution. Karslruhe, Germany, FZI - Research Center for Information Technologies. **PhD Thesis**.

Stojanovic, L., A. Maedche, et al. (2002). "User-Driven Ontology Evolution Management." Knowledge Engineering and Knowledge Management: Ontologies and the Semantic Web **2473**(2002): 7 - 27.

Sure, Y., M. Erdmann, et al. (2002). OntoEdit: Collaborative Ontology Development for the Semantic Web. International Semantic Web Conference 2002 (ISWC'02), Sardinia, Italy

Thorsen, K. and C. Rong (2008). Data Integration in Oil and Gas at Norwegian Continental Shelf. 22nd International on Advanced Information Networking and Applications - Workshop, Okinawa, Japan.

Tudorache, T., J. Vendetti, et al. (2008). Web-Protégé: A Lightweight OWL Ontology Editor for the Web. OWL: Experiences and Direction (OWLED) Workshop, Karslruhe, Germany.

Vrandecic, D., H. S. Pinto, et al. (2005). "The diligent knowledge processes." Journal of Knowledge Management **9**: 85 - 96.

Every reasonable effort has been made to acknowledge the owners of copyright material. I would be pleased to hear form any copyright owner who has omitted or incorrectly acknowledged.

Chapter 2 – Literature Review

2.1 Introduction

In this chapter, an overview of ontology evolution is presented and includes how ontology editors have contributed to the process of ontology evolution. A brief introduction of different ontology editors is provided. The chapter then introduces ontology evolution in a visual form and by outlining knowledge management. The chapter also discusses formal social networking (Wiki) and informal social networking (i.e. Weblog). This chapter concludes with a critical evaluation of the existing approach. The discussion is twofold: (i) currently, there is no platform that supports ontology evolution, and (ii) there is a lack of community support for the evolution process.

2.2 Ontology Evolution

Several studies have suggested various methods that support automation. Moreover, studies have enhanced the evolution of ontology by developing ontology editor plug-ins. Other studies present evolution more visually, while several have proposed digital libraries which enable the ontology editors to analyse and make recommendations to users. These recommendations are about changes, which ontology requires. In addition, researchers have developed Web-Protégé (Tudorache, Vendetti et al. 2008), intended to improve the shared communication between the users, which makes the ontology evolution process easy and more accessible.

2.2.1 Ontology Editor: Protégé

In recent decades, there has been an increase in the development of ontology editors, some of which include OntoEdit (Sure, Erdmann et al. 2002), Protégé (Gennari, Musen et al. 2003) and OilEd (Bechhofer, Horrocks et al. 2001). Each of these ontology editors has both advantages and shortcomings. Nevertheless, Protégé is the most widespread, commonly-used ontology editor. Moreover, the developers have produced a Web version known as Web-Protégé. (Tudorache, Vendetti et al. 2008). Therefore, this research will use Protégé because it is considered to be the most suitable ontology editor.

Protégé, developed by Mark Musen, emerged in 1987. It has undergone several development phases, each of which is an improvement on the previous version. These phases were Protégé 1; Protégé 2; Protégé- 2000 (Gennari, Musen et al. 2003). However, there are many other Protégé phases, the most recent version being Protégé 4.0 at time of writing. Ontology protégé provides more flexibility since Protégé-2000; this is by allowing the development of plug-ins in Protégé. Since that time, many plug-ins have been developed. Developers use two ways to develop plug-ins-slot-widget plug-ins and tab plug-ins (Gennari, Musen et al. 2003).

Another Protégé's core system feature, which is in favour of ontology evolution, is the client-server mode to carry out editing on ontology by many users. This means that many users are able to connect to the Protégé server through Protégé clients and to update and edit ontology. This provides feedback to all the users; this is because they are able to see ontology updates instantly. Nonetheless, Protégé is not an application that is Web-based and is one which restricts the users' access to ontology online. In addition, it does not provide a practical approach that allows the users to become involved in discussion of any changes in ontology when needed.

2.2.1.1 Protégé Plug-In

There are many plug-ins which add more features to Protégé. Moreover, these are intended to automate or semi-automate the process of ontology evolution. In this section, we describe several plug-ins developed to support ontology evolution.

Change Management Plug-In

The Change Management Plug-In serves many purposes for Protégé. First, it records the changes, and does so with an illustration of the change as well as with Annotation Ontology (CHAO) (Noy and Klein 2003) with timestamp and the author. Therefore, after enabling the plug-in in the Protégé, the changes are monitored to determine whether there are any changes. Furthermore, all users can have contact with CHAO; this is in either Protégé or the Protégé knowledge base, and API (Noy, Chugh et al. 2006).

Secondly, the Change Management Plug-in has two additional views: detailed and summary views. The detailed view displays the low-level changes which are made to any change in ontology. Conversely, the summary view categorises the low-level changes to higher-level changes. Moreover, users can choose a change or even view annotations for any change. The user can also provide annotations for the chosen change or group. Furthermore, by enabling the Change Management Plug-in, more menu items are viewable on the classes tab of the Protégé application (Noy, Chugh et al. 2006). Nevertheless, although the Change Management Plug-in does track all changes, it provides no features to enable the users to discuss the change, or debate and collaborate before making a change.

PROMPT Plug-In

Another plug-in developed for Protégé is the Prompt Plug-in which offers several ontology management functionalities, and includes four main functions. First, it compares the recent ontology, then it moves frames in between current projects as well as included projects. Thirdly, it merges ontology; and eventually extracts a part of ontology, and then adds it to recent ontology projects (Liang, Alani et al. 2005).

The comparison process is very important in this ontology evolution context; therefore, this section describes the comparison of ontology evolution in more detail.

When a user activates plug-in, there is an addition of another tab with Protégé's name; this is the Prompt tab. Plug-ins operates into two ways. At first, if the sample of CHAO exists, it makes use of its instances to put together the changes, which they then present to the users. Secondly, if CHAO instances are non-existent, it makes use of the PROMPT algorithm to compare versions and discover structural differences (Noy, Kunnatur et al. 2004). Furthermore, PROMPT provides users with a feature to reject or accept changes. In addition, it provides a list of any changes for every class, or for chosen classes to view. Therefore, the users are in a position to either acknowledge or decline all changes or a particular change. In some cases, where CHAO is available, a list is compiled of the users who have made changes to the ontology; this includes a display of any concepts which conflict with modifications made by others (Noy, Chugh et al. 2006).

In addition, CHAO provides two more types of views: the Table view and the Tree view. The Tree view is intended to show all changes; this is in order to either reject or accept the changes. The Table view enables users to save all changes and the output of all these changes in a file (Liang, Alani et al. 2005). However, the PROMPT plug-in is used for the four specific functions as shown above. However, PROMPT is a plug-in developed for Protégé and is therefore not available for other ontology editors. Additionally, PROMPT does not allow users to communicate in order to compare the processed ontology.

Evolva

Evolva is an ontological evolution framework. It is intended to reduce the users' input. This approach has the following components beginning with discovery of information, then proceeding to validation of data, ontological changes, and validation of evolution and ends with management of evolution. It begins by comparing information from external domains and certifies it through a combination of heuristic rules. It then uses a continuing matching technique, which starts from the easiest and quickest to a more complex technique that is time consuming. If no relations are found, it may ask for manual checkups. Finally, Evolva begins with evolution management and validation; this is where it carries out a consistency as well as a duplication check and acquires tracing records or roll backs (Zablith 2008). Nevertheless, even with continuing efforts to improve Evolva and make ontology evolution automatic and further perform checks to ensure accuracy, it cannot be applied in all scenarios since, often, ontology requires prior discussion at the beginning of the evolution process.

2.2.1.2 Web-Protégé

Web-Protégé is a Protégé client used to network with Protégé server. This gives solution to support the ontology evolution process. Web-Protégé is a shared platform, which allows the users to renew as well as view cases and concepts. In addition, it is capable of segregating user's privileges by giving them suitable permission. Moreover, this assists others to see and track changes instantly (Tudorache, Vendetti et al. 2008).

Web-Protégé has a client and a server side. The server provides access to the ontology API. Collaboration API allows the users to comment on or make changes to certain ontology. In addition, collaboration API manages conflict that may arise when different clients make changes to similar ontology. Similarly, a client is a user interface whereby the users interact with the servers. It has Remote Procedure Calls (RPC). The RPC is a module used for communication with servers. Nevertheless, the recent version of Web-Protégé contains some challenges, which the development of subsequent versions will attempt to address. The challenges include the lack of support for the OWL (Ontology Web Language) 2.0 with its most recent infrastructure, the insistence on access rules for browsing as well as editing ontology and scalability because large ontology could be an issue with the restrictions of JavaScript (Tudorache, Vendetti et al. 2008). Nonetheless, Web-Protégé remains the most suitable web-based ontology editor available at the time of editing this thesis. The Web-Protégé is undergoing further development in the alpha version; however, not every feature works properly. It requires more extensive development if it is to be a complete workable software.

2.2.2 Ontology Evolution in a Visual Presentation

Visual Presentation is one approach that presents ontology evolution. This is inclusive of Protégé plug-ins or any other application which supports the process. More so, users identify a requirement list to visualise the changes in ontology evolution (Blundell and Pettifer 2004). Nevertheless, visual presentation is a task that is challenging. It has an expressive nature as well as a series of graphs for a duration of time (Blundell and Pettifer 2004).

Media is one domain that frequently changes and therefore necessitates immediate update of instances in the ontology. Consequently, ontology evolution becomes a main issue in media. Visualising these changes greatly assists the process of ontology evolution. In order to semi-automate and visualise evolution ontology, some studies take a sample data for six months (Weichselbraun, Scharl et al. 2007). In addition, these studies classify ontology evolution according to three terminologies that begin from core to the extended domains and eventually they wind up with peripheral. In addition, there is a discussion of the three diverse kinds of relationships. The discussion is in progress to a similar topic could change from one relationship or a terminology to another. This is especially so at a time when the priority given to topics changes. However, even having semi-automation and media ontology visual presentation, there is a need to enhance automation and move ontology to the Internet. This is because automation has some technical restrictions since all necessary tasks cannot be automated. Furthermore, the automation process may contain errors and security may be unreliable. The enhancement of media worldwide is vital as it helps to disseminate information to vast numbers of people. Any information needed in this world is available on the Internet (Trevathan 2006).

2.2.3 Knowledge Management and Digital Libraries

Knowledge management has undergone several stages. It began with KM 1.0 that is a database, which stores information. It represents this information to the user when requested. KM 2.0 adds to social networking, a method that allows either chosen users or any other user to take part in knowledge exchange. This involves various types of tools available for social networking, like Blog and Wiki, (the chapter will discuss them later). In addition, it encourages the development and sharing of the new knowledge and represents the same to the society (Dave and Koskela 2009). However, trust is one of the main factors ensuring that the data is accurate and trustworthy. Consequently, other mechanisms have emerged for the assessment of the trustworthiness and the accuracy of the data (Huang, Cheng et al. 2009).

Knowledge Management 3.0 makes use of Ontology Evolution as a framework tailored to the digital library. In addition, it is a content structure based on semantics. Two approaches have been proposed: the data-driven approach and the user-driven approach. The user-driven method uses hierarchy pruning as well as a shared approach (Haase, Volker et al. 2005).

Hierarchy pruning consists of two methods: reduction and expansion. Expansion analyses the frequency of concepts usage. It moves lesser used concepts into sub-concepts. It achieves this by grouping the sub-concepts under one main concept. Nevertheless, this method is not applicable in different instances because grouping could affect the logic behind the hierarchy created initially. On the other hand, reduction transfers less frequent concepts in combination with most frequent concepts. Nonetheless, hierarchy pruning is important in some of the instances. For instance, if one concept under the Pizza ontology is vegetarian pizza, and only few customers are ordering this, it does not mean that it is less visible as it could be a sign that the market for the product needs to be extended (Haase, Volker et al. 2005). Moreover, in the field of knowledge management, social networking features are used extensively; this is because Knowledge Management 2.0 is equivalent to Knowledge Management 1.0 added to a social network. Knowledge Management 3.0 is equivalent to Knowledge Management 2.0 added to Ontology. This enables more discussion among the users and allows ontology to grow in a timely manner. However, there hasn’t been a discussion of the use of knowledge management as a platform that allows the ontology to evolve and ensure that it is Web-based.

2.3 Social Network

Social networks have created a new world where users can participate and share knowledge with other users. Furthermore, many top Internet websites have a social network feature or they contain embedded features of a social network (Fun and Wagner 2007). This includes new technologies such as blog, community and Wiki forums. Moreover, they help to provide an enormous quantity of information which is accessible to the public. Nevertheless, due to the growth of ontology, and because the Semantic Web is a key factor in the organisation and analysis of information into a more important structure, as well as improvements to the automated process of making decisions and combining the technologies, this could prompt innovation and produce an appropriate way to address issues encountered by the ontology evolution. More so, in this study social networking is divided into two categories: Formal Social Networking - Wiki and Informal Social Networking.

2.3.1 Formal Social Networking - Wiki

Social Networking-Wiki is a Web-based application, which is Web-based. It allows users to alter the content by editing the pages online. This encourages users to team up and work together to make, build, modify and enhance the text (Ebersbach, Glaser et al. 2008). Wiki emerged in 1995 and was initially called WikiWiki Web. The software developer at that time was not happy with documents for the word processing. Consequently, Wiki functions as a portal enabling several users to participate in the process of ontology evolution (Siorpaes 2007). Nevertheless, obtaining changes in evolution ontology is of much importance to Wiki because it enables the user to analyse the changes. On the other hand, this enables the users to change and examine the preceding versions of ontology (Aseeri, Wonghtongtham et al. 2008).

The expansion and popularity of Wiki has encouraged many organisations and institutions to use Wiki even more. One major reason for the popularity of Wiki is its free online encyclopaedia, Wikipedia. The Wiki software enables visitors to collaborate on topics of general interest; users can edit, add, create or even delete articles on topics in which visitors have an interest. Each visitor is in a position to take part in the process of authoring without the necessity of authentication. More so, every user can revise other advanced work. There are about 8 million articles in 253 languages created in this manner (Potthast, Stein et al. 2008). The users fall into three categories: Lobbyists, Spammers and Vandals.

Lobbyists are users who come up with their own plan. Spammers solicit services or products and Vandals are those who damage other users' work. For this reason, Wikipedia develops rules for manual acknowledgment or they handle the cases which are inappropriate. In addition, there is less automation in this process of acknowledging and recognising the users (Potthast, Stein et al. 2008). Semantic Wiki directly places semantic mark-ups in the machines; this is to make sure that the data is readable and it aligns with people's readability of the article. Users find the semantic Wiki info box below each page (Vrandecic, Pinto et al. 2005). This helps the editors to comprehend the mark-up; this is because the semantic Wiki system elaborates the input. Semantic Wiki helps to create a domain ontology by through suggesting domains and ranges the roles according to specific concepts.

Nevertheless, Wiki is applicable mainly for information that is more concrete and established. Some of its information calls for an informal discussion prior to putting it into the Wiki. Consequently, Wiki is put to the best use when there is a developed debate or information.

2.3.2 Informal Social Networking

Informal social networking allows the user to have access to informal debate on topics of discussion, which are preferably new. Users can raise issues and communicate their views freely, giving them the opportunity to acquire new knowledge. The main informal social networking in the Internet is Weblog, Semlog and Community forum. These are discussed in detail below.

2.3.2.1 Weblog

In the past few years, Web logging has become popular in the world of the Internet. It provides an essential medium which enhances people's interaction with each other. Web logging users fall into various categories: habitual, personal, active and blogging larkers, ranging from the most addictive to the least addictive users (Fun and Wagner 2007).

A sole Weblog may contain many features. First, every blog contains its own link that is permanent, "permalink". Weblog users can refer to the permalink when trying to reach a blog. This could be their own Webpage or a part of the Webpage. Blog space is a communication space for the users (Searls and Sifry 2003). In addition, it is a kind of repository for knowledge and shares it via the online hyperlinks. In the Blog space, bloggers read and refer to topics also known as Blog rolls on the sidebar. The users make comments and or tag information on each other’s postings. This enables them to develop a better relationship because of the topics in which they have a common interest (Zhou and Davis 2007).

There are several other applications such as Word press, Blogger and Egblog. They encourage and give opportunity to its users to participate and manage their blogs easily. In addition, there are many applications where users choose plug-ins which enable them to choose extra features for their personal blogs. Moreover, this creates scope for new ideas and allows these to be shared with other people. Weblog helps in creating articles for different areas. It allows users to post their comments and responses to the articles. In addition, Weblog helps in the writing of articles on several topics, and subsequently this improves the ontology evolution process. Weblog provides features which enhance the process of ontology evolution by allowing users to raise their discussion on a particular ontology concept which creates a systematic link between ontology and Weblog.

2.3.2.2 Semantic Weblog (or Semlog)

Weblog estimates that, since 1999, the number of Web loggers has increased to about one million and it is on the increase. Nonetheless, currently Weblogs do not have semantic structures. A Weblog consists of many posts and is an HTML language, which does not have metadata that describes it (Yan, Frank et al. 2003). Consequently, Semantic Weblog provides remedies for this issue. This involves Semlog (Ohmukai, Takeda et al. 2004) and the Web Scripter (Yan, Frank et al. 2003). Semantic Weblog offers a comprehensive environment for authoring, gathering, publishing as well as creating human relationships. The relationship enables people to exchange ideas and knowledge easily and informally (Ohmukai, Takeda et al. 2004). This is into two layers. Information activity is the first layer with three functions which are to collect, make and donate the information. Communication activity is the second layer and its three functions are to relate, to collaborate and to present people. Weblog supports these activities more than do the Webs that support only the publishing of information. Semlog extends Weblog by including more flexible but same operations such as clipping and aggregation. It facilitates searching and contacting others (Ohmukai, Takeda et al. 2004).

Semlog uses the format of lightweight metadata. This is like the RSS (Really Simple Syndication) and it activates the flow of information and the interests it has. This involves three interest levels, which are to check, to clip and to post. Checks indicate that users browse Websites but they do not know the type of content found in the Web. Clip means that the users have more interest in the site as compared to the check users. In addition, clip can be listed in the users' bookmarks or their favourites. Finally, post indicates that the users can add comments. More so, they can publish any new information to the sites. This is an indication that users have more interest in the site for viewing and contributing to its content. Semlog uses the RNA, which is an extension of the RSS aggregator. RNA (RSS Aggregator Service) has numerous functions including registration and loading of RSS, building of RSS tree, redistribution of RSS, content clipping, trackback tracing, updating and sanitizing as well as caching RSS (Ohmukai, Takeda et al. 2004).

Web Scripter enables general users to assemble reports quickly and easily; this is by fusing and extracting information from many, heterogeneous semantic sources of the Web (Yan, Frank et al. 2003). In addition, it improves Weblogs. This involves viewing of the posts in two columns in the Web Scripter. Furthermore, the columns are used for semantic mark-ups concerning the posts. On the other hand, Web Scripter supports RDFS (Resource Description Framework Schema) authoring report. In addition, Web Scripter benefits the users who may have the same interests in several blogs and this provides users with additional features. After creating the reports, it matches with other people's Semantic Weblog (Yan, Frank et al. 2003). On the other hand, Semantic Blog contains the same features as Weblog. It structures the unstructured data in Weblog. Furthermore, blogging is a great tool to use for writing articles on a specific topic and share it with everyone via the Internet. This includes having them more structured by Semlog. However, it is not built to allow viewing of ontology or to archive previous ontologies. It is intended more for all the users involved to discuss rather than to view an ontology and comment on it.

2.3.2.3 Community Forum

Community forum is an informal social network found in numerous Websites on the Internet for many topics and reasons. This began with Usenet, which is the most famous technology available since 1979. The Internet helps these community forums to become widespread. Online Forum is virtual communication which begins with a thread that contains information, requests or questions. Further replies follow the messages sorted in a descending manner. Every community forum concentrates on debating a certain area and building the discussion for several purposes which include sharing information, coordination and providing emotional support. A forum could be private or public. Public forums are for use by everybody. Private forums have restrictions and are common within internal organisations connections. The users have certain specific roles. This could be as administrators, members or moderators with writing and reading permission (Harth, Breslin et al. 2004).

Community forums usually target specific people with similar interests. Nevertheless, these societies exist within the boundaries of their sub-category interests. This begins with a general cross posts phenomenon. Cross posts are articles or posts which users think could be of any interest to other communities or sub-groups. Consequently, Traditional Social Network Analysis fails to recognize that these people know each other because they are not visible to one another. Therefore, in order to obtain a solution that will enable people to link to each other and remain within their interests, there is a need for more network structures. Certain domain knowledge can be made available to users through the use of ontology; this offers a link between persons with interest in a similar topic (Malzahn, Szeini et al. 2005). Moreover, community forums are used for asking questions and sharing knowledge on topics raised; therefore, they should be helping the ontology to evolve further. However, there is no systematic link between ontology and community forums, and this presents a challenge. What is needed is a means of viewing, editing and allowing the ontology to evolve after a discussion has occurred. Therefore, this research will look into the link between ontology evolution and social networks and the technology which will allow the ontology to evolve, since currently this is lacking.

2.3.2.4 Ticketing Support System

The Ticketing Support System is used to address issues that are currently in many industries such as banks, government ministries and Web host. The Ticketing

Support System plays a significant role in international banks like Citibank because it improves efficiency and staff productivity. In addition, Web host organisations such as Digital Pacific find this to be an excellent method for supporting their clients; this is because Web host organisations are at different locations. This leads to a need for virtual support that the Ticketing Support System exemplifies well. The Ticketing Support System is used for a specific reason: to resolve the issues of the clients in their Web hosts, for example. Nonetheless, it does not offer an informal open discussion where the staff are able to discuss any issues which they experience in their work. This restricts the use of the tool in many instances.

2.3.2.5 Lightweight Community Driven Tools and Semantic Web

To automate and convert users' information, which is shared for a better meaningful base of knowledge, Lightweight Community-driven Tools together with the Semantic Web Technology are used. This is through involving the Semantic Web and the ontology. More so, the sincerity of the Lightweight Community Driven Tools and trustworthiness of its knowledge artefact is an issue in these tools. In addition, the high levels of information created with the Lightweight Community-driven Tools make it difficult to retrieve information that users are seeking. Consequently, ontology and Semantic Web can be used convert information from implicit data to explicit data (Dietrich, Jones et al. 2008). Nevertheless, there is no consideration of any change to the information. It is important to combine social networking and Semantic Web into a process after the conversion of information into explicit data.

2.4 Critical Evaluation of Existing Approaches: An Integrative View

This section extends the critical assessment of existing approaches to ontology evolution as reviewed in previous sections. The Internet is a necessary technology which most people incorporate in many aspects of their daily lives. In the earlier era of the Internet, the sharing of information and knowledge was a one-dimensional task. Users of Internet access the virtual world of Web to retrieve knowledge without being able to directly contribute to the updating process of the knowledge. In time, with the advent of Web 2.0, the Internet was transformed into a communal world of users where information and knowledge are shared in a collaborative manner. With the benefits of the Web 2.0, the involvement of users in processes such as ontology evolution is feasible. In the previous chapter it was established that knowledge captured in ontology needs to be updated to reflect any changes in the knowledge itself. In other words, the business needs to have an updated ontology as a process that involves every user. Hence, users can be actively involved in the process of ontology evolution. This improved ontology is satisfying for users because they participate in the ontology evolution process. In addition, without the involvement of different users who can communicate and participate in the evolution process of ontology, the updating of ontology will be a much slower process. This makes the discussion of ontology evolution vital and a platform to allow this process to occur is necessary. Therefore, the main issues of the existing approaches can be summed up as:

- Community users are not involved in the ontology evolution process.
- No ontology evolution platform is available that allows the ontology to evolve.

2.4.1 The need for Users Involvement in the Ontology Evolution

Issues in various organisations often arise when staff members are not able to take part in the process of evolution ontology. This includes the challenges that the staff face as they attend to their daily chores. For this reason, it is necessary to involve all participants in order to have an updated and better ontology. For instance, offshore staff that deal with gas and oil industries are faced with several issues every day (Thorsen and Rong 2008). Based on their experience, they could be major contributors to the changes required in ontology in the gas and oil industry. In many papers previously reviewed, the writers examined the issues relating to ontology evolution. They created tools so that they can overcome them through automating the process more. However, the writers did not consider the involvement of the oil and gas offshore staff. They provide them with a chance to take part in the changes needed to be made to the ontology by, for example, increasing the quality of the ontology and making this ontology consistent. The community forum is a section of Web 2.0. They provide users with opportunities to add more information and they give their feedback on specific topics. However, community forums – just liker other social networks – have not been applied to support ontology evolution.

2.4.2 The Need for a Platform for Ontology Evolution

With the increasing need to involve the wider community in the ontology updating process, a common medium for exchange of knowledge in an informal mode becomes a necessity. The medium serves as an open space for its users to share knowledge on specific issues leading to improvement of knowledge captured in ontology. In order to ensure efficiency, the same medium should include a feature that allows the ontology to be edited.

Several researchers have developed plug-ins for Protégé (Liang, Alani et al. 2005; Plessers, De Troyer et al. 2007; Zablith 2008) intended to automate and semi automate the ontology evolution. Nevertheless, they lack a suitable platform for communication and for making decisions according to accrued information. Weblogging provides its users with a flexible means of raising any issue regarding ontology (Fun and Wagner 2007). In addition, weblogging offers friendly interfaces, which attract more users and contribute to the evolution of ontology. However, Weblog has not been applied for ontology evolution.

2.5 Summary of Critical Evaluation of Existing Approach

Table 2.1 summarises the literature review and the concepts that have been discussed. Moreover, it presents the features of the proposed platform and the issues that it will be addressing. The table below presents further details:

Features	**Web-Protégé**	**Protégé Plug-ins**	**Visual Presentation**	**Knowledge Management**	**Wiki**	**Blog**	**Community Forum**	**Ticketing System**
Web-based	✓				✓	✓	✓	✓
Provide Platform to Discuss Ontology Changes	Not fully functional				✓	✓	✓	✓
Support Collaborative Communication	Not fully functional			✓	✓	✓	✓	✓
View Ontology	✓	✓	✓	✓				
Ontology Archive	✓	✓	✓	✓				
Total Score	**60%**	**40%**	**40%**	**60%**	**60%**	**60%**	**60%**	**60%**

Table 2-1: Summary of each tool and features it provides in relation to the proposed platform

Five features have been presented and analysed in Table 2.1, the achievement of which is the aim of this research. Each has been given an equal 20% weight. This table indicates that the majority of the concepts provide 60% of the features needed by the proposed platform. Hence, Web-Protégé and Knowledge Management do not have a platform that is suitable for hosting a discussion on ontology changes. Although, the four social networking tools are equipped with the platform, it does not support any features required for ontology evolution. Finally, regarding Protégé Plug-Ins and a Visual Presentation, each has 40% of the five features needed by the proposed platform

The proposed platform will take into consideration all five features presented in Table 2.1. This gives it a 100% score, and this includes developing a platform that will enhance collaborative communication. Consequently, this will lead to the early detection of any errors or inconsistencies within the ontology. Moreover, it will reduce the incidence of tardy or incorrect actions taken by staff. Therefore, it will increase productivity and reduce maintenance costs.

2.6 Conclusion

This chapter has reviewed the existing Protégé plug-ins that have been developed to automate and overcome ontology evolution issues. It has shown that these plug-ins have succeeded in further automating ontology evolution. However, the process lacks a means of collaborative communication. On the other hand, social networks (informal and formal) have opened further doors to communication but have not been used with ontology evolution. The chapter also introduced different social networking tools. In the next chapter, the problem statement for the ontology evolution process and the user involvement will be presented.

2.7 References

Aseeri, A., P. Wonghtongtham, et al. (2008). Towards Social Network based Ontology Evolution Wiki for Ontology Evolution. 10th International Conference on Information Integration and Web-based Applications & Services, Linz, Austria.

Bechhofer, S., I. Horrocks, et al. (2001). OilEd: A Reason-able Ontology Editor for the Semantic Web. KI 2001, Vienna, Austria

Blundell, B. and S. Pettifer (2004). Abstract: Graph Visualisation to Aid Ontology Evolution in Protégé. 7th International Protégé Conference Bethesda, Maryland.

Dave, B. and L. Koskela (2009). "Collaborative knowledge management--A construction case study". Automation in Construction**In Press, Corrected Proof**.

Dietrich, J., N. Jones, et al. (2008). "Using social networking and semantic web technology in software engineering - Use cases, patterns, and a case study". The Journal of Systems and Software**81**(2008): 2183 - 2193.

Ebersbach, A., M. Glaser, et al. (2008). Wiki Web Collaboration. Heidelberg, Springer-Verlag Berlin.

Fun, R. and C. Wagner (2007). "Weblogging: A study of social computing and its impact on organisations". Decision Support Systems**45**(2008): 242 - 250.

Gennari, J. H., M. A. Musen, et al. (2003). "The evolution of Protégé - an environment for knowledge-based systems development". International Journal of Human-Computer Studies**58**(2003): 89 - 123.

Haase, P., J. Volker, et al. (2005). "Management of dynamic knowledge". Journal of Knowledge Management**9**(5): 97 - 107.

Harth, A., J. Breslin, et al. (2004). Linking Semantically Enabled Online Community Sites. Galway, Ireland, National University of Ireland.

Huang, T., S. Cheng, et al. (2009). "A blog recommendation generating mechanism using an SBACPSO algorithm". Expert Systems with Applications**36**(2009): 10388 - 10396.

Liang, Y., H. Alani, et al. (2005). Ontology Change Management in Protégé. AKT DTA Colloquium, Milton Keynes, United Kingdom.

Malzahn, N., S. Szeini, et al. (2005). "Ontology Facilitated Community Navigation – Who Is Interesting for What I Am Interested in?" Modeling and Using Context **3554**(2005): 292-303.

Noy, N. F., A. Chugh, et al. (2006). A Framework for Ontology Evolution in Collaborative Environments. 5th International Semantic Web Conference 2006, Athens, USA.

Noy, N. F. and M. Klein (2003). A component-based framework for ontology evolution. In Workshop on Ontologies and Distributed Systems at IJCAI-03, Acapulco, Mexico.

Noy, N. F., S. Kunnatur, et al. (2004). Tracking Changes During Ontology Evolution. 3rd International Semantic Web Conference (ISWC2004), Hiroshima, Japan.

Ohmukai, I., H. Takeda, et al. (2004). Metadata-Driven Personal Knowledge Publishing. ISWC2004. Hiroshima, Japan**:** 591 - 604.

Plessers, P., O. De Troyer, et al. (2007). "Understanding ontology evolution: A change detection approach". Web Semantics: Science, Services and Agents**5**(2007): 39 - 49.

Potthast, M., B. Stein, et al. (2008). Automatic Vandalism Detection in Wikipedia. European Conference on Information Retrieval. Glasgow, Scotland.

Searls, D. and D. Sifry (2003). "Building with Blogs". Linux Journal **2003**(107): 4.

Siorpaes, K. (2007). Lightweight Community-Driven ontology Evolution. 6th International and 2nd Asian Semantic Web Conference, Busan, Korea.

Sure, Y., M. Erdmann, et al. (2002). OntoEdit: Collaborative Ontology Development for the Semantic Web. International Semantic Web Conference 2002 (ISWC'02), Sardinia, Italy

Thorsen, K. and C. Rong (2008). Data Integration in Oil and Gas at Norwegian Continental Shelf. 22nd International on Advanced Information Networking and Applications - Workshop, Okinawa, Japan.

Trevathan, V. (2006). A Guide to the Automation Body of Knowledge. Research Triangle Park. NC, USA.

Tudorache, T., J. Vendetti, et al. (2008). Web-Protégé: A Lightweight OWL Ontology Editor for the Web. OWL: Experiences and Direction (OWLED) Workshop, Karslruhe, Germany.

Vrandecic, D., H. S. Pinto, et al. (2005). "The diligent knowledge processes". Journal of Knowledge Management **9**: 85 - 96.

Weichselbraun, A., A. Scharl, et al. (2007). Capturing Ontology Evolution Processes by Repeated Sampling of Large Document Collections. On the Move to Meaningful Internet Systems 2007: OTM 2007 Workshops. Vilamoura, Portugal.

Yan, B., M. Frank, et al. (2003). WebScripter: Grass-Roots Ontology Alignment via End-User Report Creation. ISWC2003. Sanibel Island, Florida, USA**:** 676 - 689.

Zablith, F. (2008). Dynamic Ontology Evolution. International Semantic Web Conference (ISWC 2008), Karlsuhe, Germany.

Zhou, Y. and J. Davis (2007). Analysis of Weblog Link Structure - A Community Perspective. WEBIST 2005/2006, Miami, USA.

Every reasonable effort has been made to acknowledge the owners of copyright material. I would be pleased to hear form any copyright owner who has omitted or incorrectly acknowledged.

Chapter 3 – Problem Definition and Solution Requirements

3.1 Introduction

In previous chapters, important premises on which this research is built upon are described. The first chapter sheds light on the significance of a timely and reliable ontology evolution in business context. Hence, an overview of concepts used in this research, which are ontology definitions, editor and its evolution, are introduced. The chapter discusses the advantages and problems of ontology evolution and motivation and the objective of this research. The first chapter concludes with the thesis structure, briefly describing the contents of each chapter. Chapter 2 focuses on the theoretical grounds for the means used for ontology evolution and several Web-based communication platforms. The literature survey on various ontology editors and different types of social network is presented. The chapter also provides critical evaluation on existing approaches.

This chapter focuses on elaborating the research problems and the approach to overcome these problems. The chapter begins by defining the problems at hand and then presents an overview of the research problems. This corresponds with the research issues related to solving the problems. In order to provide a comprehensive understanding of this research, Chapter 3 also presents the oil and gas industry as a case study. The oil and gas industry, appropriately chosen to epitomize the industry's unique characteristics of the much needed ontology evolution is introduced, followed by the solution proposal, and the choice of research method to elucidate the research problems.

3.2 Problem Overview

These days, businesses are overflowing with information and knowledge coming from various sources such as media, Internet and other business. This enormous amount of data needs to be efficiently managed to ensure that managers make informed business decisions which allows the business to generate profits. This has led some organisations to develop a more sophisticated system in order to manage the large amount of valuable data. Ontology is seen as one of the solutions that serves business purposes (Fox, Barbuceanu et al.; Maier, Schnurr et al. 2003). However, factors that affect the business are always changing and frequent updates to the ontology are required (Maedche, Motik et al. 2002; Zhu, Turner et al. 2004). The process of continuous adaptation of new information and knowledge is known as 'ontology evolution'.

Updating the ontology is particularly challenging when it comes to finding the best way to implement the change, as briefly outlined in the previous chapter. Within an organisation context, ontology evolution is ultimately the result of collective rather than individual efforts, since ontology represents the domain knowledge of users (Gendarmi and Lanubile 2006). In order to ensure that valid output of ontology evolution is achieved, the process needs to be carried out in a participatory manner. This means common agreements are favoured over directive instructions for ontology evolution. The necessity to have mainstream commitment on the course to improving ontology becomes apparent with the fact that the improved ontology will be used by users (Braun, Schmidt et al. 2007). Thus, users have an interest in being kept in the loop because they will reap the benefits.

Most existing ontologies are designed and maintained by individuals or small groups of ontology experts, not actual ontology users (Braun, Schmidt et al. 2007; Siorpaes and Hepp 2007). This is effective if the ontology users can contribute to the process of creating and maintaining the ontology they use. Although Semantic Web has been created in a readable format for machines, humans require the necessary skills and expertise to maintain the ontology (Gendarmi and Lanubile 2006). Consequently, all involved users need to acquire specific skills and must be located in the same country or city. This leads to another problem, namely remote collaborative communication issues (Koschmider and Oberweis 2005). In other words, it is a question of developing effective and efficient ways to improve the communications over ontology evolution.

3.3 Problem Facing Community Driven Ontology Evolution

In the previous section, the need to have a joint effort on ontology evolution is well established. In this section, the key problems within this area of research are identified and explained, providing a self-constructed definition at the outset. There are two key problems that need to be addressed related to the issue of how ontology is amended:

- Community users are not involved in the evolution process.
- No ontology evolution platform is available.

3.3.1 Community Users Support Problem

Definition: *Users Support Problem* refers to providing an opportunity for all users to communicate with each other and raise or share concerns regarding the current ontology. As new information consistently flows into every organisation, appended with individual interpretations and knowledge, ontology evolution is undeniably a task requiring a team effort rather than an individual one. This demands users’ active participation in the process of changing the knowledge captured in ontology. The active involvement of users is essential in order to reduce misunderstandings and possible conflicts over the outcome, i.e. evolved ontology. However, this issue is not simple to tackle due to differences in firm size and specific industry. Especially for large organisations, communications amongst employees are typically segregated according to management level. This also creates further problems since all employees cannot participate in the development process within the organisation. Users of ontology are required to utilise the evolved ontology, while the process of updating the ontology is conferred to higher management level. The top-down approach is more likely to leave users disoriented about the ontology; this issue could be resolved by adopting a participatory approach. The latter actively incorporates wider users’ knowledge throughout the process, resulting in a better understood ontology. There are industry sectors that have to perform their operations from various locations due to the nature of their business. For example, the employees within the oil and gas industry are working from locations far and wide so it is almost impossible to allow everyone to participate in the development of ontology. However, it is necessary to involve as many users as possible since it would diminish the subsequent challenges in using the ontology. This issue of a demanding interaction amongst business actors is recognised as an informal community support problem.

Definition: *Informal Community Support Problem* refers to informal correspondence of users that allows them to reach a decision on a related issue that has been raised. This means the process starts from a basic discussion of an issue and ends with reaching a decision on this particular issue. The whole process does not mandate a unanimous final decision; rather, it emphasises rigorous communication as the pathway to the decision.

3.3.2 Platform for Ontology Evolution Problem

Definition: *Platform for Ontology Evolution Problem* refers to a place or stage which plays the virtual host to the ontology evolution process. As described above, it is essential to provide a communication forum for users ahead of ontology evolution. Most of the current systems or plug-ins concentrates on automating the ontology evolution process; however, they fail to provide a platform allowing informal and/or formal discussion to take place. Results of such discussion are considered as input for ontology to be evolved, placing a significant emphasis on the usefulness of such a platform. Hence, the platform should provide features that support ontology evolution. Further, it should enable users to raise issues as well as make comments which will increase the flow of data to be viewed and discussed in an instant manner. A timely transfer of information is important in many business sectors as any delay may cost an organisation millions of dollars, especially for businesses that are heavily affected by time critical factors. The financial sector is a case in point. A slight movement of a stock price in Japan, for instance, will influence the performance of regional capital markets as well as markets outside the region as major financial markets are interlinked. The failure to have accurate and relevant information will lead to bad decision making and loss of profits.

3.4 Underlying Research Issues

The research problems as stated in the previous section correspond with underlying research issues that are defined in this section. We have identified two research issues as follows:

- Lightweight Community Support.
- Ontology Evolution Platform.

3.4.1 Research Issue 1: Lightweight Community Support

Lightweight Community Support refers to issues related to the lack of informal communication between remote users. It should be noted that once an informal communication is realised and information becomes concrete, a formal communication is established. The argument for engaging this vital constituent is the substantial and essential amount of information that the users attain, in particular those associated with their specific role. Users engaged in technical activities have relevant expertise and skill background that their feedbacks are more justifiable on issues requiring technical input. On the other hand, professionals with managerial skills have more weight on their opinions on administrative issues. Within this context, involving every user to actively participate in the evolution process presents a challenge especially for particular type of business sector. An example of such sector is oil and gas industry.

The nature of working in distant locations in the oil and gas industry presents to staff communication obstacles and complication with regards to timing. Updated information has to reach the intended users in a timely manner; otherwise, productivity and overall organisation performance are affected. Other issues include lack of equal opportunity for all staff within the oil and gas industry to raise and discuss issues. For example, the staff who work on the drilling operations on offshore sites and the management which has offices on the onshore site, have different access to different resources which leads to unequal opportunity for all the staff (Thorsen and Rong 2008).

All staff within the oil and gas organisations are considered in this research as each one works in a different environment and situation which has different perspectives on the existing processes. Staff benefits from various educational backgrounds as well as the experience accumulated over their working years. Together with their various reactions to particular situations or problems that arise during their employment, this contributes to the development of corporate knowledge within their particular area. In addition, the massive information that needs to be efficiently managed in this particular industry has affected business activities as indicated earlier. Therefore, each staff member within the oil and gas industry should have an opportunity to raise issues or concerns within the current processes and which should allow everyone to voice an opinion on an issue that has been raised.

3.4.2 Research Issue 2: Ontology Evolution Platform

Ontology Evolution Platform refers to issues related to developing a stage or venue to support the virtual communication between remote users in order to allow the ontology to evolve. This means providing an outlet that encourages a discussion to take place where participants can raise issues, discuss and vote on them - in other words, an ontology evolution platform. Hence, the oil and gas industry is provided as an example in order to better understand this research issue.

Staff in the oil and gas industries are currently having issues with keeping their processes updated in order to allow their employees to access the most updated information and reduce the incidence of error. This might cost the industry millions of dollars due to the high cost of operations (Thorsen and Rong 2008). The issues are more complicated especially in larger organisations that have more distributed staff working from various locations. Offshore staff are continually involved with problems and challenges related to their daily mining activities. These problems are inherently related to knowledge captured in ontology, which the offshore staff are unaware of since the existing ontology is neither recorded nor documented by offshore staff. The onshore staff are required to maintain these records and document any revisions that may occur to the ontology. Nevertheless, these onshore staff are not exposed to day-to-day onsite activities; thus, they are unaware of their offshore colleagues' need for timely information. In hindsight, it is an issue of management of knowledge captured in ontology among different divisions and sites within one business entity.

3.5 Solution Requirements

Having defined the two key research issues, solutions should be sought and incorporated to overcome the existing challenges. A Lightweight Community-driven approach is proposed and in this section we give two fundamental requirements that need to be satisfied:

- Requirement of Lightweight Community Support.
- Requirement of Ontology Evolution Platform.

3.5.1 Requirement of Lightweight Community Support

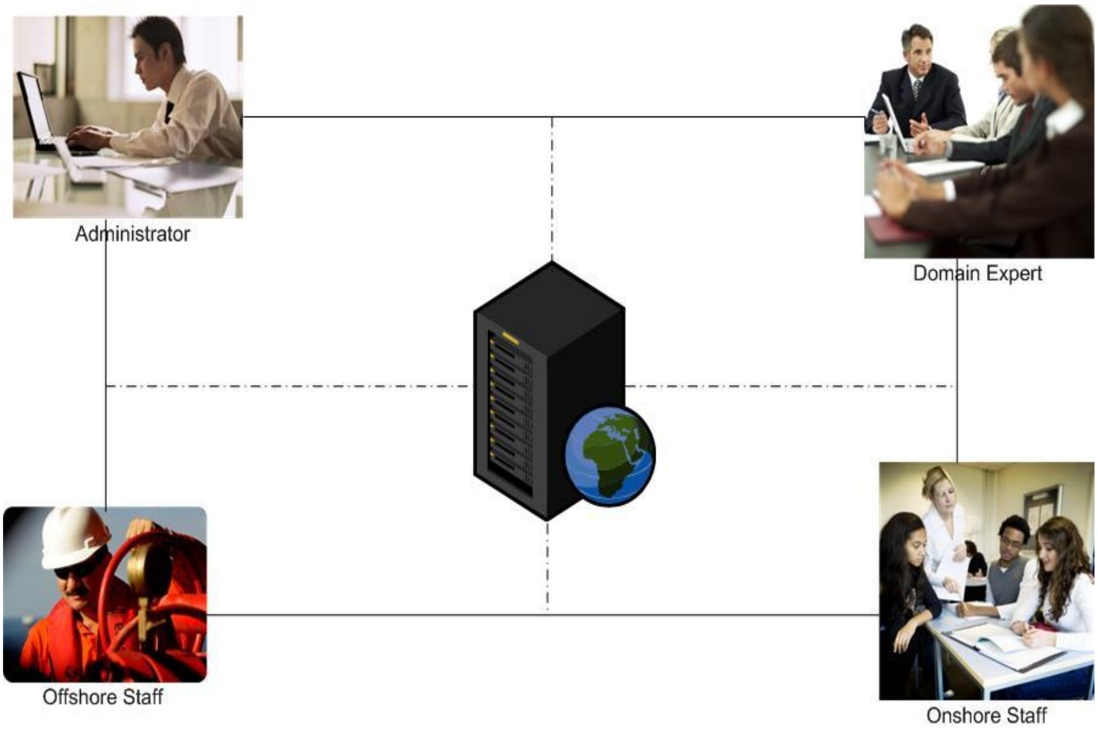


Figure 3-1: Ontology Evolution Platform.

As shown in Figure 3.1, a business entity is comprised of employees working in different roles as well as from different locations. The administrator and domain expert exemplify the two types of tasks, which are administrative and technical tasks. Administrative tasks, performed by administrators, include activities that support the business such as management of employment records, documentation of any information and shared knowledge. Technical tasks, carried out by the domain experts, focus on activities directly related to the business such as labouring tasks on natural resources extraction for a mining unit, provision of financial advice for a financial consulting, and trade of goods and services for retailers.

The Offshore and Onshore Staff illustrate the different locations of employees which is significantly relevant to a particular industry such as the mining sector. Within this industry, business is carried out from various cities and/or countries. BHP Billiton (Billiton n.d.), for example, is one of the largest mining organisation with its headquarters in Melbourne, Australia, while its management office is in London, United Kingdom. From the organisation Website (Billiton n.d.), it is reported that the marketing offices spread around the world, among others, in Germany, Antwerp (Belgium), Pittsburgh (USA), Andar Barra de Tijuca (Brazil), and Tokyo (Japan). With regards to mining, the extraction of petroleum is managed by offices in Zamzama (Pakistan), Angostura (Trinidad), Gulf of Mexico, and Western Australia. In addition, these organisations concentrate on minerals explorations that are administered by offices in Johannesburg (South Africa), Western Australia, and Singapore. Other natural resources that have become part of BHP’s business are ore, base metals, aluminium and coal.

For mining industry, employees working from different locations are not physically connected but are able to converse virtually through the proposed platform to share the knowledge captured in the ontology. The solid lines between each site in Figure 3.1 show remote contact and the dotted lines represent the interaction involved. This supports the exchange amongst all users of information, knowledge, and ideas. The platform provides equal opportunity for all users to participate in the evolution process. Using the platform, each user is granted access to actively participate in a discussion forum on a particular issue. Users with particular knowledge on the core business activities are able to interact with those with administrative tasks and vice versa. This allows knowledge to flow smoothly and reduce any misinterpretation of the outcome of the updated ontology.

3.5.2 Requirement of Ontology Evolution Platform

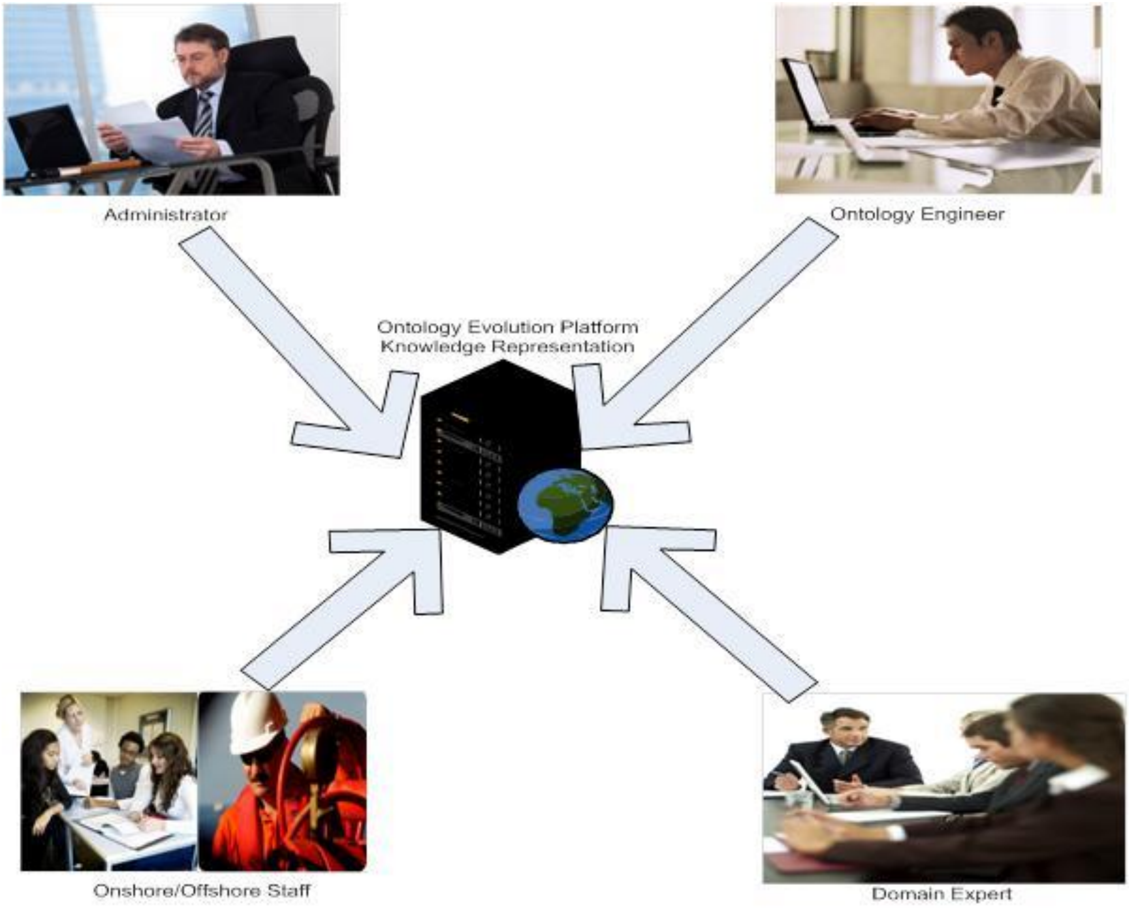


Figure 3-2: The knowledge representation for Ontology Evolution Platform.

An ontology evolution platform requires knowledge representation. As shown in Figure 3.2, the ontology evolution platform allows the staff to raise, discuss, vote and edit the ontology which is presented to them. For example, the domain experts who have knowledge representation are able to nominate issue that lead to a review process of the existing ontology. Other users, regardless of their roles in the business or their working locations, are encouraged to contribute to the discussion held during a limited time frame. At the end of the allotted time, each user casts his/her vote based on the knowledge acquired from the exchange of ideas. The vote is weighted differently depending on the users' role in the organisation. Domain experts are assigned the highest value since the members of this group have the highest credibility on the raised issue.

Users vote on any necessary change to the existing ontology; this includes retaining the existing ontology. Once the final decision has been made, if it relates to revising the ontology, the actual change is carried out by ontology engineers. Ontology engineers play an essential role in the ontology evolution process, since they have the necessary knowledge and skills. The domain experts and administrators are not capable of performing updates on the ontology, nor to visualise the effect of their views and opinions on the existing ontology. Ontology engineers are tasked with both responsibilities that resulted in evolved ontology when required.

3.6 A Case of Oil and Gas Industry

In order to provide a clear and thorough description of this research, organisations which are part of the oil and gas industry are described in this collective case study. In general, the oil and gas industry is a sub-sector of mining industry, and has particular features in performing their business practices. Within this business sector, operations are managed from offices that are likely to be located around the world (Pinde 2001). This industry, as sighted earlier in previous chapters, employs people to work together on geographically scattered workplaces. Activities directly related to the extraction of oil and gas are carried out on site where oil fields are situated, since oil is extracted from the ground or deep sea. Regardless of the exact point of extraction, organisations established offices in close proximity to the oil fields to manage these activities, i.e. onshore office. On the other hand, oil and gas organisations also have offshore offices where activities that are related to management and coordination of business not directly linked to oil and gas extraction are performed. These onshore and offshore offices are most likely to be dispersed in different cities, countries, regions even continents.

Individual organisations in the oil and gas industries are typically unlikely to participate in the same ontological systems. Communication and classification systems are likely to be developed within individual organisations and will reflect the histories, preferences, and needs unique to each organisation. Yet as all organisations active in oil and gas extraction share similar operational and economic considerations, it is anticipated that similar patterns in ontological construction, performance, and evolution are shared among them. This is especially likely among Administrator groups, where managers and systems specialists are more likely to be engaged in the exchange of information with their counterparts in other organisations. Data exchange is likely to occur among users who share similar responsibilities and who strive to prevent or resolve conditions where workers might be put at risk.

Conversely, among individual organisations in the oil and gas industry, it is highly possible that many users are not involved in knowledge sharing. The poor interactions amongst staff and offices often lead to significant delays or wrong decisions in the daily activities of staff, resulting in great losses. Data that are collected and maintained by onshore offices are not being transferred in a timely manner to the offshore offices, creating a haphazard and ineffective communication process between the two offices. This has affected productivity within the industry, since misinformation on crucial drilling data will lead to over/under drilling (Thorsen and Rong 2008). For example, information about a sudden temperature change will compel staff in the offshore office to postpone drilling activities. This affects the safety of the offshore staff. Therefore, due to daily critical operational work, it is vital to receive the information instantaneously to reduce or prevent any errors that may occur.

The issue of users' involvement also emerges from the fact that staff are working from substantially different physical areas ranging from a business hub in the capital city of a country to very remote areas such as in the middle of an ocean. This makes sharing of knowledge an even more challenging task between the onshore and offshore staff. Offshore staff work in conditions different to those of the onshore staff, and they examine and report on quite different issues in their daily work (Thorsen and Rong 2008). There is a need to integrate relevant information between the onshore and offshore branches. Failure to do so leads to a loss of vital information that can be transformed into useful knowledge and reduce overall costs. For example, there are issues related to certain proportions of oil and water during the drilling process that have to be maintained by the offshore staff to ensure high quality oil. A high percentage of water will have an impact on the quality of drilled oil. If the offshore staff persist in drilling and low quality oil is extracted, machines that have been used to drill based on false information will have less useful life.

In order to overcome the demanding task of interchange of ideas and knowledge, oil and gas organisations should provide a platform that will enhance the process of evolving knowledge amongst all users. This collaborative virtual communication is needed in order to develop and change the ontology when required.

3.6.1 Oil and Gas Ontology (OGO)

Ontology has helped to standardise the business processes within the related field that is typified by dispersed location of offices. The scattered sites pose a challenge in providing systematic interaction amongst information users and timely data transfer. In order to simplify the business process and reduce operational work, Oil and Gas Ontology (OGO) is used. In other words, ontology has helped oil and gas organisations to achieve dataflow efficiency and efficacy for the staff (Thorsen and Rong 2008). There has been research conducted into standardising ontology languages which has resulted in the development of Ontology Web Language (OWL) by the World Wide Web Consortium (W3C) (Brockmans, Volz et al. 2004). OWL language has been used by many ontology editors, for it treats the ontology in a user-friendly way. In addition, there are many research initiatives related to creating ontology for a specific domain, including Oil and Gas Ontology (Association 2008) and Software Engineering Ontology (Wongthongtham 2006).

Oil and Gas Ontology (OGO) is an ontology that has been developed especially for the oil and gas industries, which is an explicit specification of the conceptualisation of oil and gas domain. Currently, at the time of writing, there is only one main oil and gas ontology available (Association 2008), which was developed by Statoil in Norway. Statoil in Norway is one of the oil and gas industries that has been investigating the benefits of using Oil and Gas Ontology such as the AKSIO project (Norheim and Fjellheim 2006) and Norwegian Continental Shelf (Thorsen and Rong 2008). Each of these OGOs is described in detail below.

3.6.2 Active Knowledge System for Integrated Operation (AKSIO) Project

The AKSIO project was intended to develop a knowledge management system to support operations in offshore oil fields. Major operators, such as Statoil, have about 20 drilling projects around the world. This has created some issues in terms of monitoring, analysing or making decisions between projects which are costing the drilling process downtime around $0.5 – $1 million/day. Therefore, knowledge transfer between drilling projects through documented experience best practices and expert references is a vital process (Norheim and Fjellheim 2006).

The AKSIO system has two main functionalities: ontology-based annotation and contextual ontology-driven retrieval of content. These functionalities are created around two use cases. First, capture and qualify knowledge gained in the drilling operation; and second, supply relevant and timely knowledge for the planning of new wells (Norheim and Fjellheim 2006). These use cases have developed due to issues that have been raised in the current approach.

The first use case (producer use case), involves a quality assurance process comprising discipline advisors and experts. This use case involves “clearing experiences not relevant to cross-project reuse, adding annotation from discipline advisors, classifying the information and linking to experts, best practices and actions” (Norheim and Fjellheim 2006).

Second, the consumer use case is one where consumers are involved with either the planning or operations. Therefore, the objective is to find the relevant experiences that can affect their operation. AKSIO provides a search engine to use the shared ontology and find the relevant experience and use this in the current work (Norheim and Fjellheim 2006). This will help in the making of the right decision in real time which will save the downtime and manned power cost.

3.6.3 Norwegian Continental Shelf

The demand for oil and gas has increased in the last decade. This has led to the need to investigate a more efficient way of drilling and completion to reduce the pressure and protect the environment further. Therefore, an Integrated Operation project on the Norwegian Continental Shelf has been researched and implemented by the Norwegian oil and gas industries (e.g. Statoil). This is to further “support operational decisions about offshore installations by onshore control centre, developing common standards, integrated solutions, and new technologies” (Thorsen and Rong 2008).

Integrated Operation (IO) consists of two generations: “(IO G1) is to integrate processes and a person offshore and onshore to improve onshore’ ability to support offshore operations.” (Thorsen and Rong 2008).This first generation was already completed at the time of writing. IO G2 is to help operators utilize the vendors’ competences and services more efficiently than today” (Thorsen and Rong 2008). IO G2 is harder to implement as it requires high level technology, which includes developing an OGO (Thorsen and Rong 2008). The second generation involved utilizing vendor competency more efficiently. This means the second generation Oil and Gas Ontology will be a vital part of the project. Offshore oil and gas industries have massive data which has been gathered over several decades. However, these data can be used in a more organised way to increase the productivity and protect further the environment during the drilling and completion (D&C) process. Therefore, the provision of real-time data will be a valuable asset for these industries (Thorsen and Rong 2008) since it will give valuable information on the fly instead of risking human interaction errors. It will also increase production and allow staff to make decisions during drilling and completion processes (Thorsen and Rong 2008).

In order to be able to achieve this, automated analysis and response as well as prediction of the near future by machines will provide great benefits to the drilling team. This can be implemented by using Semantic Web. Hence, Oil and Gas Ontology will be required. It will further support the transfer of real-time data from offshore and onshore and vice versa and automate key work processes (Thorsen and Rong 2008).

3.7 Choice of Research Approaches

The objective of this thesis is to develop an approach to support ontology evolution that ensures collaborative and effective communication between users that are working in offices scattered in different locations. In order to carry out this development, we need to adopt a scientific research approach. Therefore, this section gives an overview of different research methods and the research method that will be used in this thesis.

3.7.1 Research Methods

There are two broad research approaches: (1) the social science approach, which can be classified into scientific and interpretive research methods; and (2) the science and engineering approach, which is “learning by doing” (Galliers 1991). Below are explanations of each method.

Social Science research focuses on phenomenon related to the social world and applies scientific procedures to produce new knowledge (Neuman 2003). This type of research is categorised into two research areas: scientific research and interpretive research. Scientific research involves gathering data in order to observe and measure a fact. This research is “characterised by repeatability, reductionism and refutability which assume that observations of the phenomena under investigation can be made objectively” (Galliers 1991). This includes laboratory experiments, field experiments, surveys, case studies, theorem proofs, forecasting and simulation (Galliers 1991). On the other hand, interpretive research is based more on concepts and phenomena. It is more open and less measurable than scientific research. It has the possibility of many interpretations for the same phenomena. This includes subjective/argumentative, action research, futures research and role/game playing (Galliers 1991). This research mainly depends on whether the methodology or the approach is well built. However, it does not explain how to create a new approach.

Science and Engineering research (March and Smith 1995; Hevner, March et al. 2004) is driven by the spirit of "learning by doing" and problem solving. This type of research, in subsequent years, was classified as the Design Science Research, which relates to the invention of a new or improved solution technology (Venable 2006). It confirms theoretical predictions (Galliers 1991). It consists of three levels: conceptual, perceptual, and practical.

- Conceptual Level (first level): creating new ideas and new concepts through analysis.
- Perceptual Level (second level): formulating a new method and a new approach through design and building the tools or environment or system through implementation
- Practical Level (third level): carrying out testing and validation through experimentation with real-world examples, using laboratory or field testing since it may lead to a new development, technique or methodology based on a set of concepts which together form a new theoretical framework. It also frequently addresses the questions of what problems need to be addressed and propose a solution.

3.7.2 Choice of Science and Engineering Based Research Method

The aim of this research is to create an improved technology to enhance the communication between offshore and onshore staff. Based on the proposed approach, a tool is developed and validated for its robustness. Therefore, the science and engineering research method is the most appropriate. The oil and gas industry is used in this research as a point of reference to justify the proposed approach and to enable the application of each step of the science and engineering research method. This research, however, does not fall into a case study research as it has different purposes. A case study research, which is a part of the social science research method, revolves around the validity issues of the research object (Gerring 2007). The science and engineering research method is classified into three phases as identified above.

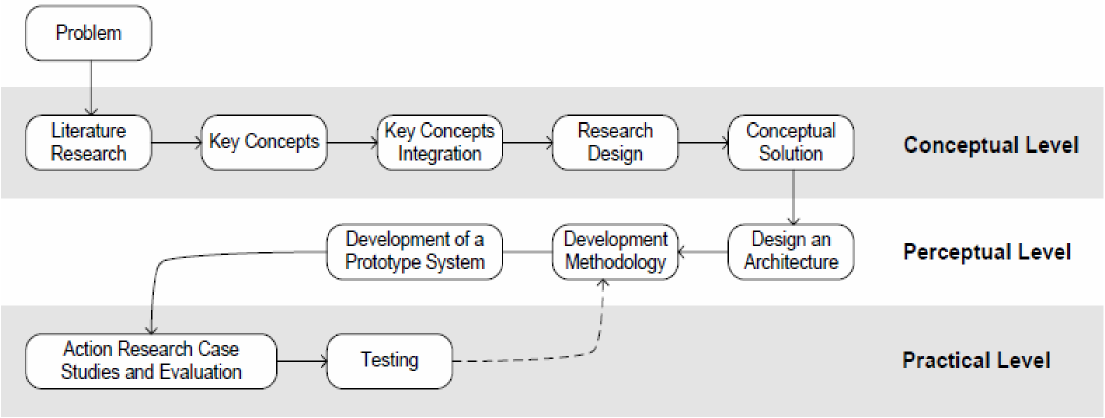


Figure 3-3: An engineering-based research approach (Wongthongtham 2006).

This research starts by identifying the problem and researching the relevant issues by conducting a literature review. This is followed by a thorough analysis of the related topics. An overall research design is developed followed by a conceptual solution. Then, at the perceptual level, the design architecture is developed. This is followed by the actual prototype development of the system. Once the prototype has been developed, further testing is conducted which will identify any need for further enhancement (development) which will undergo an iterative process until the system is completed. The next section describes each phase in more detail.

3.7.3 Science and Engineering Research Method Phases

As discussed, this research will follow three phases. This section will describe each phase and its development in more detail as follows:

Research Phase I: Conceptual Level (First Level)

The conceptual phase is where the problem is defined and the research is justified in terms of its significance. Based on the literature review of ontology evolution and social networks, the gap which this research addresses is presented. Issues faced by ontology evolution and the advantages of using the Lightweight Community-driven approach are also reviewed. In this phase, the Lightweight Community-driven approaches that are available, and previous research that has been conducted regarding applications of this approach, are explored. Lastly, a conceptual design solution has been developed which is considered as one of the most important parts of research. This involves an understanding of the domain and the creation of a proposed solution.

Research Phase II: Perceptual Level (Second Level)

The actual prototype of the Lightweight Community-driven approach is developed at this stage and the setup of Web-Protégé is used to allow the users to view and edit the selected oil and gas ontology. This will be completed by thoroughly studying the design architecture of the system. The development will be an iterative process through which the ontology platform will be tested and evaluated. Based on the outcome of the test, further development will be undertaken.

Research Phase III: Practical Level (Third Level)

Currently, there is one public oil and gas ontology available online on the Internet (e.g. PCA Caesar Association). This ontology will be studied and a sample of an oil and gas ontology will be created. The result will be evaluated against the dataset. Therefore, if any errors are been found, the iteration of the development will start again in order to overcome the issue, or it will be documented for future research consideration.

3.8 Conclusion

In this chapter, we have meticulously described the research problems. The chapter started with an outline and definition of the problems, followed by identification of the research issues, overview of the solution requirement, introduction of oil and gas as the chosen object and summary of research approaches. In the next chapter, we propose the conceptual solution for the issues addressed in this chapter, including the justification for the choice of this conceptual solution.

3.9 References

Association, P. C. (2008). "POS Caesar Association". Retrieved March 16, 2010, from http://projects.dnv.com/reference_data/RD4Browser/default.aspx.

Billiton, B. (n.d.). "Businesses Global Locations". Retrieved August, 10th, 2011, from http://www.bhpbilliton.com/home/businesses/Pages/GlobalOperationsMap.aspx.

Braun, S., A. Schmidt, et al. (2007). Ontology Maturing: a Collaborative Web 2.0 Approach to Ontology Engineering. 16th International World Wide Web Conference (WWW 07). Banff, Canada.

Brockmans, S., R. Volz, et al. (2004). Visual Modeling of OWL DL Ontologies Using UML. ISWC 2004. Hiroshima, Japan**:** 198 - 213.

Fox, M. S., M. Barbuceanu, et al. An Organisation Ontology for Enterprise Modeling. Toronto, Department of Mechnical and Industrial Engineering, University of Toronto**:** 1 - 25.

Galliers, R. D. (1991). Choosing appropriate information systems research approaches: A revised taxonomy. Information Systems Research: Contemporary Approaches and Emergent Traditions, North-Holland, Amsterdam.

Gendarmi, D. and F. Lanubile (2006). Community-Driven Ontology Evolution Based on Floksonomies. OTM Workshop, Montpellier, France.

Gerring, J. (2007). Case Study Research: Principles and Practices. Cambridge University Press, New York, USA.

Hevner, A. R., S. T. March, et al. (2004). "Design Science in Information Systems Research " MIS Quarterly **28**(1): 75 - 105.

Koschmider, A. and A. Oberweis (2005). Ontology Based Business Process Description. Proceedings of the CAiSE´05 Workshops **2:** 321 - 333.

Maedche, A., B. Motik, et al. (2002). Managing Multiple Ontologies and Ontology Evolution in Ontologging. IFIP Congress. Montreal, Canada.

Maier, A., H. Schnurr, et al. (2003). "Ontology-Based Information Integration in the Automotive Industry." Lecture Notes in Computer Science **2873/2003**: 897 - 912.

March, S. T. and G. F. Smith (1995). "Design and naturla science research on information technology". Decision Support Systems **15**(1995): 251 - 266.

Neuman, L. (2003). Social Research Methods: Qualitative and Quantitative Approaches. Pearson Education, Inc. Boston, USA.

Norheim, D. and R. Fjellheim (2006). AKSIO - Active knowledge management in the petroleum industry. ESWC2006 - 3rd European Semantic Web Conference 2006. Budova, Montengro.

Pinde, D. (2001). "Offshore oil and gas: global resource knowledge and technological change". Ocean & Coastal Management **44**(9 - 10): 579 - 600.

Siorpaes, K. and M. Hepp (2007). myOntology: The Marriage of Ontology Engineering and Collective Intelligence. 4th European Semantic Web Conference 2007 (ESWC 2007). Insbruck, Austria.

Thorsen, K. and C. Rong (2008). Data Integration in Oil and Gas at Norwegian Continental Shelf. 22nd International on Advanced Information Networking and Applications - Workshop, Okinawa, Japan.

Venable, J. (2006). The Role of Theory and Theorising in Design Science Research. 1st International Conference on Design Science. Claremont, California, USA.

Wongthongtham, P. (2006). A methodology for multi-site distributed software development. School of Information Systems. Perth, Curtin University of Technology. **Doctoral Degree**.

Zhu, F., M. Turner, et al. (2004). Dynamic Data Integration using Web Services. Proceedings of IEEE International Conference on Web Services.

Every reasonable effort has been made to acknowledge the owners of copyright material. I would be pleased to hear form any copyright owner who has omitted or incorrectly acknowledged.

Chapter 4 – Solution Proposal and Conceptual Framework

4.1 Introduction

In previous chapters, concepts relevant to this research were discussed; in particular, constructive edges and concerns regarding the evolution of ontology in a business were identified and highlighted. The discussions were followed by an examination of various ontology editors and existing approaches to ontology evolution, from which two research problems and key research issues emerged. The two key issues that were identified and discussed in detail in the previous chapter are: (i) Lightweight Community Support and (ii) Ontology Evolution Platform. In Chapter 3, requirements for the solution to ensure its workability are identified, which include a platform and knowledge representation. Both requirements complement each other and are amalgamated on the platform. The chapter then introduced the oil and gas industry which will be used this research as a case study, and concluded with a discussion of the chosen research method.

This chapter elucidates the proposed solution and conceptual framework to address the aforementioned issues. At the outset, an overview of the Lightweight Community -driven Approach is presented. The approach incorporates three key principles which are: (i) conceptual framework of Lightweight Community-driven Approach, (ii) implementation of Lightweight Community-driven Approach, and (iii) validation and verification of the Lightweight Community-driven Approach. The chapter elaborates on the first key principle, the conceptual framework, and sheds light on the advantages and disadvantages of the proposed approach. This chapter further explores two key solutions that will address the issues of the lack of community users' involvement in ontology evolution and the non-existence of an ontology evolution platform. The first solution is the application of Lightweight Community Support that enables users to work collaboratively on updating the knowledge captured in ontology. The second solution is the development of an Ontology Evolution Platform that features various groups of users with respective roles and a ticketing support system.

4.2 Overview of Lightweight Community-driven Approach

The use of a Lightweight Community Driven Approach is crucial for business in engaging users who are located in many dispersed sites. The advantage of the approach is two-fold as thoroughly explained in the previous chapter. Firstly, it allows users with different levels of knowledge to converse and share their information in an informal way. Secondly, it can overcome geographical and time boundaries which are the consequence of scattered office locations in different parts of the world.

In the oil and gas industry, onshore and offshore staff requires an outlet whereby they can engage with others and discuss issues related to their daily activities. Information and knowledge pertaining to these daily activities is captured in ontology and change over time. This requires ontology to be updated accordingly. The proposed medium for staff to be involved in the process of updating ontology takes the form of a platform. The platform facilitates the collaborative participation of relevant users with attention to time constraints and domain reliability of ontology evolution. This participatory approach is designed to address the information asymmetry issue in ontology evolution (Siorpaes 2007). Thus, the central feature of the platform is that it enables each staff member to take part in an ongoing discussion.

In this section, an overview of the developed approach is presented. As shown in Figure 4.1 below, the Lightweight Community-driven Approach consists of three key principles: conceptual framework, creation of a prototype and a validation and verification of the created prototype.

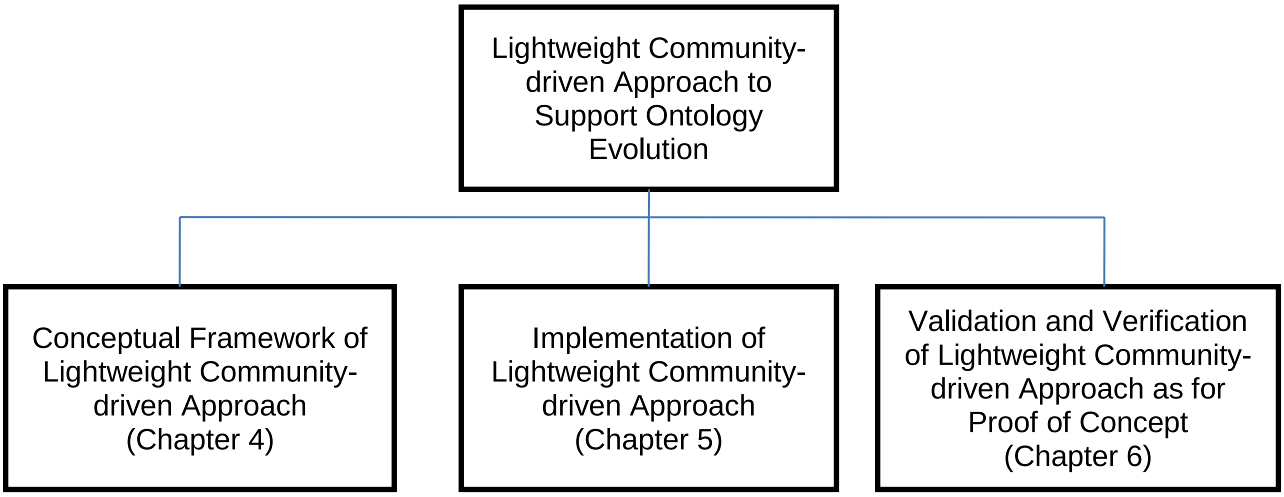

Lightweight Community-driven Approach to Support Ontology Evolution
Conceptual Framework of Lightweight Community-driven Approach (Chapter 4)
Implementation of Lightweight Community-driven Approach (Chapter 5)
Validation and Verification of Lightweight Community-driven Approach as for Proof of Concept (Chapter 6)

Figure 4-1: Graphical Representation of Lightweight Community Driven Approach

The Lightweight Community-driven Approach consists of three key principles. The first principle is the conceptual framework which provides the features and relationship between each component of the approach. The second principle describes the key processes used to develop the platform which will be discussed in Chapter 5. This includes platform construction, showing not only what the platform comprises, but also how the developed platform enables users' involvement in ontology evolution as well as the evolution process itself. Finally, the third principle involves the validation and verification of the Lightweight Community-driven Approach. This includes proofing the concept by developing a prototype and is discussed in detail in Chapter 6. Each of these principles is elucidated in detail in the next section.

4.3 Key Principles of Lightweight Community-driven Approach

The Lightweight Community-driven Approach is divided into three main principles. It starts with a description of the conceptual framework. Then, the key processes of implementation are described. Finally, the validation and the verification of the platform are discussed. These principles are discussed in the following sequence:

4.3.1 Principle 1: Conceptual Framework of Lightweight Community-driven Approach

The development of a conceptual framework is a crucial step in constructing any platform as it has to fulfil all predefined requirements and lucid for implementation. The conceptual framework provides a working strategy and a scheme for the Lightweight Community-driven Approach. It includes the core components of the platform and their interconnection within the context of this research. In accordance with the underlying issues and the solutions' requirements elucidated in Chapter 3, the conceptual framework of Lightweight Community-driven Approach incorporates groups of users, operational and ontology evolution layers. The groupings of users signify the importance of wide range knowledge intrinsically developed throughout the users' academic and professional lives. Each staff is assigned to a particular group based on the issue at hand. This ensures credible people are in the right group that their potential contributions to ontology evolution are properly channelled. The operational layers support this process and reflect the actual exchange of knowledge among users by way of a dialogue. These two layers of the platform are intended to address the security concerns of the users and to ensure that each participant is provided with opportunities to articulate his/her views and opinions. This chapter presents and illustrates the conceptual framework in Section 4.4.

4.3.2 Principle 2: Implementation of Lightweight Community-driven Approach

The process to implement the Lightweight Community-driven Approach involves developing and entering the platform. In the development stage, the features of the platform are incorporated in a developed prototype that is based on heuristic values. Hence, the steps performed are in accordance with the requirements of the aforementioned perceptual level (Galliers 1991). Upon completion, the entering process takes the form of populating the platform with sample data. Examples representing real case scenarios in the oil and gas industry are used to demonstrate the functionalities of the platform. These illustrations address the critical issue pertaining to ontology evolution - that is, ontology evolution undertaken collaboratively (Galliers 1991; Gendarmi and Lanubile 2006; Braun, Schmidt et al. 2007).

4.3.3 Principle 3: Validation and Verification of Platform

Proof-of-concept is intended to validate and verify the proposed system as part of science and engineering based research (Galliers 1991). The platform that consists of social networking, permission and ontology evolution layers is validated in terms of its goals in facilitating communication amongst staff virtually that lead to updated ontology. Proof-of-concept provides a justification for the platform, which includes the lightweight community support and ontology evolution platform. The first relates to the users' involvement in the evolution process and the latter to improvements to the ontology of particular issues. Finally, validation is carried out on the platform to check the consistency and validity of the Lightweight Community-driven Approach. Further details of the implementation stage are given in Chapter 6.

4.4 Conceptual Framework for Lightweight Community-driven Approach

The development of a Lightweight Community-driven Approach is extensively performed based on the previously introduced conceptual framework. In this section, a Lightweight Community Support is presented. The platform is divided into three layers, as shown in Figure 4.2.The platform accommodates all legitimate users which are team members/leaders, domain experts, ontology engineer, administrator, and system auditor. These users are divided into two different groups: the Admin Group and the Contributor Group. The Admin Group includes the administrator and system auditor. The Contributor Group consists of domain experts, an ontology engineer, team members and team leaders. As briefly described earlier, membership of each group - particularly of domain experts - is determined by the issues that are raised. This entails further control through segregation of roles.

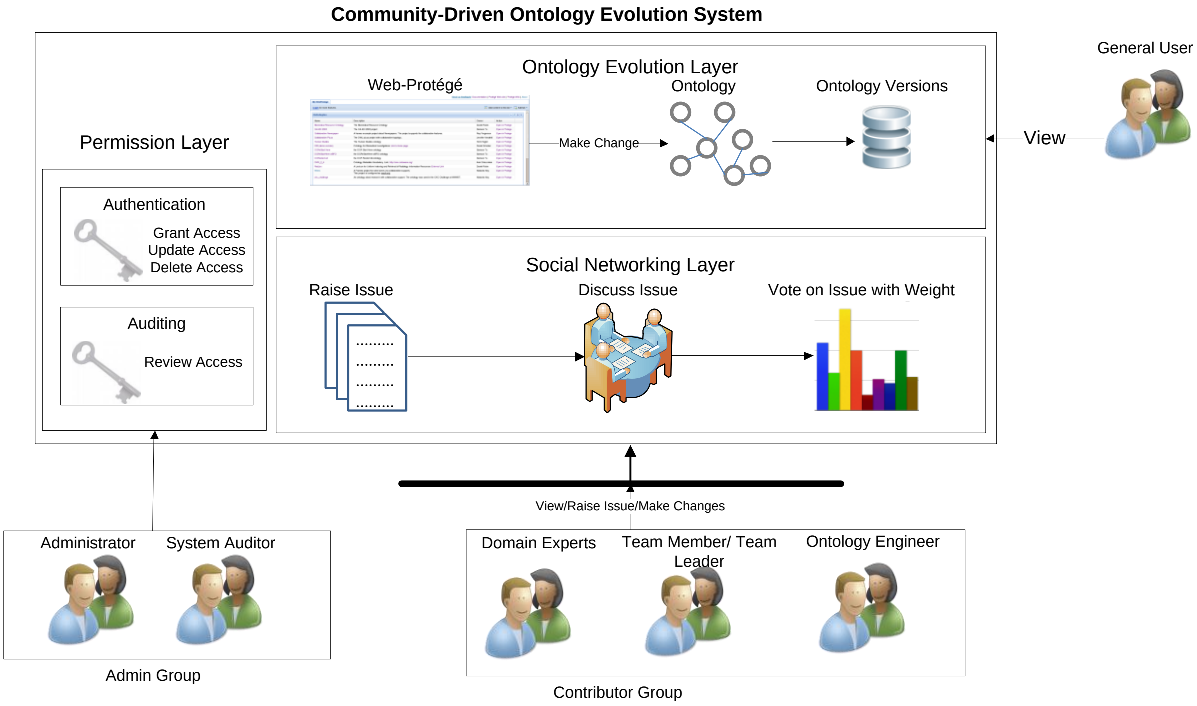


Figure 4-2: Conceptual Framework of the Lightweight Community-driven Approach

Figure 4.2 above shows that the Platform is divided into three layers: (i) Social Networking Layer, (ii) Permission Layer and (iii) Ontology Evolution Layer. The Social Networking Layer is where the domain experts, ontology engineer and team member/leader are able to raise, discuss and vote on an issue. The Permission Layer allows the administrator to manage the access of the users and groups. Finally, the Ontology Evolution Layer is available for all the users to view by using Web-Protégé

(Tudorache, Vendetti et al. 2008). In addition, general users are able to view the ontology in the Ontology Evolution Layer. With this permission, each user is able to essentially see the related ontology that is targeted to evolve. Upon completion of the evolution process, users are able to see the updated ontology. The actual change of ontology is manually performed in this layer by the ontology engineer. Each layer is described in more detail in Section 4.7.

4.5 Advantages and Disadvantages of Using Lightweight Community-driven Approach

A Lightweight Community-driven Approach has been used in many websites. It has provided opportunities to all users to participate in a discussion-based activity dynamically and in an informal way. However, it has several disadvantages which are discussed in further detail below.

	Advantages	Disadvantages
Lightweight Community-driven Approach	1. Open platforms (or an informal platform) which can lead to innovations and new ideas. 2. Due to the nature of informal discussion, it encourages users to participate and discuss their opinions freely. 3. Creates a lot of data within a very short time frame. 4. Enables informal discussion.	1. In some cases, leads to misunderstanding or to a wrong answer or solution. 2. May not be suitable to represent concrete ideas as it aims for informal discussion.

Table 4-1: Advantages and Disadvantages of Lightweight Community-driven Approach

As illustrated in Table 4.1, the Lightweight Community-driven Approach encourages users to discuss issues in an informal way. This may lead to innovation as it allows everyone to raise and share their opinions. The approach also creates an opportunity for information to be exchanged in an efficient way. However, the openness of a Lightweight Community can produce problems such as the likelihood of misunderstanding of the discussion content that may lead to a wrong conclusion. Another problem arises from the interaction style that relates to the appropriate way to raise concrete ideas. These errors can be avoided if the Lightweight Community has a systematic process to follow such as that proposed in this research.

Lightweight oncologies can be applied in the oil and gas industry to streamline communications, identify systems errors, and clarify and classify significant problems. Oil and gas exploration is, by its nature, a geographically diverse practice as extraction is applied throughout different regions with fossil fuel reserves. Remote communication is critical to ensure that organisational goals are met and internal organisational policy is being upheld during off-site decision-making. One example of the value of such practices within the petroleum industry is the Deepwater Horizon oil spill which occurred in 2010 and contaminated the Gulf of Mexico, as the equipment failures associated with this accident were attributed to lack of communication between the rig operators and the rig owners (Goldenberg, 2010).

Efforts to promote communication between geographically diverse locations can be facilitated by establishing a central communications hub, which functions as a single centralized station through which all communications are cleared. Using a Lightweight Community Driven Approach, it is possible to establish an ontological organisational strategy to the information passing through this central hub. This organisational strategy would facilitate communications by establishing a hierarchy of management for all information. Considerations which would be used to inform this hierarchy would include: (a) location of origin; (b) the name and title of the sender; (c) the nature of the message; and (d) whether information pertained to a new or to an established topic. Other considerations might include security concerns, as addressed previously in this chapter.

Communications structured by using the Lightweight Community Driven Approach can be transmitted and received for different purposes. It is probable that formal communications between administration and employees or, among teams, between leaders and followers, will serve as the primary purpose of the communications hub. After the ontological organisational strategy is established, other uses for a central communications hub are likely to emerge. Both formal and informal social networking can be facilitated through this hub, and both of these types of social networking can be used to promote community-building and improve the organisational culture. The central hub will allow all staff to communicate with each other and discuss their daily issues which are raised by other staff within the organisation. When applied as a single communications center, the hub will reduce formal barriers between the staff where it allows them to discuss the issues in an informal style until a decision has been made on a specific issue. Informal discussion can be of great value to organisations, as participants can discuss the issues they are facing within daily operations and activity. The staff will also share and enhance their overall knowledge awareness of the entire staff within the organisation. This also creates a room for innovation and creativity to enhance the overall processes within the organisation.

Updates to the ontological system which governs information organisation within the communications hub can be made to promote improved clarity of communications. Updates must be made on a regular basis to reduce errors in how information is organized, analysed, and disseminated for use within the organisation. For example, in oil and gas organisation, it is vital to have real time data. Data which is outdated no longer reflects circumstances as they are found within the industry, and can contribute to serious problems such as financial loss.. These details can be very vital such as the percentage of water and oil while drilling in an oil field and how deep the drill should be performed and oil be extracted (Thorsen and Rong 2008)., Communication is a vital part in many organisations to be sure the data is updated and save unnecessary loss due to errors which is performed of an out-dated data.

4.6 Lightweight Community Support as a Solution

This section elaborates on the functionality of Lightweight Community Support to answer the research problems as discussed in the previous chapter. As discussed at length in (Klein 2004; Stojanovic 2004; Vrandecic, Pinto et al. 2005; Noy, Chugh et al. 2006), ontology evolution requires the active involvement of users. The reason for this is to ensure the appropriateness and usefulness of the evolved ontology. Initiatives or the need to update the current ontology are introduced by different users such as the domain expert, while the actual revision is managed by the ontology engineer. Throughout the process of updating the ontology, all stakeholders need to be involved and be provided with an outlet to express their ideas, wants and requirements of the updated ontology. The collaborative efforts will minimise any errors or confusion about the new ontology. In other words, the use of the platform for collective communication increases the efficiency of the ontology evolution process.

Consistently, within the oil and gas industry, the active involvement of employees in collective communication to update the ontology is required to enhance the productivity and reduce the incidence of operational errors which leads to substantial savings in operational costs (Thorsen and Rong 2008). Within such a business context, collaborative communication will provide various opportunities for the users to be engaged in discussion regarding the update of ontology, which may affect daily business activities. The three main concepts are: (i) collaborative communication, (ii) committed and responsible participation and (iii) update of ontology by users for users.

4.6.1 Collaborative Communication

Virtual communication has grown dramatically since Web 2.0 (Wang, Xiong et al. 2007) allowing users to communicate with each other, not only to raise but also to discuss issues. The process of exchanging information and knowledge has become more efficient as technology advances further. It overcomes physical and distance limitations of participants in a way that everyone has access to the intended community forums. For example, Virtual Communication is a way of organising and exchanging information and knowledge regardless of the existing distance between the staff. Onshore and offshore staff in the oil and gas industry are required to work together in an efficient manner and be provided with updated knowledge captured in the ontology, even though they are not physically located at the same place. The forum provides more opportunities that allow the entire staff to discuss any issues and share knowledge that consequently leads to evolving ontology by simply logging into the community forum. This includes allowing the data to flow between each branch without any constraints and delays. This is possible by reducing the gap between the entire staff by connecting them virtually. As a result, the physical distance between users is reduced and participation in improving knowledge captured by ontology is increased.

4.6.2 Committed and Responsible Participation

Increasing collaborative communication between users is the initial repositioning required for improved communication. An important aspect of this is ensuring the quality of this communication. While a Lightweight Community Support provides opportunities for users to raise their opinions and judgments on the issues, the quality of communication depends on the users themselves. As they are presented with authorisation to organise a discussion, users are more confident about performing deliberations on a topic requiring domain expertise. This encourages users to be more committed to updating knowledge captured in the ontology. Furthermore, all users are more likely to be committed to issues that are raised by them. It will allow them to take ownership of what they have raised and make the changes necessary in order to bring about a particular decision.

Aside from the increased involvement of users, a Lightweight Community Support contributes to the quality of the ontology, which is also important. An outdated ontology may cause problems that lead to operational errors, increases in costs and lower productivity levels. Therefore, it is essential for staff to participate responsibly in order to ensure the value of the discussion outcome. This means the end product of such collaborative communication should accurately reflect the staff knowledge on a particular topic.

4.6.3 Update Ontology by User for User

The two aforementioned concepts contribute to a more methodical update of ontology, which entails roles and responsibilities. Everyone is not only given equal opportunity to raise and discuss an issue related to existing ontology, but is also expected to engage in discussion with a certain level of commitment and responsibility for his/her own benefit. A commitment to ensuring organised collaborative communication will allow the users to increase their knowledge and build the intellectual human resource. This knowledge will be built and will provide updated information to the users which in turn will increase their job performance and productivity. For each user, the platform provides various roles and, accordingly, assigned different weight on each opinion or vote. Vote is casted at the end of every discussion of an issue to determine if ontology should be updated or not. The roles and the weightings for domain experts, ontology engineer, team member, team leader, administrator and system auditor are presented below.

4.6.3.1 Domain Experts

As discussed in the previous section, there are different types of members within the Contributor Group, namely the domain experts, ontology engineer and team member/team leader. Based on the domain specific issue, domain experts' memberships are determined by the issue. In other words, domain experts are the staff personnel who have the knowledge and expertise in the specific area to which the raised issue pertains. A higher value is assigned to their feedback and votes than other subgroups within the contributor group. The advantage of this arrangement comes in two fold. First, it contributes to the quality of the discussion and, by extension, the evolved ontology as the result. Second, it provides an opportunity for the users to act in different roles; this means that the nature of ownership in this sub-group is dynamic. One staff member may be a domain expert for issue A and a team member for issue B. Members who are domain experts are usually staff located in onshore branches whose role is to provide advice to the offshore staff.

4.6.3.2 Ontology Engineer

The ontology engineer is an expert on ontology but has less domain knowledge within the oil and gas industry than do the domain experts. This role is created to smooth the ontology evolution process. The improved ontology is a result of rigorous discussion performed by members of the Contributor Group where domain expert holds the utmost authority. However, members of this sub-group do not necessarily have knowledge of ontology. This requires an ontology engineer to provide his/her ontology perspective regarding the issue raised by the other members of the platform, including other ontology engineers. This will be vital in cases where the domain expert or other members agree that a change in ontology is required but not feasible within the current ontology language. Hence, the ontology engineer is assigned less expertise value than are the domain experts.

4.6.3.3 Team Leader

The team leader is part of the Contributor Group who has more working experience than other team members. The team leader is usually a supervisor or a manager or a staff member who has a great deal of experience in a particular area. Their votes have higher value than those of team members due to their higher rank in terms of organisation and experience. However, their experience may be irrelevant to the issue raised which in some cases means that they are not an expert in terms of the issue raised. Therefore, their votes have less value than those of the domain experts but are higher than those of team members.

4.6.3.4 Team Member

A team member is a member of the Contributor Group who has less knowledge or expertise on the issue that has been raised. For example, offshore staff who work in drilling operations have less expertise than onshore staff who design the drilling procedures. While the former is responsible for applying the procedures at a practical level, the latter is responsible for the same issue on a more conceptual level. Thus, the offshore staff will be relegated to team members and the onshore staff to domain experts. The members of this sub-group are located in either onshore or offshore branches. Their feedback and discussion is taken into account with less weight than that of the domain experts. Moreover, most of the users in the team member category constitute the biggest number of users classified under the Contributor Group.

4.6.3.5 Administrator

The administrator is a member of the Admin Group whose main role is to manage the access of the users and the group to which they are assigned. The role of Administrator is limited to managerial responsibilities, that is, only managing the users and groups. The administrator does not have the right to interfere with the knowledge domain. Therefore, an administrator will not be able to raise, view or participate in a discussion about any issue raised. The administrator has the right to add and delete the users and groups.

4.6.3.6 System Auditor

The system auditor’s main role is to review the credentials of users to whom the administrator has granted access. This activity is carried out on a half-yearly basis. The idea is to provide continuous inspection and monitor the authenticity of users. This is to ensure that the users still work within the organisation and/or require access to the platform. Although the system auditor is notified about staff employment status, there is a possibility of human error whereby the system auditor is overlooked and not informed when a staff member’s working relationship with the organisation is terminated. System auditors, as with the administrator, will not be able to participate in discussions about any issues raised.

The role of the platform changes in accordance with the different membership within the group. For example, a staff member with expertise in safety procedures is a member of the domain experts group when an issue is raised regarding safety. However, when the issue changes to venture exploration, the particular staff member takes on the role of team member. Therefore, a member of one group is restricted to participate within the roles of the other group.

4.7 Ontology Evolution Platform as a Solution

The Ontology Evolution Platform is developed because none exists at present. In this section, each layer (Social Networking Layer, Ontology Evolution Layer and Permission Layer) of the proposed platform and its functionalities is described in detail. Further, the Ticketing Support System is presented as a means to raise an issue which is used mainly in financial institutions and Web host sectors. Lastly, a combination of Ticketing Support System and community forums that shows the differences and similarities between these two tools is described.

4.7.1 Social Networking Layer

The Social Networking Layer is the first layer where the exchange of knowledge from users take place resulted in a shared one. Each participant can contribute to the development of the ontology as a result of dynamic communication. This layer is accessible only to the Contributor Group. The three main steps are: (i) raise issue; (ii) discuss issue; and (iii) vote on issue. Further explanation is given below.

- **Raise Issue:** All users within the Contributor group are able to raise an issue if they notice there is an error or something needs adjusting within the current ontology or business process. This allows the users to have greater opportunity to raise any issues they notice within the current ontology or business process to make the improvement more participatory. Each issue has a timeframe set by the user who has raised that issue and will be closed after that timeframe is passed. This means the discussion is conducted in an efficient way.
- **Discuss Issue:** Once an user raises an issue, an email is sent to all other users notifying everyone about the issue and timeframe. Users are able to discuss the issue by adding their comments. This includes replying to each other’s comments. These discussions are organised by the user who raised the issue within the timeframe specified. The timeframe is adjustable depending on how the discussion progresses. Everyone within the Contributor Group has the access to make adjustments to the issue raised. In cases where the user who raised the issue is temporarily or permanently unavailable, the platform provides an access to another user to perform this task. This feature facilitates discussion in an orderly fashion with reliable outcomes.

- **Vote on Issue:** Each user will have different weighting based on their expertise in the field. Weightings are divided into three expertise values: (i) domain experts who have the greatest weight due to their expertise regarding the raised issue; (ii) the team leaders who are assigned less weight than the domain experts because they do not have knowledge equal to that of the domain experts; however, they hold more senior managerial positions than do the team members. This managerial status, especially when it is established over a certain length of time, represents knowledge in a wider context. Someone who is at a top management level is inherently able to foresee and place domain-related issues in the right business context, thus bringing additional value to the discussion; (iii) the ontology engineers also have greater weight than the team leader but lower than the domain expert. This is a result of their high expertise in ontology and presumably lower managerial rank in the organisation; and (iv) of the two, the team member will have the least weight due to the lack of expertise and knowledge. While the discussion is in progress, users are able to vote on the issue. The vote will be automatically calculated according to the different weight allotted to different groups in the Contributor Group. There are three options on the vote; each represents a recommendation of what action should be taken with regards to the ontology, i.e. to modify or to keep the ontology as it is. The final recommendation based on accumulation of weighted votes will be concluded only at the end of the set date. The ontology engineer will act upon this recommendation and make the necessary update to the ontology.

4.7.2 Permission Layer

The second layer, Permission Layer, is where users under the Admin Group provide, update and delete every user's access to the system. The users within the Admin Group do not have access to the features in the Social Networking Layer due to the different purpose of each layer. Activities in the Social Networking Layer are focused on the rationale of ontology evolution, while those in the Permission Layer are concentrated on the authenticity of users. This resulted in segregation of duties between the Admin Group that perform managerial duties and the Contributor Group that is tasked with technical duties. Below are further explanations of each feature within this layer:

- **Grant Access:** This feature allows users to be added to their correct group. The group will be either the Admin group or the Contributor Group. If the user has access to the Admin group, all the features within the Permission Layer will be enabled for the user. If the user has access to the Contributor group, all the access within the Social Networking will be enabled.
- **Update Access:** The administrator will be able to update the user's access to the right group if required. For example, if a user is relocated to a different department, his or her expertise may change and he or she may belong to a different domain. Thus, the user in the Admin Group will use this feature to update the user to the right group or to update the user profile.
- **Delete Access:** Once any user leaves the organisation, the administrator will be able to delete his/her access to the system.

4.7.3 Ontology Evolution Layer

The third layer in the system is the Ontology Evolution Layer. Within this layer, the Contributor Group is able to manage the ontology. This layer allows the users to view the current ontology, create a temporary ontology (with the suggested change) or even archive the ontology once the ontology has been updated to a newer version. These three types (Current Ontology, Temporary Ontology and Archived Ontology) are explained further below.

- **Current Ontology:** is the latest version of the ontology that is used by the entire staff when performing their daily work. This version is considered as the most updated ontology before any attempt at evolution is carried out. In other words, this version is the basis for review and discussion in the Social Networking Layer.
- **Temporary Ontology:** is created as a sample to allow the discussion to proceed further. In a business organisation, knowledge about ontology is limited to the ontology Engineer. The majority of the employees are domain experts in a particular industry. This presents predicaments to the ontology evolution process. Users share their ideas in communicative and formal rather than technical language. In order to support the discussion as well as reach a

consensus, the ontology engineer constructs a sample of ontology based on ongoing discussion. This is used mainly to allow discussions to occur while an issue is open by providing a visualisation of the proposed evolution. This means that it is possible to have more than one temporary ontology for one issue to help users to view different versions of the proposed change within the current ontology and to decide on which change needs to be made.

- **Archived Ontology:** is the previous ontology that is archived once the decision has been made to change it. This old version of ontology is viewable by all users as a reference if required. This may be useful in cases where the new ontology (current ontology) is not working as initially planned. Thus, the users may decide to use the older version rather than the new version of the ontology.

For example, oil and gas industry is a dynamic business sector which information and data are in need of constant and timely update. This applies to, for example, the safety procedures of the fire fighting system where the ontology may change once users raise and discuss issues. Once it is decided that a new procedure needs to be implemented, the new ontology will be available on Web-Protégé for the users to view, and the old version will be marked as archived ontology.

4.7.4 Ticketing Support System

A Ticketing Support System is used mainly to manage issues which arise in different areas within an organisation. This means that once the ticket is raised, it is assigned to the right group that will be responsible for resolving the issues that have been raised. This has been used in many industries and in large financial institutions that have multi-site locations. This has helped to centralise many departments and solve many issues thanks to the current advancement of technology.

Other types of businesses that use a ticketing support system are IT support departments. This applies particularly to Web host organisations because virtual support is vital to them. This comes as a result of their clients' locations that are sporadically dispersed in many locations. One of the known Australian Web host organisation that uses a ticketing support system is Digital Pacific (Pacific n.d)

Digital Pacific provides their users with online support via a ticketing system. Users are able to raise, view, edit, update and reply to the ticket that they have initiated.

They are able to choose which category the issue is related to (e.g. Billing, VPS Support). Moreover, the ticketing system is more complex in other industries such as financial institutions, where it needs to be assigned to a specific department.

In the case of oil and gas industry, staff are working in distributed environments where working virtually is a vital part of their daily business activities. A Ticketing Support System is an excellent solution which allows the users to raise issues. Therefore, a ticketing system is a crucial part of many industries and ensures that issues are solved in a timely and efficient manner. However, most organisations use the ticketing system to resolve issues but do not allow the staff to participate in developing the organisation's business processes. This can be done by combining some of the features of the ticketing support system with community forums, which is discussed in the following section.

4.7.5 Combination of Community Forums and Ticket Support Systems

In Chapter 2, the community forum and ticket support systems have already been outlined. Therefore, this section will compare the differences and similarities between the appointed systems.

	Differences	Similarities
Community Forums	1. Different Topics 2. Can be used for raising a query, information or an issue but not solving a problem. 3. The replies to the post can be from any user within the community or, in some cases, from anyone who visits the Website or system.	1. Requires Login authentication. 2. Describes an issue 3. Have some type of vote feature. 4. Both are a type of social network.
Ticket Support Systems	1. Focused Topics 2. Used mainly to ask for support or raise an issue to solve a problem. 3. Usually a reply to the issue is given by an expert in that field.	

Table 4-2: Differences and Similarities between Community Forums and Ticket Support Systems

As shown in Table 4.2, there are no significant differences between the community forum and ticket support systems. However, these minor differences can lead to look at the topic raised in a different way. Therefore, this research considers both of their characteristics which have created a new system to incorporate both of their features into one platform. This includes allowing the users to discuss any topic. Furthermore, the platform allows all staff to participate within the platform. However, each user has a different weighting when participating, which gives an opportunity for everyone to participate but allocates a higher value to users who are experts on the issues raised or on the domain specified.

4.8 Conclusion

In this chapter, overview, strategy and conceptual framework of the Lightweight Community Driven Approach are discussed in detail. Both advantages and disadvantages of the approach are also touched upon. Finally, the lightweight community support and ontology evolution platform as research solutions are explored. In the following chapters, the two other key principles of the Lightweight Community-driven Approach will be discussed in detail. Chapter 5 will provide details on the second key principle which is the implementation and Chapter 6 will focus on the last key principle which is the evaluation of the approach. Finally, Chapter 7 will conclude this research and present the future work needed within the related field.

4.9 References

Braun, S., A. Schmidt, et al. (2007). Ontology Maturing: a Collaborative Web 2.0 Approach to Ontology Engineering. 16th International World Wide Web Conference (WWW 07). Banff, Canada.

Galliers, R. D. (1991). Choosing appropriate information systems research approaches: A revised taxonomy. Information Systems Research: Contemporary Approaches and Emergent Traditions, North-Holland, Amsterdam.

Gendarmi, D. and F. Lanubile (2006). Community-Driven Ontology Evolution Based on Floksonomies. OTM Workshop, Montpellier, France.

Goldenberg, S., (2010). BP oil spill blamed on management and communication failures. Retreived 15 May 2012 from http://www.guardian.co.uk/business/2010/dec/02/bp-oil-spill-failures.

Klein, M. (2004). Change management for distributed ontologies. Amsterdam, Vrije Universiteit. **PhD Thesis**.

Noy, N. F., A. Chugh, et al. (2006). A Framework for Ontology Evolution in Collaborative Environments. 5th International Semantic Web Conference 2006, Athens, USA.

Pacific, D. (n.d). "Submit a Ticket." Retrieved 9 November 2010, from http://www.digitalpacific.com.au/esupport/index.php?_m=tickets&_a=submit.

Siorpaes, K. (2007). Lightweight Community-Driven ontology Evolution. 6th International and 2nd Asian Semantic Web Conference, Busan, Korea.

Stojanovic, L. (2004). Methods and Tools for Ontology Evolution. Karslruhe, FZI - Research Center for Information Technologies. **PhD Thesis**.

Thorsen, K. and C. Rong (2008). Data Integration in Oil and Gas at Norwegian Continental Shelf. 22nd International on Advanced Information Networking and Applications - Workshop, Okinawa, Japan.

Tudorache, T., J. Vendetti, et al. (2008). Web-Protégé: A Lightweight OWL Ontology Editor for the Web. OWL: Experiences and Direction (OWLED) Workshop, Karslruhe, Germany.

Vrandecic, D., H. S. Pinto, et al. (2005). "The diligent knowledge processes." Journal of Knowledge Management **9**: 85 - 96.

Wang, W., R. Xiong, et al. (2007). "Design of a Web2.0-based Knowledge Management Platform." IFIP International Fedration for Information Processing**2**(2007): 237-245.

Every reasonable effort has been made to acknowledge the owners of copyright material. I would be pleased to hear form any copyright owner who has omitted or incorrectly acknowledged.

Chapter 5 – Implementation for Lightweight Community Driven Approach

5.1 Introduction

In Chapters 3 and 4, two key issues and solutions for these issues of lightweight community support and ontology evolution platform are discussed. The solutions require a common platform that allows vigorous discussions to take place. In the previous chapter the conceptual framework of the platform was presented. This includes how the platform allows the staff to participate in the ontology evolution of any specific domain after an informal discussion between different staff. This is also introduced by a case study which is related to the oil and gas domain.

In this chapter, the implementation of the approach is explained in great detail. The pseudocode of the platform for lightweight community support is presented, including all the functions of the platform. Also, additional details about each function are given. The chapter continues with further explanation of how the calculation of the voting is done in the platform.

5.2 Pseudocode for Lightweight Community Support

This section displays the pseudocodes which describe how each function of the platform is built. This is to show how the system has been created and describe each function that is developed to ensure the involvement of users in ontology evolution process.

5.2.1 Login

This function validates user login details then checks whether or not the input username exists in the database. If the user is found, then it checks whether the user has a password the same as the input password. Finally, if the passwords match, then the user is redirected to the user control panel. The control panel displays the access privileges of the users. The privileges depend on the group to which the user is assigned. These groups include: (i) Administrator, (ii) Domain Expert, (iii) Team Member, (iv)Team Leader, (v) System Auditor, or (vi) Ontology Engineer.

```
FUNCTIONlogin with username and password
    BEGIN
        /* flag is set to FALSE to say the system that no user is logged in yet */
        SET flag to FALSE
        /* the for loop iterates for every user in the database */
        FOR each user in user list
            /* finds user in the database */
            IF user [name]EQUALS TO username
                /* if user is found the matches the password */
                IF user [password]EQUALS TO user password
                    /* if password matches get user permission details from database */
                    SET privilege to user [privilege]
                /* checks privilege and assigns the permissions to the user accordingly */
                    CASE privilege
                        /* if user is administrator assign administrator privilege */
                        Administrator   : SETGLOBAL permission of admin for user

                        /* if user is Domain Expert assign Domain Expert privilege */
                        Domain Expert   : SETGLOBAL permission of domain expert for user

                        /* if user is Team Member assign Team Member privilege */
                        Team Member     : SETGLOBAL permission of team member for user

                        /* if user is Team Leader assign Team Leader privilege */
                        Team Leader     : SETGLOBAL permission of team leader for user

                        /* if user is System Auditor assign System Auditor privilege */
                        System Auditor : SETGLOBAL permission of system auditor for user
                    /* if user is Ontology Engineer assign Ontology Engineer privilege */
                        Ontology Engineer      : SETGLOBAL permission of ontologist user
                    ENDCASE

                    /* Since user is found assign flag to TRUE */
                    SET flagEQUAL TO TRUE
                    Break FOR loop // Breaks the for loop here from
            ENDIF
        ENDFOR
        /* Checks whether the user has been found in the database or not */
        IF flagIS EQUAL TO TRUE
            /* if user is found redirects user to user control panel */
            CALL control panel
        ELSE
            /* otherwise prints the error message for incorrect value */
            PRINT Your username or password is incorrect.
        ENDIF
    END
ENDFUNCTION
```

Figure 5-1 : Login Pseudocode

5.2.2 Control Panel

The Control Panel function creates the user interface and provides the options for user Action on the basis of the user permission. The options are based on whether the user is within the Administrator or Contributor Group. The Administrator Group will be able to manage the users and group which has an option menu of Home, Projects, Groups, Users, Feedback and Logout. The Contributor Group will be able mainly to manage projects and issues which has an option menu of Home, Project, Feedback and Logout.

```
FUNCTIONcontrol panel
    BEGIN
        SET permissionIS EQUAL TOGLOBAL permission
        CREATE user interface according to the permission
        IF permission ISEQUAL TO Administrator
            /* if user is administrator these action’s facility */
            CASE menu options

                /* redirects user to home interface */
                Home        : CALL home

                /* redirects user to projects interface */
                Projects    : CALL projects

                /* redirects user to groups interface */
                Groups      : CALL groups

                /* redirects user to users interface */
                Users       : CALL users

                /* redirects user to feedback interface */
                Feedback    : CALL feedback

                /* logs out the user and ends its session */
                Logout      : CALL logout
            ENDCASE
        ELSE
            /* if user is not administrator these action’s facility */
            CASE menu options

                /* redirects user to home interface */
                Home        : CALL home

                /* redirects user to projects interface */
                Projects    : CALL projects

                /* redirects user to feedback interface */
                Feedback    : CALL feedback

                /* logs out the user and ends its session */
                Logout      : CALL logout
            ENDCASE
        ENDIF
    END
ENDFUNCTION
```

Figure 5-2: Control Panel Pseudocode

5.2.3 Project

The Project function creates projects interface for the current user and provides options for user action as per the user permission. If the user is in the Administrator Group, viewing projects is permitted, but not adding or editing. If the user is in the Contributor group, the user will be able to add a project. The page lists all the projects have been created and also provides in the action column the actions that the users are able to perform, i.e. Add Project, View Project, View Issues.

```
FUNCTIONprojects
    BEGIN
    SET Array projects [][]to user projects initialized multi-dimensional array
        /* iterates for every project in the database */
        FOR each projects as project

            /* Prints Name of the Project */
            PRINT Project Name     -project [name]

            /* Prints Start date of the Project*/
            PRINT Start Date       -project [start date]

            /* Prints End date of the project*/
            PRINT End Date         -project [end date]

            /* Prints Status of the project whether active or not */
            PRINT Active           -project [active]
            IF user is administrator
                CASE project Actions
                    /* redirect user to view project interface */
                    View Project    : CALL view project with project [id]

                    /* redirect user to view issues interface */
                    View Issues     : CALL view project issues with project [id]
                ENDCASE
            ELSE
                CASE projectActions

                    /* redirect user to add project interface */
                    Add Project     : CALL add project

                    /* redirect user to view project interface */
                    View Project    : CALL view project with project [id]

                    /* redirect user to edit project interface */
                    Edit Project    : CALL edit project with project [id]

                    /* redirect user to delete project interface */
                    Delete Project : delete project[id] from database

                    /* redirects projects interface recursively */
                                   CALL projects

                    /* redirect user to view issues interface */
                    View Issues     : CALL view project issues with project [id]
                ENDCASE
            ENDIF
        ENDFOR
    END
ENDFUNCTION
```

Figure 5-3: Project Pseudocode

5.2.4 Issue

The Issue function creates an issues interface for the user and provides options for user action according to the user access. This function allows the user to either view the issue and the vote result or manage the issue by adding, editing, deleting, or viewing the issue as well as voting on the actual issue raised and view the vote result.

```
FUNCTIONviewproject issues with project id
    BEGIN

        /* Initializes a multi-dimensional array and set its value to the Project Issues
        details in the database */
        SET Array issues [][]to project issues

        /* iterates for every issue in the issues array */
        FOR each issues as issue

            /* Prints name of the issue */
            PRINT Issue Name      -issue [name]

            /* Prints End date of the issue */
            PRINT Issue End Date  -issue [end date]

            /* Prints Webprotege for the issue */
            PRINT Webprotege      - issue [Webprotege]

            IF user is administrator
                CASE issue Actions

                    /* redirects to view issue with issue array as argument */
                    View Issue    : CALL view issue with issue

                    /* redirects to vote result with issue array as argument */
                    Vote Result   : CALL vote result with issue
                ENDCASE

            ELSE

                CASE issue Actions
                    /* redirects to add issue interface with project id as argument */
                    Add Issue     : CALL add issue with project id

                    /* redirects to view issue with issue array and project id as argument */
                    View Issue    : CALL view issue with issue

                    /* redirects to edit issue with issue array and project id as argument */
                    Edit Issue    : CALL edit issue with issue and project id

                    /* deletes the issue details from database and redirects to the view issue
                    Recursively with project id as argument */
                    Delete Issue  : DELETE issue [id] from the database
                                  CALL view project issues with project id

                    /* redirects user to voting interface with issue array and project id
                    as argument*/
                    Vote          : CALL vote with issue and project id

                    /* redirects to vote result with issue array as argument */
                    Vote Result   : CALL vote result with issue
                ENDCASE
            ENDIF
        ENDFOR
    END
ENDFUNCTION
```

Figure 5-4: Issue Pseudocode

5.2.5 View Issue

If the user clicks on View Issue, this function takes the issue array as input and creates a view issue interface in order for the user to be able to see the details of the actual issue raised. The fields that the user is able to view (Issue Name, Issue Subject, Issue Description, Web-Protégé, Ontology Element, Start Date, End Date and Status). The Web-Protégé and Ontology Element Field are developed especially in the issue page in order to be able to link the voting system with the Ontology which is viewed in the Web-Protégé. The Web-Protégé field is used to define the name of the element which is targeted for discussion in the issue raised. Ontology Element is used to describe what that element is related to.

```
FUNCTIONviewissue with issue[]
    BEGIN

        /* Prints name of the issue */
        PRINT Issue Name           -issue [name]

        /* Prints subject of the issue */
        PRINT Issue Subject        -issue [subject]

        /* Prints name of the issue */
        PRINT Issue Description    -issue [description]

        /* Prints name of the issue */
        PRINT Webprotege           -issue [Webprotege]

        /* Prints Ontology element of the issue */
        PRINT Ontology Element     -issue [ontology element]

        /* Prints start date of the issue */
        PRINT Start Date           -issue [start date]

        /* Prints end date of the issue */
        PRINT End Date             -issue [end date]

        /* Prints status of the issue whether Open or not */
        PRINT Status               -issue [status]

        IF user IS NOT administrator
            /* If user is not administrator then provide Edit Issue options */
            CASE option

            /* redirects to edit issue interface with issue array as argument */
                Edit    : CALL edit issue with issue
            ENDCASE
    END
ENDFUNCTION
```

Figure 5-5: View Issue Pseudocode

5.2.6 Edit Issue

The Edit Issue function takes issue array as argument then creates user input form and populates the issue array information to the form as user submits the details with changed values then the details are saved into the database.

```
FUNCTIONeditissue with issue [] and project id
    BEGIN

        CREATE user input form fields
        POPULATE values from issue to the input fields

        /* Reads issue name */
        READ name

        /* Reads issue subject */
        READ subject

        /* Reads issue description*/
        READ description

        /* Reads issue element */
        READ ontology element

        /* Reads issue topic */
        READ topic

        /* Reads issue Webprotege */
        READ Webprotege

        /* Reads issue start date for the issue*/
        READ start date

        /* Reads issue end date for the issue*/
        READ end date

        /* Reads issue status of the issue (open or not) */
        READ status

        /* Assigns currently logged in user id to a variable */
        SET userid EQUAL TO current user id

        /* Initiates voting value for Unresolved issue to 0(Zero) */
        SET Unresolved issue EQUAL TO 0

        /* Initiates voting value for Viewpoint issue to 0(Zero) */
        SET View point issue EQUAL TO 0

        /* Initiates voting value for Modify issue to 0(Zero) */
        SET Modify issue EQUAL TO 0

        VALIDATE all the fields for valid Inputs
        UPDATE issue corresponding to issue [id] into database
        CALL view project issues with project id
    END
ENDFUNCTION
```

Figure 5-6: Edit Issue Pseudocode

5.2.7 Add Issue

The Add Issue function takes the project id as argument and creates a user input form for accepting user input. When the user invokes “submit”, the form details are saved into the database.

```
FUNCTION add issue with project id
    BEGIN
        CREATE user input form fields

        /* Reads issue name */
        READ name

        /* Reads issue subject */
        READ subject

        /* Reads issue description*/
        READ description

        /* Reads issue element */
        READ ontology element

        /* Reads issue topic */
        READ topic

        /* Reads issue Webprotege */
        READ Webprotege

        /* Reads issue start date for the issue*/
        READ start date

        /* Reads issue end date for the issue*/
        READ end date

        /* Reads issue status of the issue (open or not) */
        READ status

        /* Assigns currently logged in user id to a variable */
        SET user id EQUAL TO current user id

        /* Initiates voting value for Unresolved issue to 0(Zero) */
        SET Unresolved issue EQUAL TO 0

        /* Initiates voting value for Viewpoint issue to 0(Zero) */
        SET View point issue EQUAL TO 0

        /* Initiates voting value for Modify issue to 0(Zero) */
        SET Modify issue EQUAL TO 0


        CREATE new issue and SAVE project id, name, subject, description,
        ontology element, topic, Webprotege, start date, end date, status, user
id,
        Unresolved issue, Viewpoint issue, Modified issue into the database
        CALL view project issues with project id
    END
ENDFUNCTION
```

Figure 5-7: Add Issue Pseudocode

5.2.8 Vote Options

A vote function takes the argument of issue as array which contains all the issue-related properties. It checks proper conditions then shows the user's voting options. The user may choose from three options; (i) Unresolved Issue, (ii) Viewpoint Issue, or (iii) Modify Issue. Unresolved Issue occurs when the users do not find any solutions to the raised issue. Viewpoint Issue means that the ontology should stay the same. Finally, Modify Issue, means that an action such as revision of the ontology is demanded. Once the user votes on an option, his/her vote is saved in the database.

```
FUNCTIONvote with issue [] and project id
    BEGIN

        /* checks issues expiration */
        IF issue [end date] is LESS THAN EQUAL TO current date

            /* check whether the user has ever voted this issue or not */
            IF user has voted for issue [id]

                /* if user has voted for this issue previously show this message */
                PRINT you have already voted for this issue.
            ELSE

                /* create voting options for user vote */
                CASE vote options
                    Unresolved issue  :PRINT Unresolved issue
                                      IF selected

                                          /* assign current value of issue to variable */
                                              SET value EQUALS TO issue [Unresolved issue]
                                              Add user expertise value to value

                                              /*assigns new value to issue */
                                              SET issue [Unresolved issue]EQUAL TO value
                                      ENDIF

                    Viewpoint issue   : PRINT Viewpoint issue
                                      IF selected

                                              /* assign current value of issue to variable
*/
                                              SET value EQUALS TO issue [Viewpoint issue]
                                              Add user expertise value to value

                                              /*assigns new value to issue */
                                              SET issue [Viewpoint issue]EQUALTOvalue
                                      ENDIF
                    Modify issue      :PRINT Modify issue
                                      IF selected

                                          /* assign current value of issue to variable */
                                              SET value EQUALS TO issue [Modify issue]
                                              Add user expertise value to value

                                              /*assigns new value to issue */
                                              SET issue [Modify issue]EQUAL TO value
                                      ENDIF
                ENDCASE
                IF user votes

                    /* save new values of issue to the database */
                    SAVE issue
                    CALL view project issues with project id
                ENDIF
            ENDIF
        ELSE
            /* if issue date has expired show this message */
            PRINT issue has ended.
        ENDIF
    END
ENDFUNCTION
```

Figure 5-8: Vote Options Pseudocode

5.2.9 Reputation Value Calculation

After all users have cast their votes on each option, i.e. Unresolved Issue, Viewpoint Issue and Modify Issue, these are calculated by taking into account the reputation value of each user. As explained in the previous chapter, the reputation value captures users' past performance in discussions on issues regarding the ontology evolution process. In addition, a different weight is assigned to each of the votes according to each voter's expertise on the discussed issue. This calculation produces the expertise value. Both calculations are described in the next two sections.

Firstly, the current reputation value of the user is calculated and stored in the database. In order to enhance and provide a practical solution, voting on each issue needs to be calculated differently as each user has a different background and experience in terms of the issue raised. Therefore, this platform will calculate the total value of each vote option selected by the user. Further explanation is given below.

As briefly outlined above, there are two values which this platform takes into account to weight the submitted vote: expertise value and reputation value. Regarding expertise value, users with more experience and knowledge are given a higher value. This starts with the highest expertise value given to the domain expert followed by ontology engineer, team leader and team member. The domain expert has an expertise value of 0.8, ontology engineer 0.7, team leader 0.5 and team member 0.2. On the other hand, the reputation value is calculated as either 1 or 2 and is based on past reputation points the users have accumulated on different projects. The reputation value always starts with the value of 1 which allows the user to increase the value by voting on the right option. This encourages the staff to participate in order to increase their vote weight. Reputation value is identified according to the category of issues, for example drilling, safety, construction, and transport issues. The staff member's reputation value is calculated based on his/her voting performance on a selected issue. Therefore, staff members have different reputation values for different issue. The Markov Model is used to consider the change within the reputation value and the most possible future reputation value (Wongthongtham 2006).

There are three steps in calculating the reputation value. The first step is to calculate the Current State Value (CSV) which is the latest reputation value. The second step is the calculation of the Markov Matrix; and finally, the CSV is multiplied with the Markov Matrix. The CSV has two possibilities. If the latest reputation value is 2 then the matrix is [0 1]. However, if the reputation value is 1, the matrix is [1 0]. For example, the reputation value history of a member is [2,1,2,1,2,2].

Second, the Markov Matrix needs to be calculated. In order to do this, state transition needs to be calculated first. As there are only 1s or 2s within the reputation value, the state transition has four possibilities. The possibilities are 1 – 1 state, 1 – 2 state, 2 – 1 state and 2 – 2 state. As from the example above, there is no 1 -1 state; there are two 2 – 1 state; another two 1 – 2 state and finally one 2 – 2 state as shown below.

$$\text{Transition States Matrix} = \begin{bmatrix} 0 & 2 \\ 2 & 1 \end{bmatrix}$$

Once the frequency of state transition has been counted, the Markov Matrix is calculated. In our case, 1 -1 state has 0/2 which is 0%, 1 – 2 state has 2/2 which is 100%, the 2 – 1 state is 2/3 which is 66.6% and 2 – 2 state is 1/3 which is 33.3% as shown below (Chang, Dillon et al. 2006).

$$\text{Markov Matrix} = \begin{bmatrix} 0 & 1 \\ 0.66 & 0.33 \end{bmatrix}$$

Finally, by multiplying the Markov Matrix with the CSV, a reputation value is counted. This is based on the result. If the value a is greater than b, it means the reputation value is 1 and if the value b is greater than a, then the reputation value will be 2 as shown below (Wongthongtham 2006).

$$\text{Reputation Value Probability} = \begin{bmatrix} 0 & 1 \\ 0.66 & 0.33 \end{bmatrix} \text{ x } [0 \quad 1] = [0.66 \quad 0.33]$$

The points for each vote as shown in Figure 5.11 above are calculated as a summation of every voter’s expertise and reputation value. A detail explanation of the calculation is provided in the next section.

```
FUNCTION calculate_reputation_value

    /* id of last issue user has voted which has been closed now */
    SET last_vote_issue_id to id of last issue user has voted and has been closed

    /* result of the last issue user has voted which has been closed now */
    SET last_result_issue_id to result of last issue user has voted and has been
closed

    /* Get values of user's reputation history from database */
    SET Array reputation_history[] to users reputation history

    /* get current reputation value of database */
    SET crv to user's current reputation value

    /* gets number of elements in reputation history array */
    SET history_length EQUAL TO CALL count with reputation_history

    /* Declare array for current state value matrix */
    DECLARE Array csv_mat[2]

    /* creates contents in current state value matrix */
    IF reputation_history [history_length MINUS 1] IS EQUAL TO 1
        SET csv_mat[0] EQUAL TO 1
        SET csv_mat[1] EQUAL TO 0
    ELSE
        SET csv_mat[0] EQUAL TO 0
        SET csv_mat[1] EQUAL TO 1
    ENDIF

    /* initiates total number of 1's in reputation history array by 0(Zero) */
    SET ones EQUAL TO 0

    /* initiates total number of 2's in reputation history array by 0(Zero) */
    SET twos EQUAL TO 0

    /* initiates total number of occurrence of 1-1's in reputation history array
by 0(Zero) */
    SET one_one EQUAL TO 0

    /* initiates total number of occurrence of 1-2's in reputation history array
by 0(Zero) */
    SET one_two EQUAL TO 0

    /* initiates total number of occurrence of 2-1's in reputation history array
by 0(Zero) */
    SET two_one EQUAL TO 0

    /* initiates total number of occurrence of 2-2's in reputation history array
by 0(Zero) */
    SET two_two EQUAL TO 0

/* variable to check pairs of 1-1's, 1-2's, 2-1's or 2-2's is initiated with 0 */
    SET last_value EQUAL TO 0
```

Figure 5-9: Calculation Reputation Value Pseudocode

```
/* iterates for each element of reputation history */
    FOR each reputation_history[] as rValue

        /* counts occurrence of 1's in reputation history */
        IF rValue IS EQUAL TO 1
            INCREMENT ones by 1
        ENDIF

        /* counts occurrence of 2's in reputation history */
        IF rValue IS EQUAL TO 2
            INCREMENT twos by 1
        ENDIF

        /* counts occurrence of 1-1's in reputation history */
        IF last_value IS EQUAL TO 1 AND rValue IS EQUAL TO 1
            INCREMENT one_one by 1
        ENDIF

        /* counts occurrence of 1-2's in reputation history */
        IF last_value IS EQUAL TO 1 AND rValue IS EQUAL TO 2
            INCREMENT one_two by 1
        ENDIF

        /* counts occurrence of 2-1's in reputation history */
        IF last_value IS EQUAL TO 2 AND rValue IS EQUAL TO 1
            INCREMENT two_one by 1
        ENDIF

        /* counts occurrence of 2-2's in reputation history */
        IF last_value IS EQUAL TO 2 AND rValue is EQUAL TO 2
            INCREMENT two_two by 1
        ENDIF
        last_value EQUAL TOrValue
    ENDFOR

    /* Declares Markov's 2x2 matrix */
    DECLARE Array markov_mat[2][2]

    /* Sets value at position 11 in Markov's Matrix */
    SET markov_mat[0][0] EQUAL TO one_one DIVIDED BY SUM OF one_one and one_two

    /* Sets value at position 12 in Markov's Matrix */
    SET markov_mat[0][1] EQUAL TO one_two DIVIDED BY SUM OF one_one and one_two

    /* Sets value at position 21 in Markov's Matrix */
    SET markov_mat[1][0] EQUAL TO two_one DIVIDED BY SUM OF two_one and two_two

    /* Sets value at position 22 in Markov's Matrix */
    SET markov_mat[1][1] EQUAL TO two_two DIVIDED BY SUM OF two_one and two_two

/* declares linear array of size 2 for result of Markov's Matrix Calculation */
    DECLARE Array res_mat[2]

    /* sets contents of result matrix according to Markov's Matrix calculation */
    SET res_mat[0] EQUAL TO SUM OF
                              markov_mat[0][0] MULTIPLIED BY csv_mat[0]
                              and
                              markov_mat[1][0] MULTIPLIED BY csv_mat[1]
    SET res_mat[1] EQUAL TO SUM OF
                              markov_mat[0][1] MULTIPLIED BY csv_mat[0]
                              and
                              markov_mat[1][1] MULTIPLIED BY csv_mat[1]
```

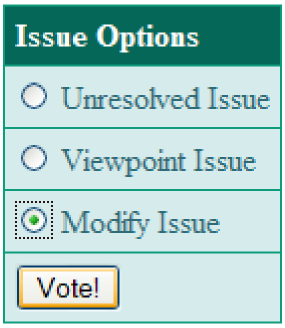
Figure 5-9: Calculation Reputation Value Pseudocode

```
    /* if users last vote was correct */
    IF last_vote_issue_id IS EQUAL TO last_result_issue_id
        /* checks reputation history content's length */
        IF history_lengthIS LESS THAN 4
            SET crvEQUAL TO 2
            reputation_history[history_lengthMINUS 1] EQUAL TOcrv
        ELSE
            /* if count of either of number of 1's or 2's is less than 2 */
            IF onesIS LESS THAN 2 ORtwosIS LESS THAN 2
                SETcrv EQUAL TO 2

           ELSE /* if count of both 1's and 2's are greater than 2 or equal to 2*/

            /* if first value in resultant matrix is greater than its 2nd Value */
                IF res_mat [0] IS GREATER THANres_mat [1]
                    SET crvEQUAL TO 1
                ELSE
                    SET crvEQUAL TO 2
                ENDIF
            ENDIF
        ENDIF
    ELSE
        /* checks reputation history content's length */
        IF history_lengthIS LESS THAN 4
            SET crvEQUAL TO 1
            reputation_history[history_lengthMINUS 1] EQUAL TOcrv
        ELSE
            /* if count of either of 1's or 2's is less than 2 */
            IF onesIS LESS THAN 2 ORtwosIS LESS THAN 2
                SET crvEQUAL TO 1

            ELSE/* if count of both 1's and 2's are greater than 2 or equal to 2 */

            /* if first value in resultant matrix is greater than its 2nd Value */
                IF res_mat [0] IS GREATER THANres_mat [1]
                    SET crvEQUAL TO 1
                ELSE
                    SET crvEQUAL TO 2
                ENDIF
            ENDIF
        ENDIF
    ENDIF

    /* store the contents of reputation history array and current
    reputation value into database */
    STORE reputation_history and crv into database
ENDFUNCTION
```

Figure 5.9: Calculation Reputation Value Pseudocode

5.2.10 Voting Points Calculation

When the discussion time expires, as shown by the “End Date”, each member of the Contributor Group casts his/her vote. Figure 5.10 shows the three options. Thus, in order to provide a comprehensive view of this, a simulation of the platform is presented.



Issue Options

○ Unresolved Issue

○ Viewpoint Issue

● Modify Issue

Figure 5-10: Vote Options

Assuming that Reputation Value has been calculated and stored, the next step is to consider a composition of the Contributor Group as follows: 2 staff members are domain experts, 4 staff are team leaders, 10 staff are team members and 2 staff are ontology engineers. At the end of a discussion, each member votes on the issue as presented below:

	Unresolved Issue	**Viewpoint Issue**	**Modify Issue**	**Total**
Domain Expert			2	2
Ontology Engineer	1		1	2
Team Leader	1	2	1	4
Team Member	4	1	5	10
Total	6	3	9	18

Table 5-1: Number of Votes

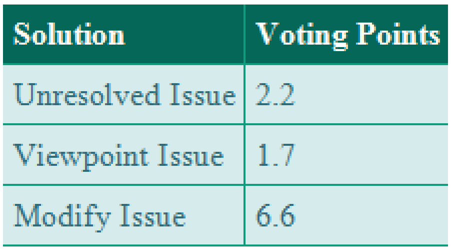
Table 5.1 shows that most of the votes placed the issue as a Modify Issue. The Lightweight Community-driven Platform acknowledges the different values of votes, which are based on expertise and reputation values. Each value reflects experience and knowledge. The expertise value captures present intellectual competence and experience; thus, values are assigned in descending order for domain experts (0.8), ontology engineer (0.7), team leader (0.5), and team member (0.2). These values are predefined for simplification purposes but it is the firms' discretion to decide them in real-world application. Since domain experts have the relevant knowledge and the most reliable feedback, they are assigned the highest value. The ontology engineers are ranked second as they have a better knowledge of the ontology that allows them to alter the course of discussions. The team leaders are allocated 0.5 since their working experience will add value to the discussion. Lastly, the team members have the lowest expertise value since they have the least knowledge on that specific issue.

Table 5.2 presents an example of how the values are calculated for an issue. In this example, the assumption is made that the discussion closed after the users voted on the issue.

Solution	**Domain Expert**	**Ontology Engineer**	**Team Leader**	**Team Member**	**Voting Points**
Unresolved Issue		0.7 x 1	0.5 x 1	0.2 x 1	2.2
				0.2 x 2	
				0.2 x 1	
				0.2 x 2	
Viewpoint Issue			0.5 x 2	0.2 x 1	1.7
			0.5 x 1		
Modify Issue	0.8 x2	0.7 x 2	0.5 x 2	0.2 x 2	6.6
	0.8 x 1			0.2 x 2	
				0.2 x 1	
				0.2 x 2	
				0.2 x 2	

Table 5-2: Total Voting Points

Most of the users agreed that this is a Modify Issue with a total of 6.6 voting points. However, some of the users disagreed and voted it as an Unresolved Issue with a total of 2.2 points; there is the Viewpoint of 1.7. In this scenario, it is clear that most of the staff agrees that the raised issue is a Modify Issue and a change to the ontology is required for that issue. The result of the voting is presented in Figure 5.11, showing Modify Issue has the highest voting points of 6.6.



Solution	Voting Points
Unresolved Issue	2.2
Viewpoint Issue	1.7
Modify Issue	6.6

Figure 5-11: Vote Results

The challenge of a domain-based discussion is to ensure the reliability of results. The demonstration of the voting feature shows that this has been overcome by the different weights assigned to the voting done by different sub-groups. This avoids the validity issue that quantity prevails over the discussion outcome. Taking into account the number of team members in the Contributor Group, this potentially affects the voting count but not the voting points. Domain experts, ontology engineers, team leaders and team members have expertise values ranging from the highest to lowest. Each value represents the knowledge and expertise of each member. Each user accumulates a different reputation value which represents his/her trust level in terms of the suggestions they make regarding the raised issue. Of the two values, expertise has more significance which means the platform holds a sound line of reasoning in high esteem. The logical consequence of such outlook is an improved ontology of an issue, although the platform also provides options for the ontology to remain unchanged.

```
FUNCTION calculate voting points [] and issue id

        BEGIN

        /* assigns all the issue options in the database to issue option array*/

                SET Array issue options [] to all issue options

                FOR each issue options as issue options

        /* assigns all the result users in the database to users array*/

                SET Array users [][] to all users

                        FOR each users as users

                        /* set users expertise value*/

                        SET users[issue options][link expertise weight]

                        /* set users reputation value*/

                        SET users[issue options][reputaiton value]

                        /* set user vote points to variable */

                        SET vote points        users[issue options][link expertise
weight] * users[issue options][reputation value]

                        /*assign vote points to vote points variable*/

                        vote points += vote points

                        ASSIGN vote points to vote points

                        ENDFOR

                        /*print total vote points*/

                        PRINT total vote points - vote points variable

        `       ENDFOR

        END

ENDFUNCTION
```

Figure 5-12: Voting Point Calculation Pseudocode

5.2.11 Voter Results

This function takes the issue array as argument and prints voting options along with total values of votes to date. It shows the total score for each option.

```
FUNCTION vote result with issue []
    BEGIN

        /* Prints current vote for Unresolved Issue */
        PRINT Unresolved Issue     - issue[Unresolved issue]

        /* Prints current vote for Viewpoint Issue */
        PRINT Viewpoint Issue      - issue[Viewpoint issue]

        /* Prints current vote for Modify Issue */
        PRINT Modify Issue         - issue[Modified issue]
    END
ENDFUNCTION
```

Figure 5-13: Vote Results Pseudocode

5.2.12 Users

This function takes the user array as argument and prints user details. This function is accessible to the administrator who is able to perform four main actions: (i) Add User, (ii) Edit User, (iii) View User, and (iv) Delete User.

```
FUNCTION users
    BEGIN
        /* assigns all the users in the database to users array*/
        SET Array users [][] to all users

        /* iterates for each user in the database*/
        FOR each users as user

            /* Prints name of the user*/
            PRINT Name     - user [given name]

            /* Prints email address of the user*/
            PRINT Email    - user [email]

            /* Prints status of the user, whether active or not*/
            PRINT Active   - user [active]
            CASE user Actions

                /* redirects user to view user interface with user array*/
                View    :CALL view user with user

                /* redirects user to edit user interface with user array*/
                Edit    :CALL edit user with user

                /* deletes user information from the database and redirects to
                users interface recursively*/
                Delete  : delete user information from database
                        CALL users
            ENDCASE
        ENDFOR
    END
ENDFUNCTION
```

Figure 5-14: Users Pseudocode

5.2.13 View User

A view user function displays the details of the user by calling the user array. The administrator is able, after viewing the user details, to call the edit user function to update the actual user details.

```
FUNCTION view user with user []
    BEGIN
        FOR each users as user

            /* Prints Given Name of the user */
            PRINT GivenName       - user [given name]

            /* Prints surname of the user */
            PRINT Surname         - user [surname]

            /* Prints username of the user */
            PRINT Username        - user [username]

            /* Prints password of the user */
            PRINT Password        - user [password]

            /* Prints email address of the user */
            PRINT Email           - user [email]

            /* Prints Phone number of the user */
            PRINT Phone           - user [phone]

            /* Prints Group of the user in which user belongs to */
            PRINT Group           - user [group]

            /* Prints Category of the user */
            PRINT Category        - user [expertise value]


            /* Prints Expertise value of the user */
            PRINT Expertise Value - user [expertise value]

            /* Prints Reputation value of the user */
            PRINT Reputation Value- user [reputation value]

            /* Prints Status of the user, whether active or not */
            PRINT Active          - user [active]
            CASE user Actions

                /* redirects user to edit user interface with user array*/
                Edit    :CALL edit user with user
            ENDCASE
        ENDFOR

    END
ENDFUNCTION
```

Figure 5-15: View User Pseudocode

5.2.14 Edit User

This function creates an interface to edit user information and, after completing the process, the user is redirected to the user’s interface. The administrator is able to update the user details and save the updated information in the database.

```
FUNCTION edit user with user
    BEGIN
        CREATE user input form for user fields
        Populate current values of the user into form fields

        /* set changed name of users given name*/
        SET user [given name] to user input value of given name

        /* set changed surname of users surname*/
        SET user [surname] to user input value of surname

        /* set changed username of users username*/
        SET user [username] to user input value of username

        /* set changed password of users password*/
        SET user [password] to user input value of the password

        /* set changed email of users email*/
        SET user [email] to user input value of the email

        /* set changed phone of users phone*/
        SET user [phone] to user input value of the phone

        /* set changed group of users group*/
        SET user [group] to user selected group name

        /* set changed category of users category*/
        SET user [category] to user selected category

        /* set changed status of users*/
        SET user [active] to user set status


        CASE user actions
        /* As user invokes for submit all the fields are validated then new values
            are saved into the database and user is redirected to users interface*/
            Submit: VALIDATE all the user fields
                    Save new values to database
                    CALL users
        ENDCASE
    END
ENDFUNCTION
```

Figure 5-16: Edit User Pseudocode

5.2.15 Add User

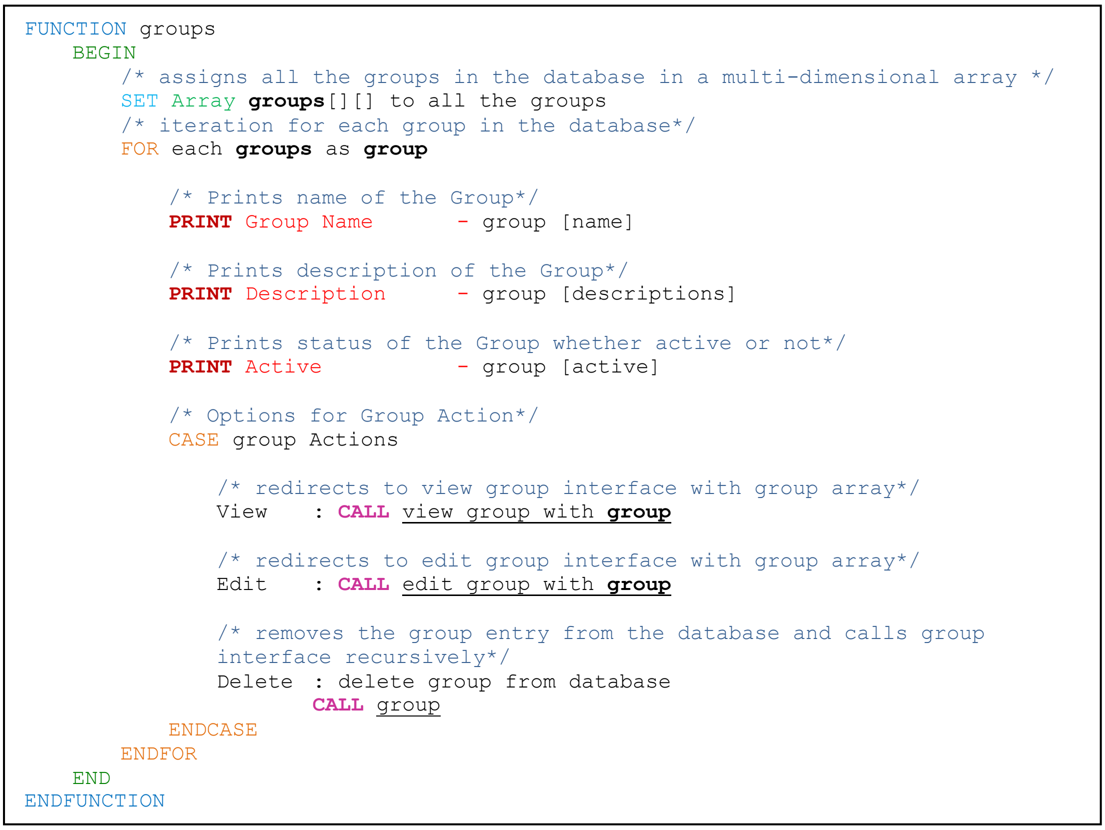
This function creates an interface to add a new user and after completing the process, the user is redirected to the user's interface. The administrator is able to create a new user and save the details in the database.

```
FUNCTION add user
    BEGIN

        /* create a linear array to store temporary user information*/
        SET Array user[]
        CREATE user input form for user fields

        /* sets given name of the user*/
        SET user [given name] to user input value of given name

        /* sets surname of the user*/
        SET user [surname] to user input value of surname

        /* sets username of the user*/
        SET user [username] to user input value of username

        /* sets password of the user*/
        SET user [password] to user input value of the password

        /* sets email address of the user*/
        SET user [email] to user input value of the email

        /* sets phone number of the user*/
        SET user [phone] to user input value of the phone

        /* sets group for the user*/
        SET user [group] to user selected group name

        /* sets category for the user*/
        SET user [category] to user selected category

        /* sets status of users*/
        SET user [active] to user set status


        CASE user actions
        /* As user invokes for submit, all the fields are validated then values
        are saved into the database with new user account and user is redirected
        to users interface*/
            Submit: VALIDATE all the fields
                    Create new user with these values into the database
                    CALL users
        ENDCASE
    END
ENDFUNCTION
```

Figure 5-17: Add User Pseudocode

5.2.16 Groups

This function creates the group interface and lists all the groups which have been created in the platform. The groups which this platform has pre-sent are the administrator, domain expert, team member, team leader, system auditor and ontology engineer. The administrator is able to see that the actual groups exist as well as add, edit or view the groups.



```
FUNCTION groups
    BEGIN
        /* assigns all the groups in the database in a multi-dimensional array */
        SET Array groups[][] to all the groups
        /* iteration for each group in the database*/
        FOR each groups as group

            /* Prints name of the Group*/
            PRINT Group Name      - group [name]

            /* Prints description of the Group*/
            PRINT Description     - group [descriptions]

            /* Prints status of the Group whether active or not*/
            PRINT Active          - group [active]

            /* Options for Group Action*/
            CASE group Actions

                /* redirects to view group interface with group array*/
                View    : CALL view group with group

                /* redirects to edit group interface with group array*/
                Edit    : CALL edit group with group

                /* removes the group entry from the database and calls group
                interface recursively*/
                Delete  : delete group from database
                        CALL group
            ENDCASE
        ENDFOR
    END
ENDFUNCTION
```

Figure 5-18: Groups Pseudocode

5.2.17 View Group

The View Group function calls the group array and displays the group details. The administrator is able to call the edit group function to update the field details of the group.

```
FUNCTION view group with group []
    BEGIN
        FOR each groups as group

            /* Prints name of the Group*/
            PRINT Group Name       - group [name]

            /* Prints description of the Group*/
            PRINT Description      - group [descriptions]

            /* Prints whether belongs to discussion group or not*/
            PRINT Discussion group - group [discussion group]

            /* Prints status of the Group whether active or not*/
            PRINT Active           - group [active]

            /* Options for Group Action*/
            CASE group Actions

                /* redirects to edit group interface with group array*/
                Edit    : CALL edit group with group
            ENDCASE

        ENDFOR
    END
ENDFUNCTION
```

Figure 5-19: View Group Pseudocode

5.2.18 Edit Group

The Edit Group function creates an interface for editing group information in the database and provides the facility to replace the previous values with new changed values by the user.

```
FUNCTION edit group with group []
    BEGIN
        CREATE user input form fields
        POPULATE values from group to the input fields

        /* set the new name of the group to the user input group name*/
        SET group[name] to changed value

        /*set the new description of the group to the user input group
description*/
        SET group[description] to changed value

        /*set the new discussion group value to the user input discussion group */
        SETgroup[discussion group] to changed value

        /*set the new status of the group to the user input status whether active
or not*/
        SET group[active] to changed value

        CASE group action
            Submit  : VALIDATE all the fields for valid Inputs
                      UPDATE group information corresponding to group [id] into
database

                    /* redirects to group interface*/
                    CALL groups
        ENDCASE

    END
ENDFUNCTION
```

Figure 5-20: Edit Group Pseudocode

5.2.19 Add Group

The Add Group function creates an interface for adding a new group to the database and redirects the user to the group's interface. All the details are saved in the database.

```
FUNCTION add group
    BEGIN
        CREATE user input form fields

        /* sets name of the group */
        SET group[name] to changed value

        /*sets the description of the group to */
        SET group[description] to changed value

        /*sets the discussion group value */
        SET group[discussion group] to changed value

        /*sets the status of the group */
        SET group[active] to changed value

        CASE group action
            Submit  : VALIDATE all the fields for valid Inputs
                      Create new group in the database with all these information

                    /* redirects to group interface*/
                    CALL groups
        ENDCASE

    END
ENDFUNCTION
```

Figure 5-21: Add Group Pseudocode

5.2.20 Feedback

The Feedback function sends the users' points of view to the administrator. It is actually a medium for the user to communicate with the administrator. The administrator receives an email and will be able to review and refine the voting system in order to improve it.

```
FUNCTION feedback
    BEGIN
        CREATE user input field for placing comments
        CASE
            Send    : send mail to the administrator
                    IF mail send succeeds
            /* If message sent successfully print this message on user screen */
                            PRINT Thank you!
                            PRINT Your feedback has been sent
                    ENDIF
        ENDCASE
    END
ENDFUNCTION
```

Figure 5-22: Feedback Pseudocode

5.3 Functionality of Ontology Evolution Platform

In Chapter 4, the three layers of the platform were explained. These are the Social Networking, Permission and Ontology Evolution layers. The first two layers are intended to ensure that the collaborative efforts of users are realised and the user and group access is controlled. The third layer supports the ontology evolution. On this platform, in particular with regards to the last layer, Web-Protégé is used to allow users to view and update the ontology. Therefore, pseudocodes for the ontology evolution platform are not presented as this research does not develop such Protégé; rather it utilises Web-Protégé that was developed by researchers at the Stanford Center for Biomedical Informatics Research (Tudorache, Vendetti et al. 2008).

The extended work on Protégé carried out at Stanford Research Center, i.e. Collaborative Protégé, allows users to update ontology in a group work setting. The architecture includes annotation of ontology components and change tracking (Tudorache, Vendetti et al. 2008). The former enables ontology experts and non-experts to make comments about the ontology before the evolution process takes place, while the latter is created to manage any conflicts due to multiple users being involved. Based on the direction where Protégé is developed, the focus remains on the modification of ontology. In other words, Web-Protégé serves as the tool to realise the change that is required to ontology rather than facilitating the change itself. Although multiple users are able to make suggestions or enter queries on the Protégé, these activities are not conducted in a structured manner.

Hence, the system that is developed in this study proposes a systematic collaboration of users in the ontology evolution process in a business setting. The platform assists the management of changes to ontology which later is realised through Protégé. With the emphasis on users' involvement in updating knowledge captured in the ontology, the system provides a forum where users are able to converse on a chosen issue related to ontology and contributes to the discussion based on individual expertise as well as knowledge. Consistently, the result of this timely dialogue leads to a change being made to ontology.

5.4 Conclusion

In this chapter, the pseudocodes for the main functions of the Lightweight Community-driven Platform have been presented. This includes Login, Control Panel, Project, Issue, View Issue, Edit Issue, Add Issue, Calculation, Vote Options, Vote Results, Users, View Users, Edit User, Add User, Groups, View Group, Edit Group, Add Group and Feedback. The chapter had discussed further the functionality of Ontology Evolution Platform.

5.5 References

Chang, E., T. Dillon, et al. (2006). Trust and Reputation for Service-Oriented Environments: Technologies For Building Business Intelligence And Consumer Confidence, John Wiley and Sons.

Tudorache, T., J. Vendetti, et al. (2008). Web-Protégé: A Lightweight OWL Ontology Editor for the Web. OWL: Experiences and Direction (OWLED) Workshop, Karslruhe, Germany.

Wongthongtham, P. (2006). A methodology for multi-site distributed software development. School of Information Systems. Perth, Curtin University of Technology. **Doctoral Degree**.

Every reasonable effort has been made to acknowledge the owners of copyright material. I would be pleased to hear form any copyright owner who has omitted or incorrectly acknowledged.

Chapter 6 – Validation and Verification of Lightweight Community Driven Approach

6.1 Introduction

In Chapters 4 and 5, the solution and the implementation of a lightweight community support approach are discussed respectively. The approach along with its three key principles has been elaborated in Chapter 4, whereby the first principle, conceptual framework, is elucidated. The chapter explores the two key solutions for ensuring community users' involvement and providing an ontology evolution platform. In Chapter 5, the second key principle which is the implementation of Lightweight Community-driven Approach is provided in detail. For implementation purposes, pseudocodes of the platform were explicated. This substantiates the feasibility of this approach with regards to the platform development process. Subsequently, the underlying concept of the platform needs to be justified through the validation and verification process.

Within this chapter, the last key principle, validation and verification of the lightweight community support approach, will be presented through a working system which will be shown in a series of screenshots. A small sample of oil and gas ontology is created and utilised to describe the ontology evolution process. This chapter starts with a description of ontology evolution platform's evaluation using a sample of oil and gas ontology, and then is followed by a demonstration of the prototype of the voting system and ontology evolution platform. In addition, the chapter discusses the results the demonstration that has been presented, and then concludes.

6.2 Sample of Oil and Gas Ontology

In order to validate the ontology evolution platform, a small example of oil and gas ontology has been created. In this section, the features of oil and gas ontology (Classes, Properties and Individuals) are explained using screenshots of the current ontology which all staff are using at present. Further, the evolution of ontology is demonstrated in detail where component of Ontology are shown.. For this research, the ontology is created using Protégé version 3.4.4 which is compatible with Web Protégé 0.5 alpha build 300.

6.2.1 Oil and Gas Ontology

This section contains components of the oil and gas ontology. This includes Classes, Prosperities and Individuals. Although the sample is not constructed from a particular organisation, it reflects a real-case scenario within this industry. Further details are provided below.

6.2.1.1 Classes

A Class contains a group of individuals which have common attributes. For example, Bahrain, Kuwait and Saudi Arabia are members of Country Class. Classes have hierarchy, where the higher class is called superclass and lower class is called subclass. Anything that is a member of the subclass is a member of the superclass. Based on Figure 6.1, owl:Thing is a superclass of Country and OilGasSystem. Noticeably, Country and OilGasSystem are subclasses of owl:Thing.

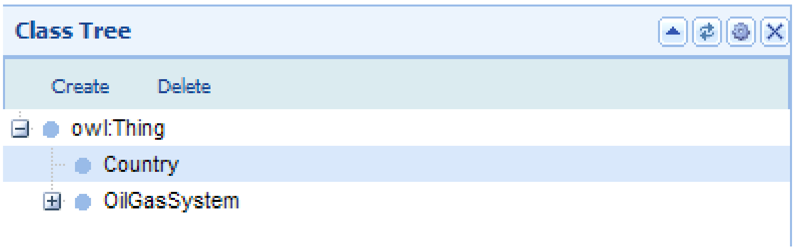
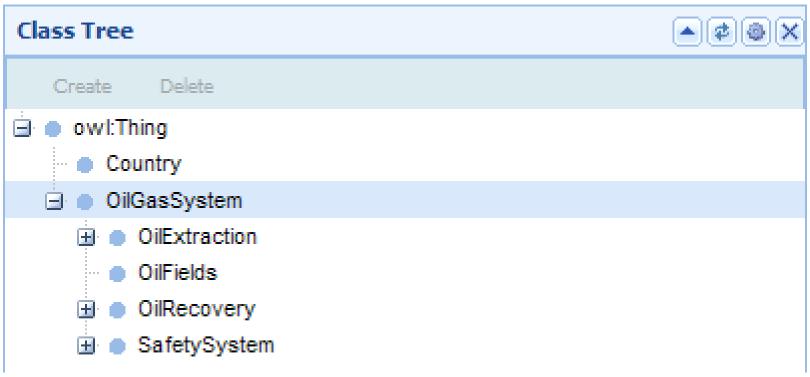


Figure 6-1: Country Class

From Figure 6.2 below, there are four subclasses in OilGasSystem, namely OilExtraction, OilFields, OilRecovery, and SafetySystem. These subclasses represent some of the components within oil and gas domain. Each component is explained below:

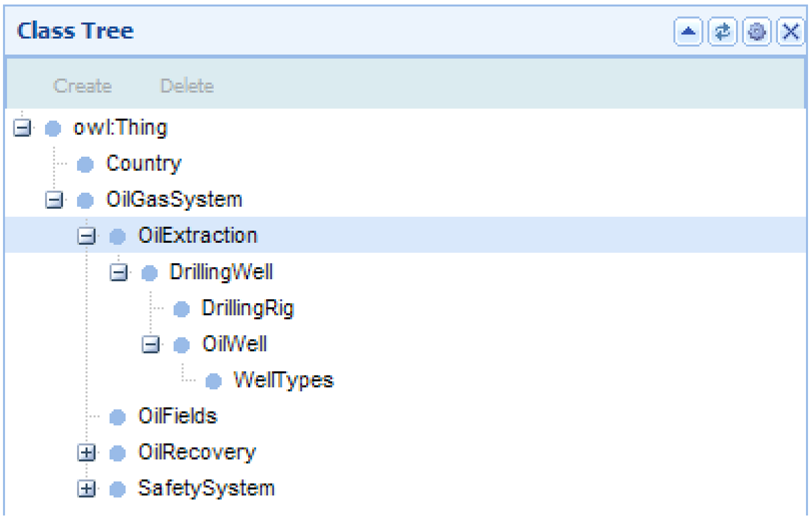
- **OilExtraction:** extracting oil is the main activity of the oil and gas industry. Organisations make efforts to discover areas where oil exists. Once the area has been located, they starts drilling to extract oil.
- **OilFields:** oil fields are the location that the organisation has discovered that contains oil.
- **OilRecovery:** as each organisation foresees the going concern of their business and potential hazard of natural destruction, they are compelled to have revitalization procedures in place.
- **SafetySystem:** safety is an essential factor across business sectors, even more so in the oil and gas industry. Due to the often hostile working environment within this particular industry, it is vital to ensure the safety of the entire staff and equipment.



```
⊟ owl:Thing
   Country
   ⊟ OilGasSystem
      ⊞ OilExtraction
        OilFields
      ⊞ OilRecovery
      ⊞ SafetySystem
```

Figure 6-2: Oil Gas System Class

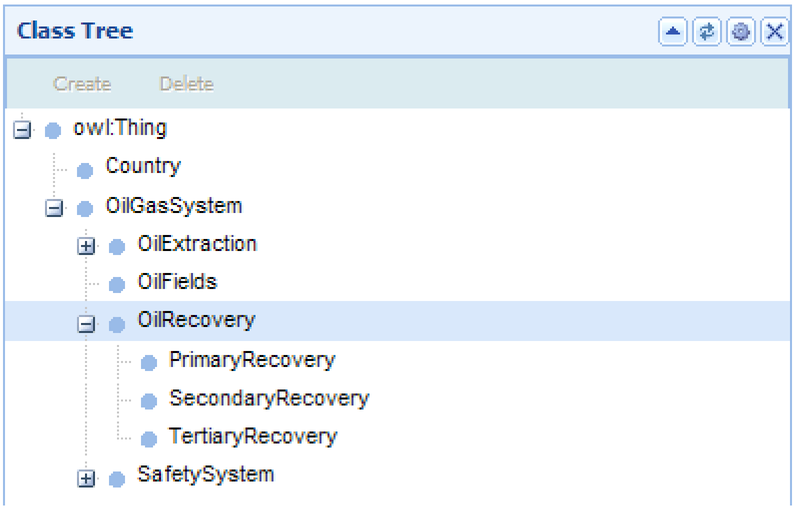
Following the same line of reasoning, Figure 6.3 shows one subclass of OilExtraction, namely DrillingWell. The two classes that have lower hierarchy than DrillingWell, i.e. they are the subclass of DrillingWell, are DrillingRig and OilWell. OilWell, in turn, has one subclass called WellTypes. As above, these are examples of members of the OilExtraction class. An organisation extracts oil from one or more wells; thus, drilling well and oil well become the subclass of OilExtraction. Oil wells are categorised in terms of: appraisal, exploration, reduction and wildcat as can be seen in a later section.



- owl:Thing
  - Country
  - OilGasSystem
    - OilExtraction
      - DrillingWell
        - DrillingRig
        - OilWell
          - WellTypes
      - OilFields
      - OilRecovery
      - SafetySystem

Figure 6-3: Oil Extraction Class

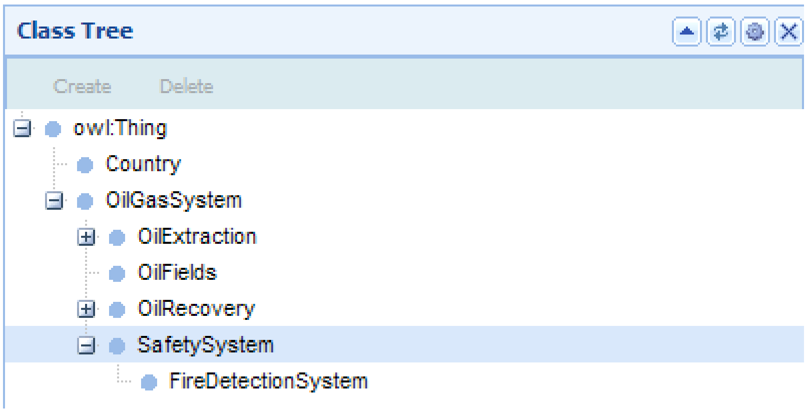
Every oil and gas organisation makes efforts to preserve oil production to ensure its business longevity. Therefore, precautions and remedial procedures are set up to protect the quality of the oil well and/or its surroundings from potential risks such as natural disaster, fire and human error. These procedures, known as oil recovery, differ from organisation to organisation, since they have the prerogative to design their own protocols according to each unique case. In Figure 6.4, three types of oil recovery protocols, namely Primary, Secondary and Tertiary Recovery, are used as an example. In this research, Primary Recovery is the first stage applied to a particular situation, followed by Secondary and Tertiary Recovery.



- owl:Thing
  - Country
  - OilGasSystem
    - OilExtraction
    - OilFields
    - OilRecovery
      - PrimaryRecovery
      - SecondaryRecovery
      - TertiaryRecovery
    - SafetySystem

Figure 6-4: Oil Recovery Class

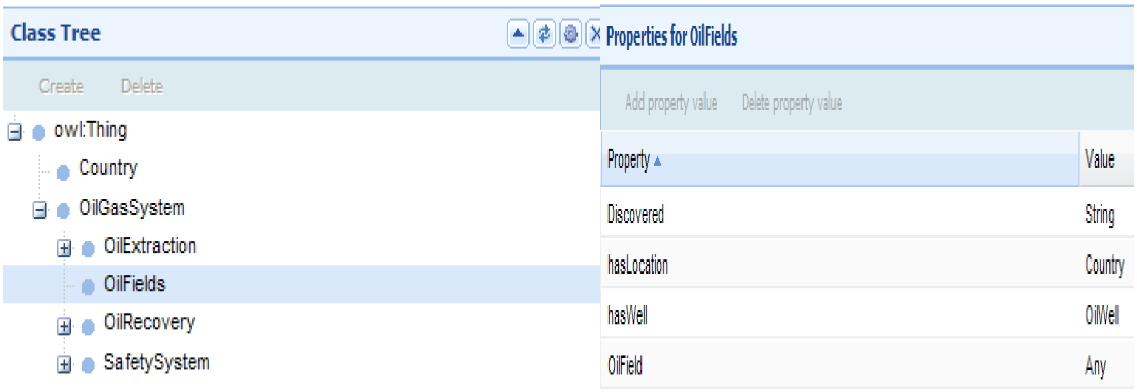
Due to its nature, the oil and gas industry requires rigorous protection against fire; thus, a fire detection system, as illustrated in Figure 6.5, is mandatory. It acts as a safety measure to prevent fatalities.



- owl:Thing
  - Country
  - OilGasSystem
    - OilExtraction
    - OilFields
    - OilRecovery
    - SafetySystem
      - FireDetectionSystem

Figure 6-5: Safety System Class

In Figure 6.6, properties that are attached to classes are shown. It shows all the different types of a particular class' property which is categorised in two types (i) Datatype Property, (ii) Object Property. For example, the OilFields class has three Datatype Property, namely Discovered, OilField and hasExhausted. The other two, hasLocation and hasWell, have Object Property.



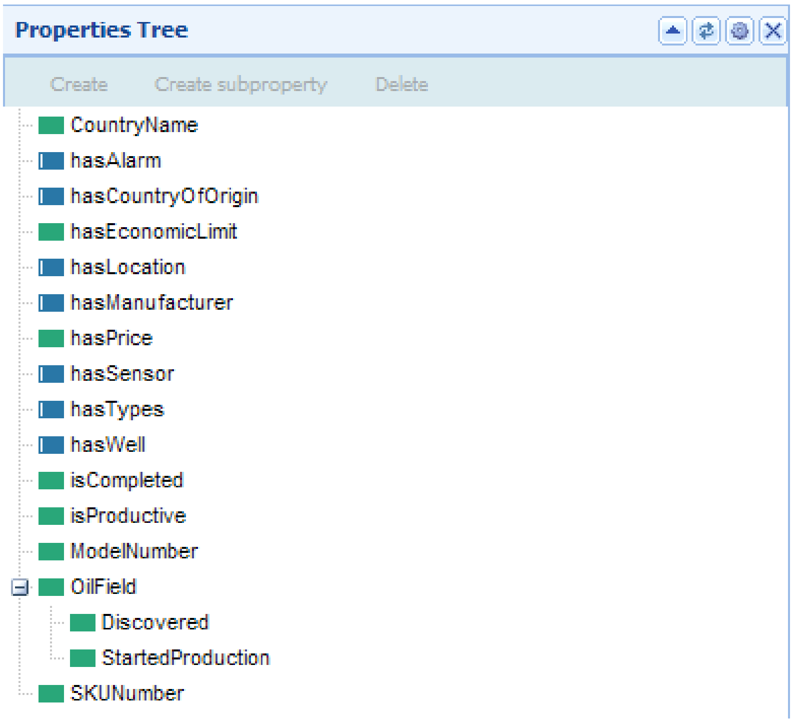
- owl:Thing
  - Country
  - OilGasSystem
    - OilExtraction
    - OilFields
    - OilRecovery
    - SafetySystem

Property ▲	Value
Discovered	String
hasLocation	Country
hasWell	OilWell
OilField	Any

Figure 6-6 Properties of Oil Fields Class

6.2.1.2 Properties

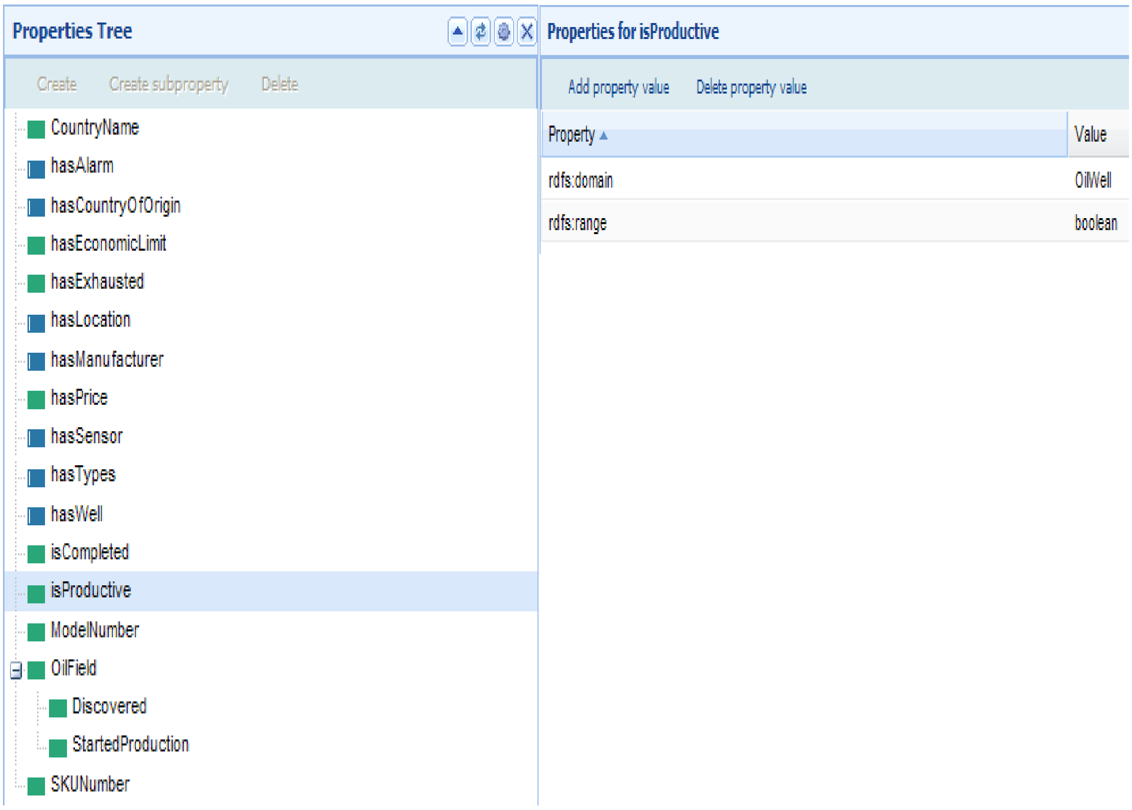
In general, Properties represent relationships. There are two types of Properties which are Datatype and Object Properties. Datatype Property connects a class (or classes) to a value such as integer or boolean. On the other hand, Object Property associates one class with another. Figure 6.7 lists some of the properties for the oil and gas industry which correspond to a particular class or classes. The items depicted in blue are Object properties and the green ones are Datatype properties. As stated previously, the difference between Datatype and object Properties will be further explained below.



- 🟩 CountryName
- 🟦 hasAlarm
- 🟦 hasCountryOfOrigin
- 🟩 hasEconomicLimit
- 🟦 hasLocation
- 🟦 hasManufacturer
- 🟩 hasPrice
- 🟦 hasSensor
- 🟦 hasTypes
- 🟦 hasWell
- 🟩 isCompleted
- 🟩 isProductive
- 🟩 ModelNumber
- 🟩 OilField
  - 🟩 Discovered
  - 🟩 StartedProduction
- 🟩 SKUNumber

Figure 6-7: Oil and Gas Ontology Properties

Each property has a domain and range. These features are not constraints for properties; rather, they provide association between properties and a class or classes. For example, in Figure 6.8, the property is Productive has an Oil Well domain and a boolean range. This means it provides further details on the productivity level of the oil well using the value ‘true’ or ‘false’. The value ‘true’ is assigned when an oil well produces oil higher than the organisation minimum standard. The value ‘false’ means the oil extracted is below the minimum standard.

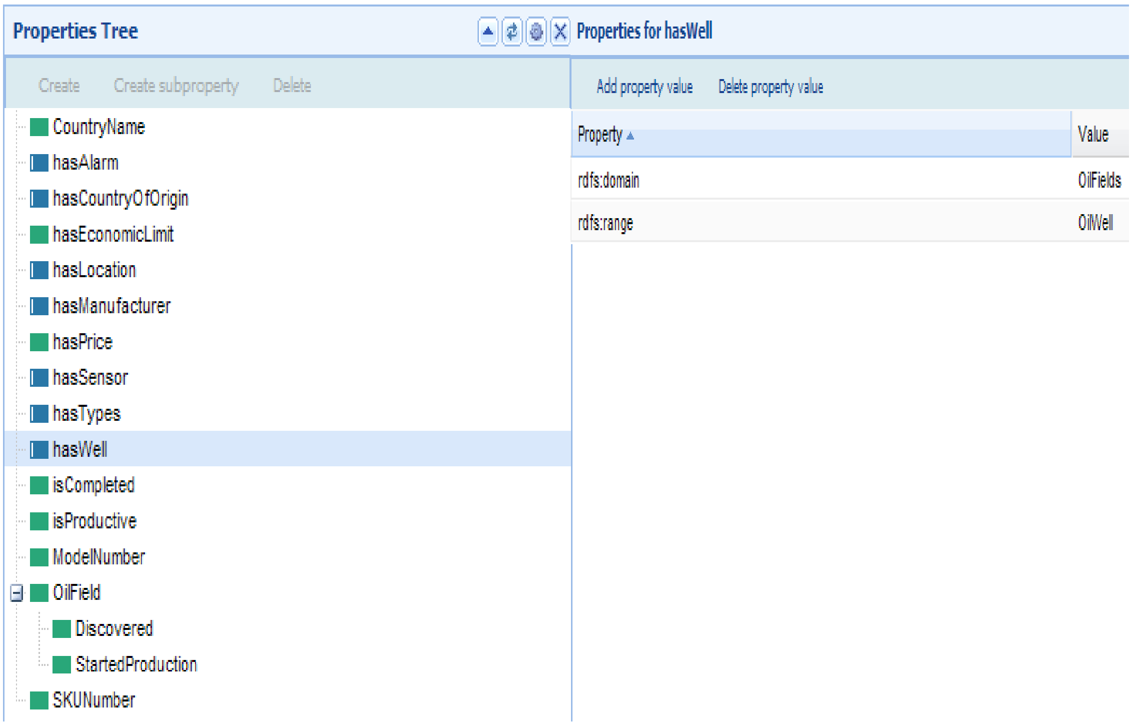


- CountryName
- hasAlarm
- hasCountryOfOrigin
- hasEconomicLimit
- hasExhausted
- hasLocation
- hasManufacturer
- hasPrice
- hasSensor
- hasTypes
- hasWell
- isCompleted
- isProductive
- ModelNumber
- OilField
  - Discovered
  - StartedProduction
- SKUNumber

Property ▲	Value
rdfs:domain	OilWell
rdfs:range	boolean

Figure 6-8: Datatype Properties

Figure 6.9 shows examples of Object properties within oil and gas industry. The domain for property hasWell is OilFields Class and the range is OilWell Class. Each oil field has one or more oil wells, thus the property hasWell shows the pathway of these two classes. This property will be linked with other individuals that will be explained further in the next section.



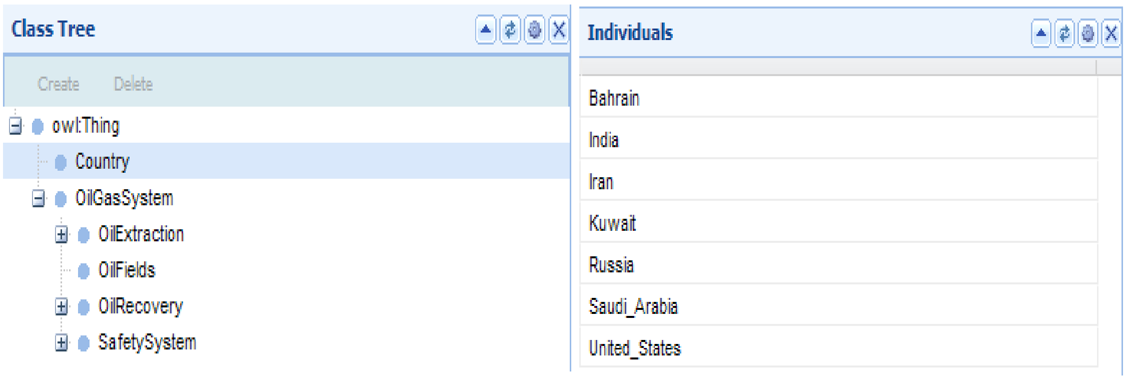
- 🟩 CountryName
- 🟦 hasAlarm
- 🟦 hasCountryOfOrigin
- 🟩 hasEconomicLimit
- 🟦 hasLocation
- 🟦 hasManufacturer
- 🟩 hasPrice
- 🟦 hasSensor
- 🟦 hasTypes
- 🟦 hasWell
- 🟩 isCompleted
- 🟩 isProductive
- 🟩 ModelNumber
- 🟩 OilField
  - 🟩 Discovered
  - 🟩 StartedProduction
- 🟩 SKUNumber

Property ▲	Value
rdfs:domain	OilFields
rdfs:range	OilWell

Figure 6-9: Object Properties

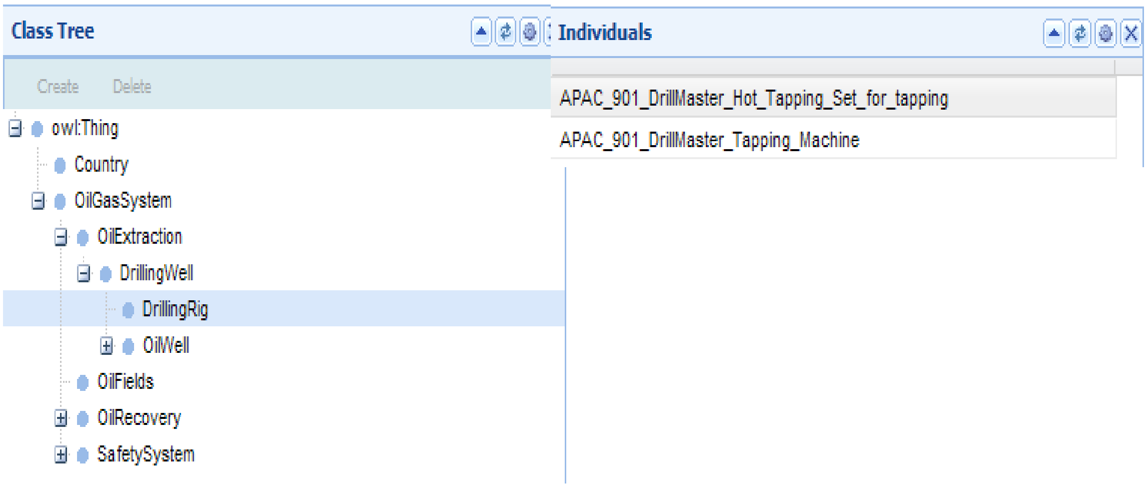
6.2.1.3 Individuals

Individuals or instances are objects in the oil and gas domain. Individuals are linked through properties to another individual. Figures 6.10 to 6.12 illustrate individuals for different classes. Country Class has individuals with names of countries where a particular oil and gas organisation has business operations. Figure 6.11, shows types of drilling rigs that this particular organisation uses in their field activities around the world. Figure 6.12, provides a list of the names of oil fields where oil is being extracted.



Bahrain
India
Iran
Kuwait
Russia
Saudi_Arabia
United_States

Figure 6-10: Individuals for Country Class



APAC_901_DrillMaster_Hot_Tapping_Set_for_tapping
APAC_901_DrillMaster_Tapping_Machine

Figure 6-11: Individuals for Drilling Rig

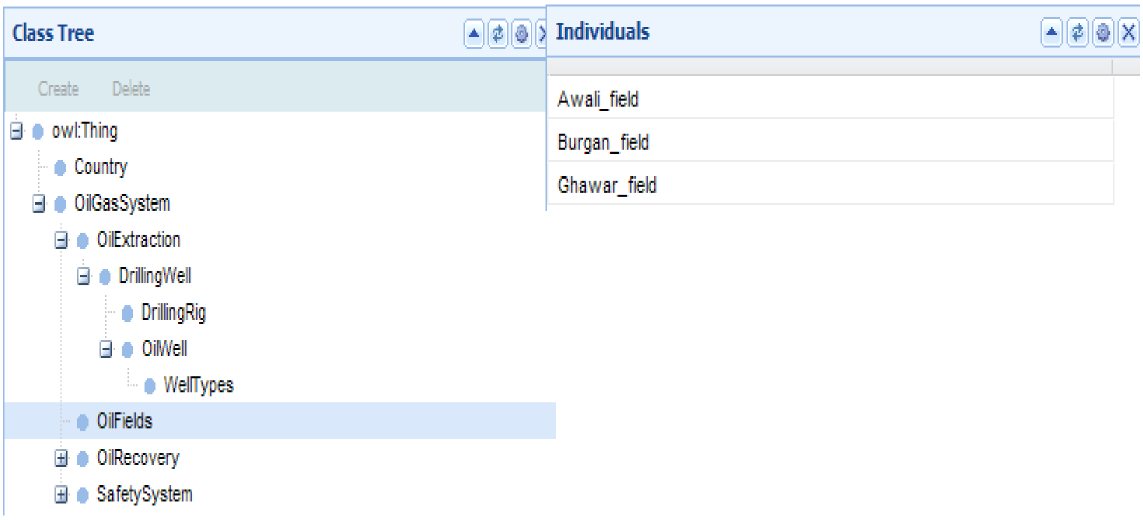


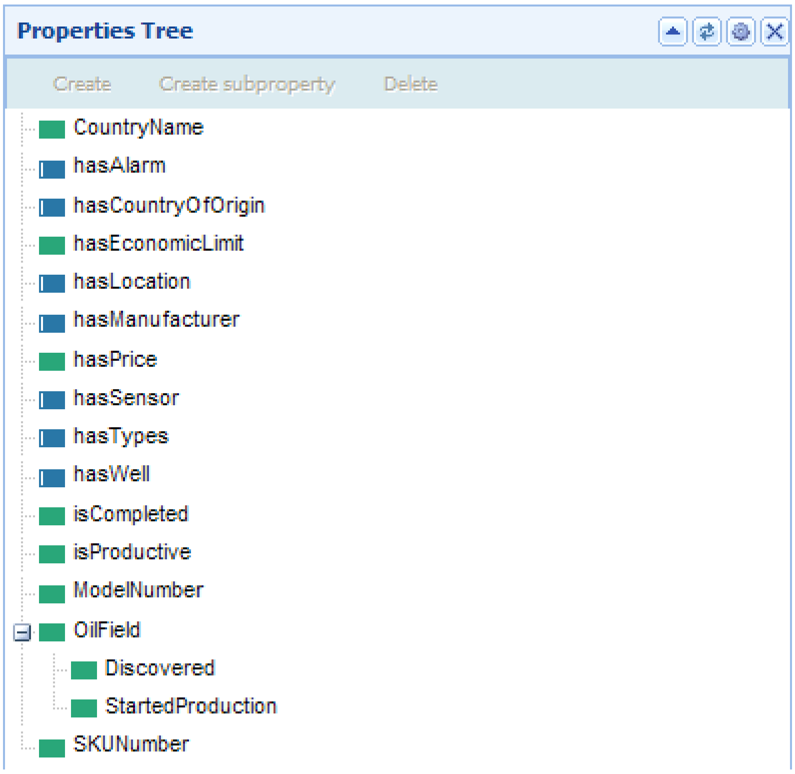
Figure 6-12: Individuals for Oil Field

6.2.2 Oil and Gas Ontology Evolution

As explained in Chapter 4, there are three types of ontology, namely Current Ontology, Temporary Ontology and Archived Ontology. Current Ontology is the existing ontology that is used by employees in a particular oil and gas organisation. Each type of ontology can be used to inform the classification system, which in turn can be used to inform how data ordered within the system can be put into use. Of these three types of ontologies, Current Ontology has all the necessary Class, Properties and Individuals as described in detail previously. It is the base of the ontology evolution process.

Temporary Ontology is a set of ontology that is created by the ontology engineer upon a request from members of the Contributor Group and Archived Ontology is an old version of updated Current Ontology. From Chapter 5 it was obvious that ontology is updated only when “Modified Issue” receives the most number of votes. Therefore, ontology is not archived if “Unresolved Issue” or “Viewpoint Issue” has the most voting points. To illustrate the ontology evolution, an example using the Burgan oil field is used.

Suppose that the team leader finds that one of the organisation oil fields, the Burgan oil field, has experienced a decrease in oil productivity as much as 30% within the last five years. This alarming information is crucial since the Burgan oil field is considered as one of the biggest oil fields in the world in which the organisation has a huge stake. From the business perspective, the organisation has two options: either to allocate substantial resources to a rejuvenation program for this oil field or to develop an exit strategy. This includes the team leader recommending a modification to the existing ontology to accommodate the possibility of an oil field being exhausted, i.e. no oil being available for extraction. Figure 6.8 illustrates Properties within Current Ontology which has “isProductive”, but does not reflect the condition of depletion. The team leader is free to ask the ontology engineer to create a temporary ontology based on his proposal that “hasExhausted” be added to Properties in Datatype Property.

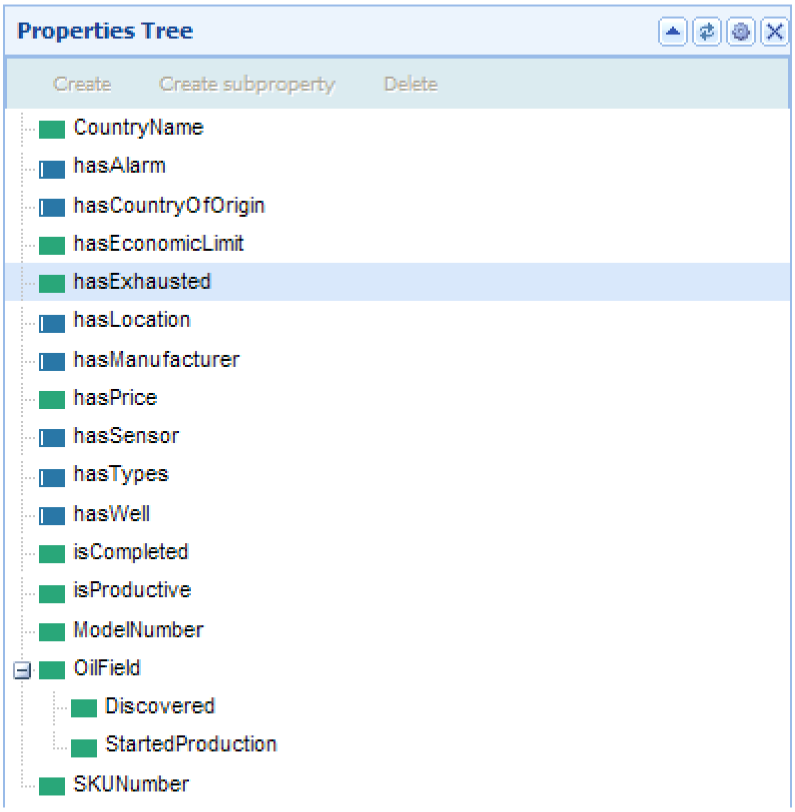


- 🟩 CountryName
- 🟦 hasAlarm
- 🟦 hasCountryOfOrigin
- 🟩 hasEconomicLimit
- 🟦 hasLocation
- 🟦 hasManufacturer
- 🟩 hasPrice
- 🟦 hasSensor
- 🟦 hasTypes
- 🟦 hasWell
- 🟩 isCompleted
- 🟩 isProductive
- 🟩 ModelNumber
- 🟩 OilField
  - 🟩 Discovered
  - 🟩 StartedProduction
- 🟩 SKUNumber

Figure 6-13: Current Ontology before Discussion

While the discussion takes place, other participants are encouraged to provide feedback and comments on the proposed ontology. As stated earlier, these participants can also propose their own version of ontology, creating multiple temporary ontologies. Each member of the Contributor Group then votes on the ontology that he/she considers as the most appropriate for the issue. The system acknowledges one man, one vote and single entry mode. This means that once a vote has been cast, it cannot be revised, even though the issue is still open for discussion. Thus, voters have to be confident about their decisions before they vote on any option.

From Figure 5.21 in Chapter 5, “Modified Issue” earned the highest voting point of 6.6; therefore, the existing ontology needs to be modified. In order to make the explanation less complicated, it is assumed that the ontology proposed by the team leader is the most favoured throughout the discussion process. Thus, the ontology engineer adds “hasExhausted” to Datatype Property as well as the related Individuals. The updated Current Ontology is presented in Figure 6.14, while Figure 6.13 becomes Archived Ontology.



- CountryName
- hasAlarm
- hasCountryOfOrigin
- hasEconomicLimit
- hasExhausted
- hasLocation
- hasManufacturer
- hasPrice
- hasSensor
- hasTypes
- hasWell
- isCompleted
- isProductive
- ModelNumber
- OilField
  - Discovered
  - StartedProduction
- SKUNumber

Figure 6-14: Current Ontology after Discussion

6.3 Ontology Evolution Platform Evaluation

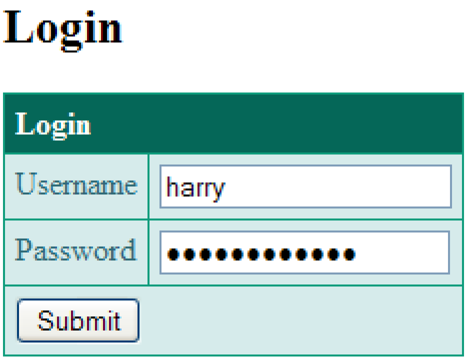
In this section, a prototype demonstration of the platform is presented. The platform presented the lightweight community support and ontology evolution support. As explained, the lightweight community support represents the increased involvement of users in the ontology evolution process. Staff are authorised to raise issues and are expected to provide comments on the issues that would require an action to be taken on the existing ontology. This initiates the entire dialogue process on a selected issue in a participatory manner. By the indicated due date decided by staff who raised the issue, each participant is required to vote on the issue. Then, a calculation is automatically generated by the platform for each vote as described in Chapter 5.

The conceptual framework that was discussed in Chapter 4 involves two main groups, namely the Contributor Group and the Admin Group. It was well-established that the two groups have different roles and responsibilities, which has been extensively elaborated in Chapter 4. To express it succinctly, the Contributor Group is authorised and responsible for the process and quality of the outcome of communication, since members of this group are experts within their own rights. The Admin Group is tasked with and accountable for ensuring the eligibility of each member of both groups. This means that the Admin Group is in charge of the authenticity of the discussion. The Admin Group is also responsible for maintaining the integrity of the ontology so it reflects current data and information, and remains relevant to the discussion.

In this section, a Lightweight Community-driven Approach is demonstrated using the following example. User accounts for Tino, Ponny, Harry, and John are created. Tino is assigned as a domain expert; Ponny is an ontology engineer, Harry is a team leader and John is a team member. In order to demonstrate how the platform functions, visualisation in the form of screenshots is presented and explained in detail.

6.3.1 Login

Each User has a username and password which provides him/her with the correct permission. The user is required to enter the username and password which has been provided by the administrator as shown in Figure 6.15.

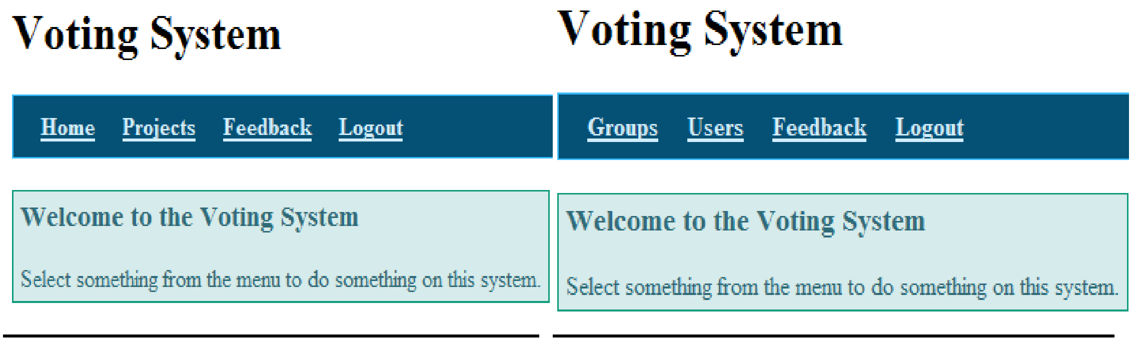


Username: harry

Password: ●●●●●●●●●●●●

Figure 6-15: Login

Once the user enters the login details, the platform checks what permission the user has. This will be providing access of either what the Contributor Group or admin group has. As stated previously in Chapter 4, the Contributor Group is able to raise and discuss an issue or project, and vote on an issue. The Admin Group, on the other hand, is responsible for management of groups and users. Issues that are raised by users are related to projects that are undertaken by the organisation. Therefore, the platform makes this available to the Contributor Group. Users who raise issues have to select a Project to which that particular issue belongs. Figure 6.16a and 6.16b below illustrate it further.



Welcome to the Voting System

Select something from the menu to do something on this system.

Welcome to the Voting System

Select something from the menu to do something on this system.

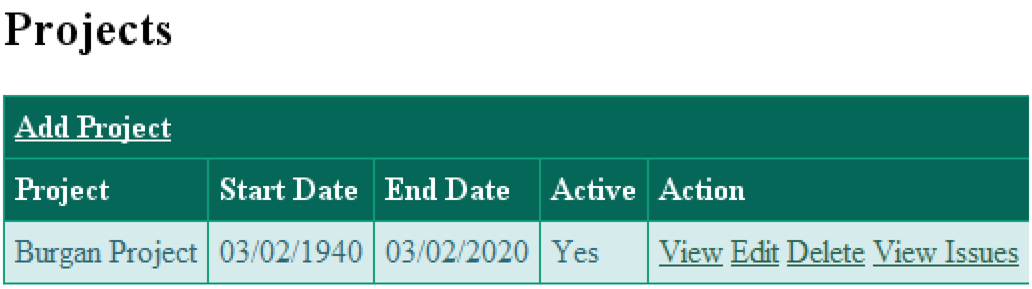
Figure 6.16a Contributor Group Access **Figure 6.16b Admin Group Access**

Figure 6-16: Access for Groups

Figure 6.16a displays features that are available to members of the Contributor Group. Each member has access to all options related to discussion of issues and can vote on an issue and view the vote result. However, a team leader possesses additional authorisation, namely to Add, Edit and Delete a project. In a nutshell, the team leader retains complete authorisation of the platform. Domain experts and team members, on the other hand, are not granted the same privilege since they are not allowed to create a project. Further demonstrations of these features are provided in the next section.

6.3.2 Project

Once the team leader clicks on the Projects link, a list of projects which is available on the platform will be rolled down. In Figure 6.17, an example of projects is given. A Burgan Project is an ongoing project that the organisation has had since February 1940 and has only recently been added to the system. The page also provides a timeframe for the project. The team leader has the right to add projects and fill in the fields related to it, including the start and the end date for the project. The next field, "Active", reflects the status of the project, if it is still ongoing ("Yes") or completed ("No").



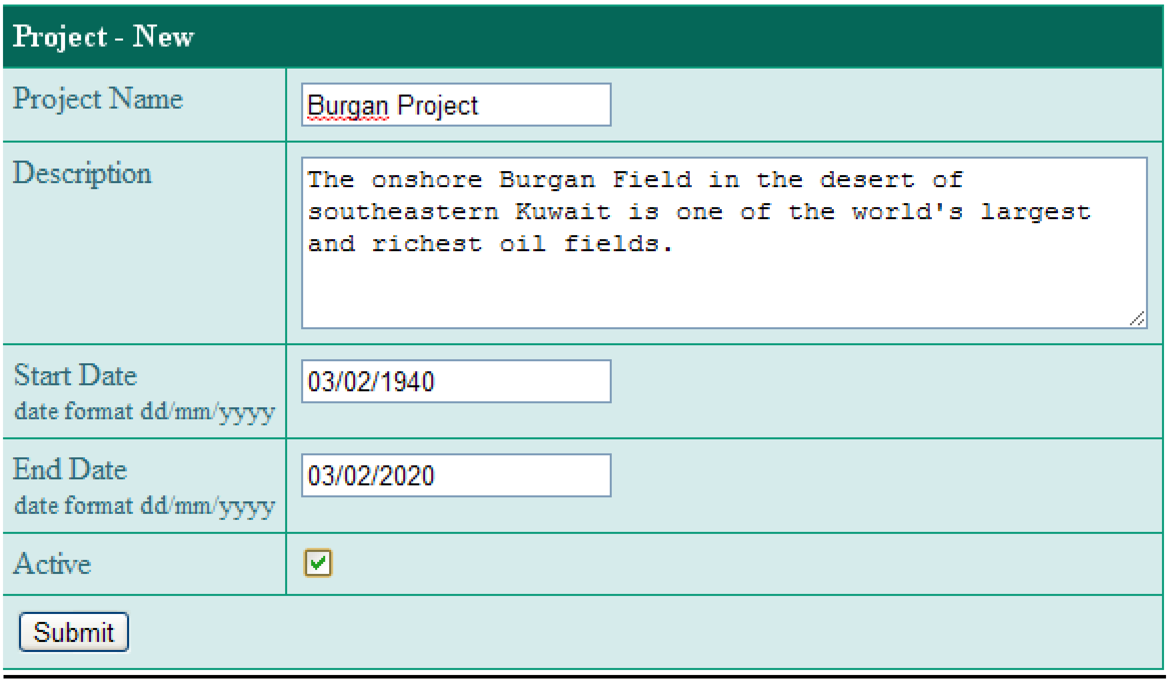
Projects

Add Project

Project	Start Date	End Date	Active	Action
Burgan Project	03/02/1940	03/02/2020	Yes	View Edit Delete View Issues

Figure 6-17: List of Projects

Figure 6.18 shows how this "Active" field is determined. A "Yes" on the "Active" field on Figure 6.17 is created by clicking the empty box next to "Active" listed as seen on Figure 6.18. When a project' "Active" field shows a "No", this means no actions can take place. The last field "Action" provides the Team Leader with the chance to view ("View"), edit ("Edit"), delete ("Delete") and view issues ("View Issues") of a project. The last feature is available for all users and should be used to gain access to discuss an issue. Taking into account that the team leader is high-ranking personnel in the organisation with recognised authorisation, including creating a project, he is entitled to complete all information needed on this page. They include a name of a project ("Project Name"), a brief description of the project ("Description"), duration of the project ("Start Date" and "End Date"), and status of the project ("Active"). Upon completion, the team leader clicks on the "Submit" button and a new project is created.

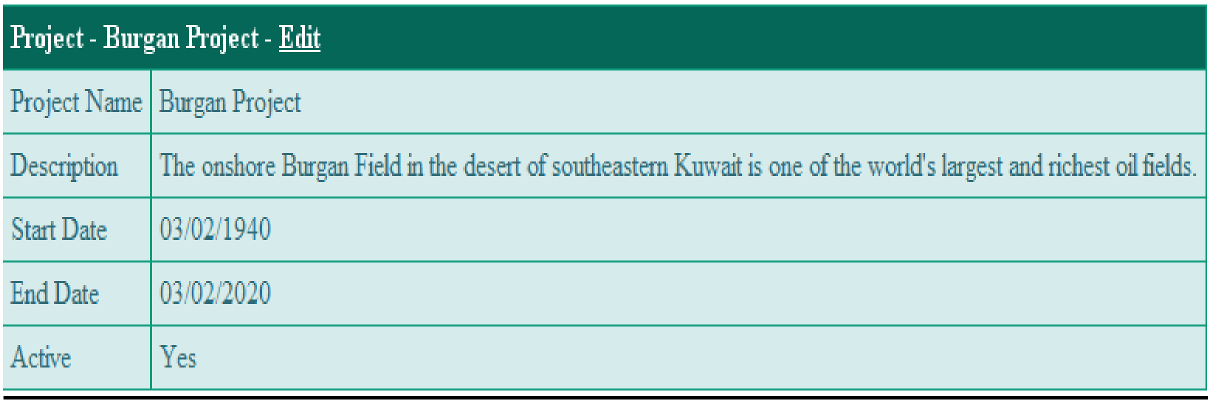


Project - New

Project Name	Burgan Project
Description	The onshore Burgan Field in the desert of southeastern Kuwait is one of the world's largest and richest oil fields.
Start Date date format dd/mm/yyyy	03/02/1940
End Date date format dd/mm/yyyy	03/02/2020
Active	☑

Figure 6-18: Add Project

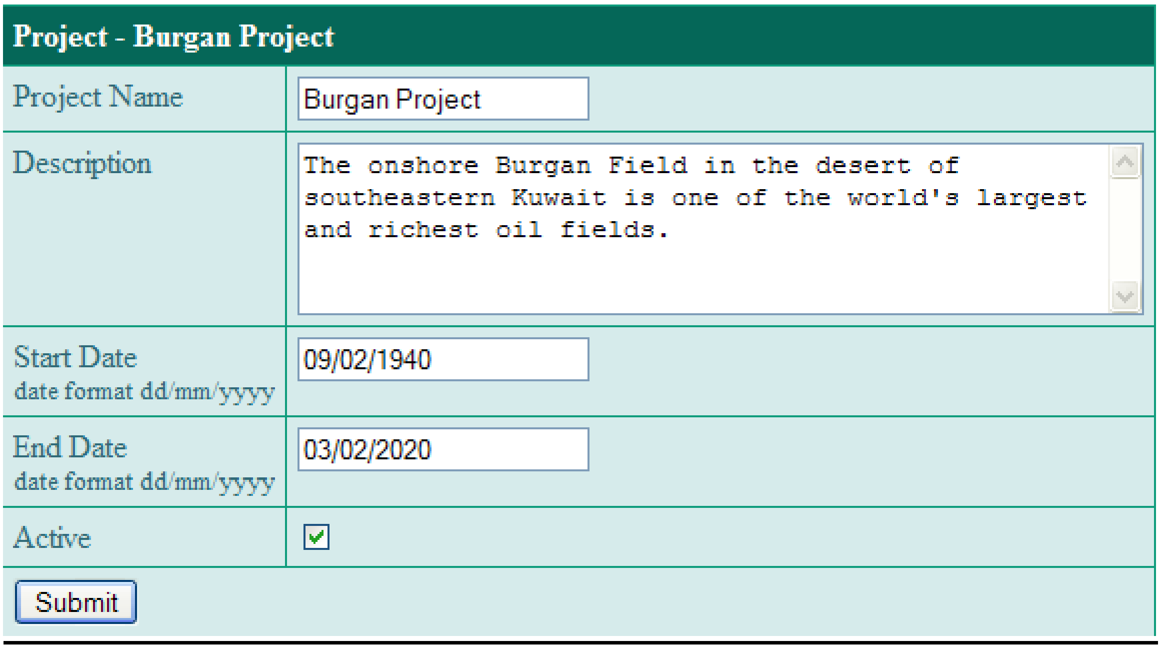
Figure 6.19 below illustrates the fields resulting from what the team leader has produced. In the future, the team leader will also be able to edit the project using a click on “Edit” as shown in Figure 6.17. The page that appears on edit project is shown in Figure 6.20.



Project - Burgan Project - Edit	
Project Name	Burgan Project
Description	The onshore Burgan Field in the desert of southeastern Kuwait is one of the world's largest and richest oil fields.
Start Date	03/02/1940
End Date	03/02/2020
Active	Yes

Figure 6-19: View Project

As shown in Figure 6.20, the team leader is free to revise any information that he created previously. For example, the team leader was informed that the working pace on Burgan Oil Fields has decelerated due to a spreading concern over a possible outbreak of bird flu that has brought a high number of fatalities across the region. As the foreseen epidemic was unavoidable, the team leader could extend the project’s timeframe and change the end date on the Burgan Project to 3 February 2025 to make provision for the unexpected event. This means activities related to the Burgan Project have a new deadline and timetable.



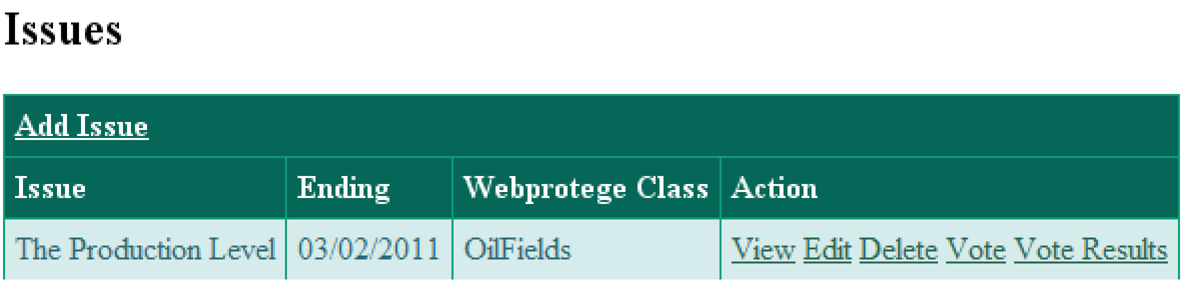
Project - Burgan Project	
Project Name	Burgan Project
Description	The onshore Burgan Field in the desert of southeastern Kuwait is one of the world's largest and richest oil fields.
Start Date date format dd/mm/yyyy	09/02/1940
End Date date format dd/mm/yyyy	03/02/2020
Active	☑
Submit	

Figure 6-20: Edit Project

Consistently, the team leader also possesses the mandate to delete a project. S/He is able to exercise this by clicking the “Delete” button as displayed in Figure 6.17. All users (Tino, Harry, Ponny and John), then, are able to start discussing issues by selecting “View Issues” as seen in Figure 6.17. Another page titled “Add Issue” will be opened (Figure 6.21) and is further discussed below.

6.3.3 Issue

This section discusses the communication that is carried out by users. Users are able to View, Edit, Delete, and Vote as well as view the Vote Result on the issue which has been raised. Once the issue is listed as shown in Figure 6.21, everyone is able to perform any actions on each issue accordingly. Further details of each action are explained below.



Issues

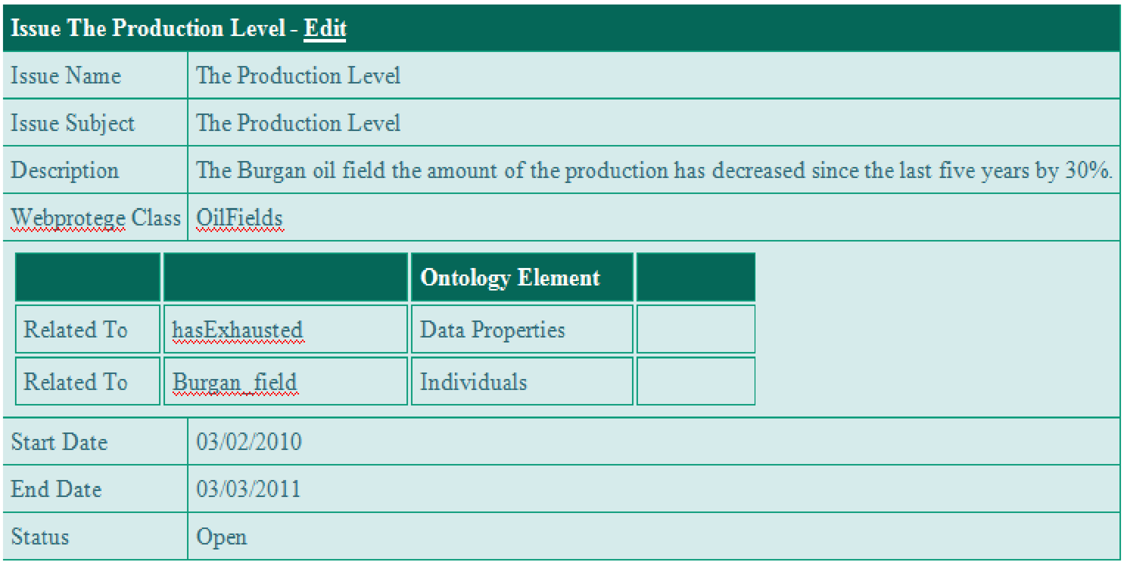
Add Issue

Issue	Ending	Webprotege Class	Action
The Production Level	03/02/2011	OilFields	View Edit Delete Vote Vote Results

Figure 6-21: List Issue

The first action that members of the Contributor Group can take is to view the issue in detail in order to understand what the issue is related to. From Figure 6.22 below, it can be seen that the issue currently being discussed is the level of oil production of the Burgan Oil Field. On this “Issue The Production Level” page, users acquire more knowledge about the issue. The data on the issue page is entered by a user who raises the issue, which in this example could be the team leader, ontology engineer, domain expert or team member. As stated, the issue is that the production level has deteriorated over the last five years by an alarming 30%. The previous event has led to another finding, namely the upsetting decrease of productivity.

The page also informs users about Web-Protégé Class and Ontology Element. The Ontology Element consists of seven options, namely “Classes”, “Data Properties”, “Object Properties”, “Quantifier Restrictions”, “Universal Restrictions”, ”Cardinality Restrictions” and “Individuals”. Each of these options refers to the part of the ontology that needs to be evolved. Other elements that are included on this page are the time limit of the issue. Users need to be informed about the termination date of the discussion. Every discussion should be limited to a certain period of time in order to produce results. Thus, the column “End Date” shows users when the discussion is expected to end. It has the date when the issue was opened for discussion and another date for when it should be closed. The length of time of the discussion is determined by the person who raises the issue. The last field, which is the “Status” will show “Open” until the end date is reached. Further, the issue can be edited on a page titled “Edit” as shown at the top of the “View Issue” page as can be seen in Figure 6.22.



Issue The Production Level - Edit	
Issue Name	The Production Level
Issue Subject	The Production Level
Description	The Burgan oil field the amount of the production has decreased since the last five years by 30%.
Webprotege Class	OilFields

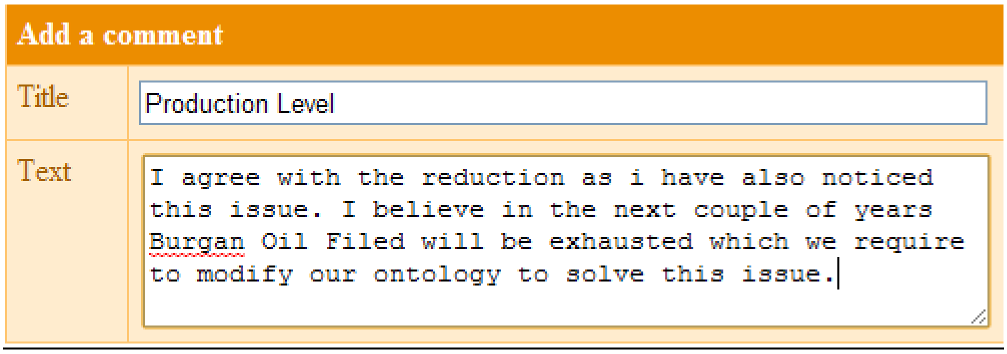
		Ontology Element	
Related To	hasExhausted	Data Properties	
Related To	Burgan_field	Individuals	

Start Date	03/02/2010
End Date	03/03/2011
Status	Open

Figure 6-22: View Issue

Once an issue has been raised, the staff are informed by an email which is sent automatically by the platform. Within the same page of "View Issue", there is another section which allows users to provide their comments on the issue as illustrated by Figure 6.23. Staff are expressing their opinions on the raised issues by way of "Add a comment" that allows Ponny, John or Tino to comment on the team leader's opinion as shown in Figure 6.23. This feature is not limited to staff with pertinent knowledge or expertise; rather, it provides equal opportunities for every member of the Contributor Group to participate in an open forum. The argument for this is that variety of knowledge and expertise is considered to enrich the discussion process.

The team leader is able to response to these comments using the same window on his screen as illustrates in Figure 6.23. Harry, The team leader, commented on the raised issue and submitted his opinion by clicking the "submit" button. This discussion process reflects a collaborative communication among staff regardless of where they are located.

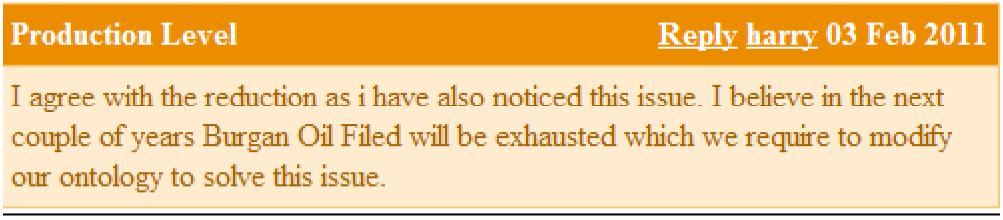


Add a comment

Title: Production Level

Text: I agree with the reduction as i have also noticed this issue. I believe in the next couple of years Burgan Oil Filed will be exhausted which we require to modify our ontology to solve this issue.

Figure 6-23: Add Comment

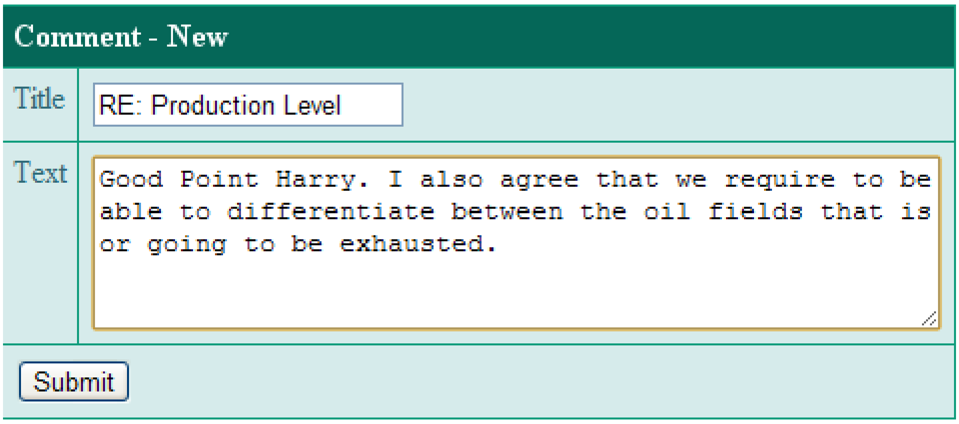


Production Level — Reply harry 03 Feb 2011

I agree with the reduction as i have also noticed this issue. I believe in the next couple of years Burgan Oil Filed will be exhausted which we require to modify our ontology to solve this issue.

Figure 6-24: View Feedback

Figure 6.24 above shows the comment from the Team Leader as seen by other users. They are able to login, reply to the Team Leader's comment, and raise their opinions. This is done by simply clicking on View Issue under the Issue List page. Adding a reply is as simple as clicking on Reply on the view issue page as shown in Figure 6.24. Once other users click on the "Reply" link, each one of them will be able to provide a reply to the Team Leader's comments as shown in Figure 6.25.



Comment - New

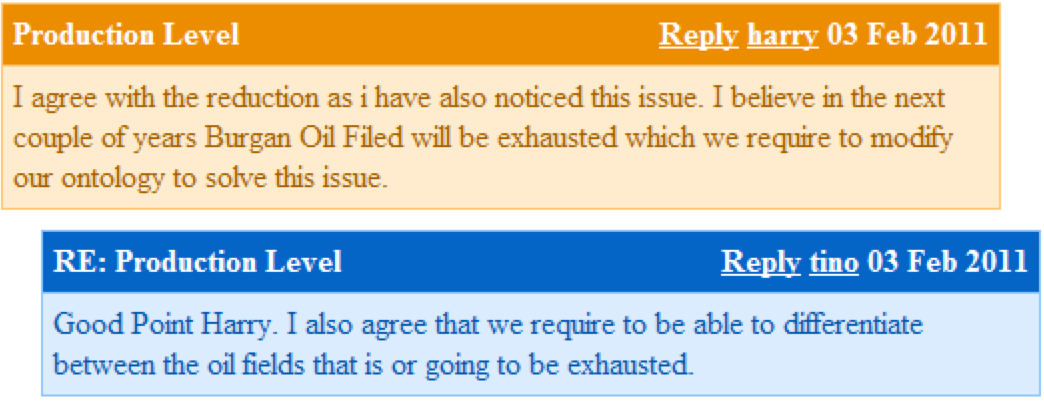
Title: RE: Production Level

Text: Good Point Harry. I also agree that we require to be able to differentiate between the oil fields that is or going to be exhausted.

Figure 6-25: Add Reply

Another user, for example Tino, submits his reply by clicking the "Submit" button. Upon submission, Tino's reply is accessible to all the users within the Contributor Group to view and they provide their replies on each other's comments as shown in Figure 6.26. In addition to this communication, a delete feature is also available.

Principally, the right to delete the issue stays with the person who raises that particular issue. In this example, The Team Leader retains the authority to remove the issue. However, in special circumstances such as the Team Leader being away on his annual leave, the Team Leader is granted with the power to do it. A deletion of an issue is possible when it is established that the issue is irrelevant to the project.



Production Level Reply harry 03 Feb 2011

I agree with the reduction as i have also noticed this issue. I believe in the next couple of years Burgan Oil Filed will be exhausted which we require to modify our ontology to solve this issue.

RE: Production Level Reply tino 03 Feb 2011

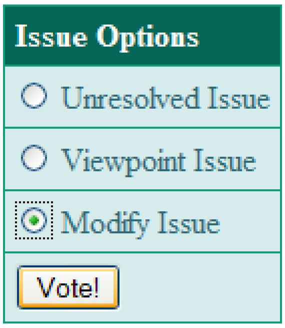
Good Point Harry. I also agree that we require to be able to differentiate between the oil fields that is or going to be exhausted.

Figure 6-26: View Reply

As the "Comment" page is open to everyone, each staff member is responsible for reading the comments provided by their colleagues before casting his/her vote. The discussion allows users to clarify any concerns or misunderstandings about the issue before voting on it. The voting features are explained further in the next section.

6.3.4 Vote Options and Vote Results

Once the users have had the opportunity to discuss the issue, a vote option is also available to all users within the Contributor Group. There are three options to choose from; (i) Unresolved Issue, (ii) Viewpoint Issue and (iii) Modify Issue as shown in Figure 6.27.



Issue Options

○ Unresolved Issue

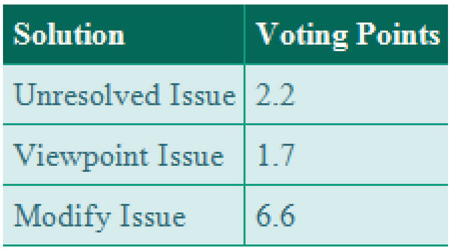
○ Viewpoint Issue

● Modify Issue

Figure 6-27: Vote Options

An unresolved issue occurs when the users do not find any solutions to the raised issue. In this case, participants in the discussion concur that the issue with the ontology remains; however, they have not come to an agreement on the actions that should be applied to it. When an issue is classified as a Viewpoint issue, this means that the ontology should stay the same. The last option, Modify issue, means that a proactive measure such as revision of the ontology is demanded.

The users will also be able to view the vote results based on what the users have voted. This is done by clicking on Vote Result on the List Issue page. The Vote Result provides the points for each option which is chosen. The point is calculated differently for each user as discussed further in the next section in this chapter. The result of the calculation is given in the Vote result page as shown in Figure 6.28.



Solution	Voting Points
Unresolved Issue	2.2
Viewpoint Issue	1.7
Modify Issue	6.6

Figure 6-28: Vote Results

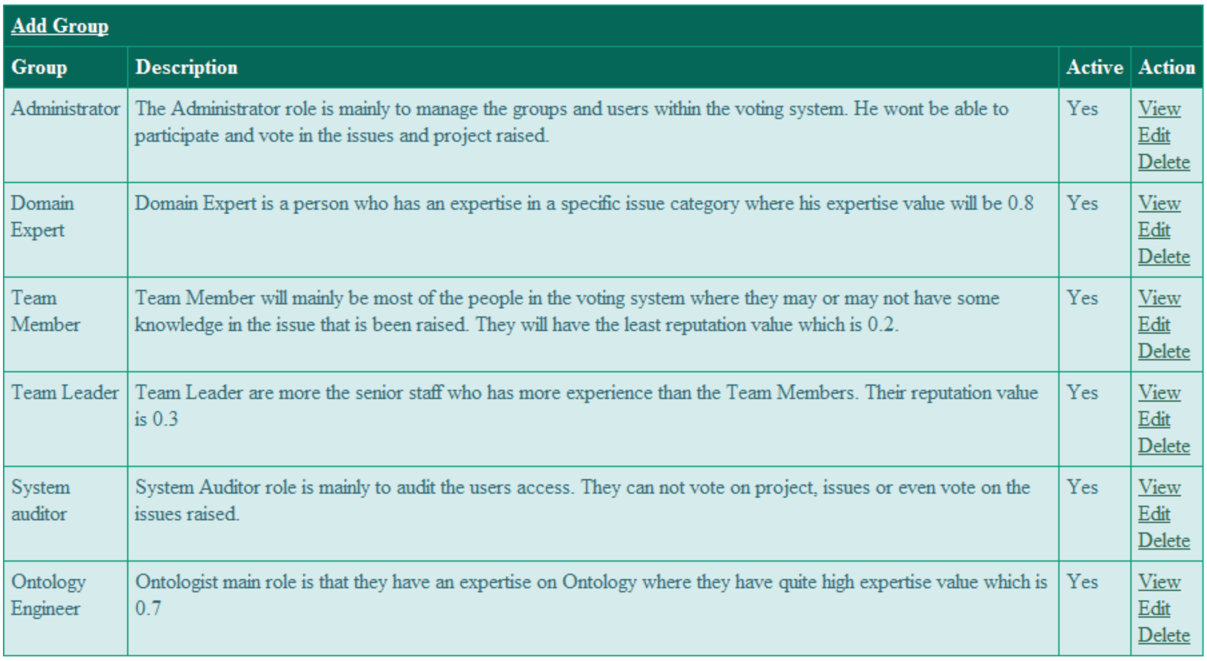
The voting feature on the platform indicates that a systematic discussion has occurred prior to the execution of a vote by each member. Thus, the decision to modify or to stay with the particular ontology justifies the quality of the discussion.

6.3.5 Groups

All the features discussed above were for the users who are members of the Contributor Group. This allows the group to raise and discuss issues. However, the Admin Group has different features from those of the Contributor Group to correspond with their role - that is, to be responsible for the users and the group to which they will be assigned.

For the first feature, when a user logs in as an administrator, s/he has three options; (i) Groups, (ii) User (iii) Feedback and (iv) Log out as shown previously in Figure 6.16b.

The administrator then selects and clicks on the Group link, where a list of members of the groups will be displayed. They are Administrator, Domain Expert, Team Member, Team Leader and System Auditor. The "Description" field contains the roles and responsibilities as comprehensively described in the previous section. The administrator will be able to add, view, edit or delete a group as shown in Figure 6.29 below.

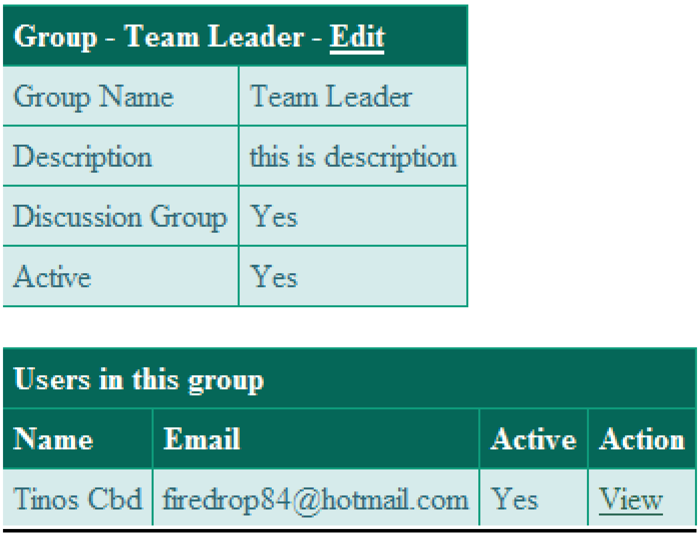


Add Group

Group	Description	Active	Action
Administrator	The Administrator role is mainly to manage the groups and users within the voting system. He wont be able to participate and vote in the issues and project raised.	Yes	View Edit Delete
Domain Expert	Domain Expert is a person who has an expertise in a specific issue category where his expertise value will be 0.8	Yes	View Edit Delete
Team Member	Team Member will mainly be most of the people in the voting system where they may or may not have some knowledge in the issue that is been raised. They will have the least reputation value which is 0.2.	Yes	View Edit Delete
Team Leader	Team Leader are more the senior staff who has more experience than the Team Members. Their reputation value is 0.3	Yes	View Edit Delete
System auditor	System Auditor role is mainly to audit the users access. They can not vote on project, issues or even vote on the issues raised.	Yes	View Edit Delete
Ontology Engineer	Ontologist main role is that they have an expertise on Ontology where they have quite high expertise value which is 0.7	Yes	View Edit Delete

Figure 6-29: List Groups

When the administrator clicks on "View", details of the group will be visible as shown in Figure 6.30. Details of the users within the chosen group, i.e. group's name, description, the group to which they belong and activity status, are also shown. With regard to "Discussion Group", groups that are categorised as "Domain Expert, "Ontology Engineer", "Team Leader" and "Team Member" have "Yes" filled into the "Discussion Group". This is consistent with the previous description of the role and responsibilities. On the other hand, both "Administrator" and "System Auditor" will have "No" on this field as they have only managerial activities. The screen also displays the list of users in the corresponding group. In this example, one member of the Team Leader group is Harry.



Group - Team Leader - Edit

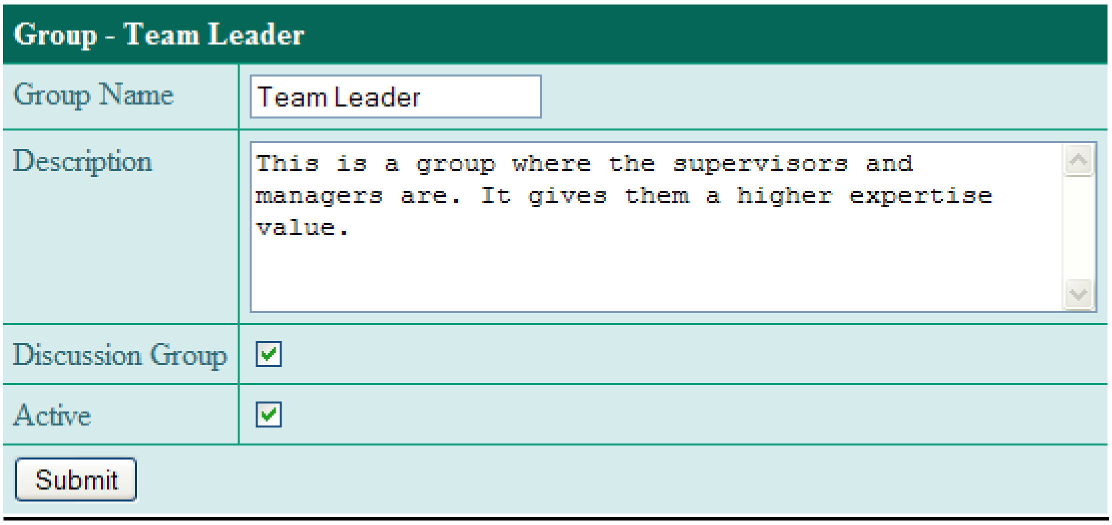
Group Name	Team Leader
Description	this is description
Discussion Group	Yes
Active	Yes

Users in this group

Name	Email	Active	Action
Tinos Cbd	firedrop84@hotmail.com	Yes	View

Figure 6-30: View Group

Once the user has viewed the details of the group, it is possible to edit the details by clicking the "Edit" link. The details to be edited are shown in Figure 6.31. The administrator is authorised to revise the roles and responsibilities of the group, the group to which the discussion group belongs and the status of the group.



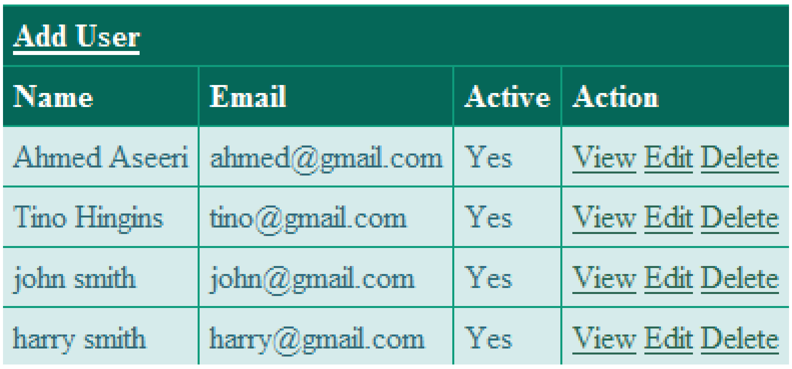
Group - Team Leader

Group Name	Team Leader
Description	This is a group where the supervisors and managers are. It gives them a higher expertise value.
Discussion Group	☑
Active	☑

Figure 6-31: Edit Group

6.3.6 Users

The second feature enables the listing of all the users within the platform. It allows the administrator to add, view, edit and delete a user as shown in Figure 6.32. In this example, another user, Ahmed, is added as the administrator.

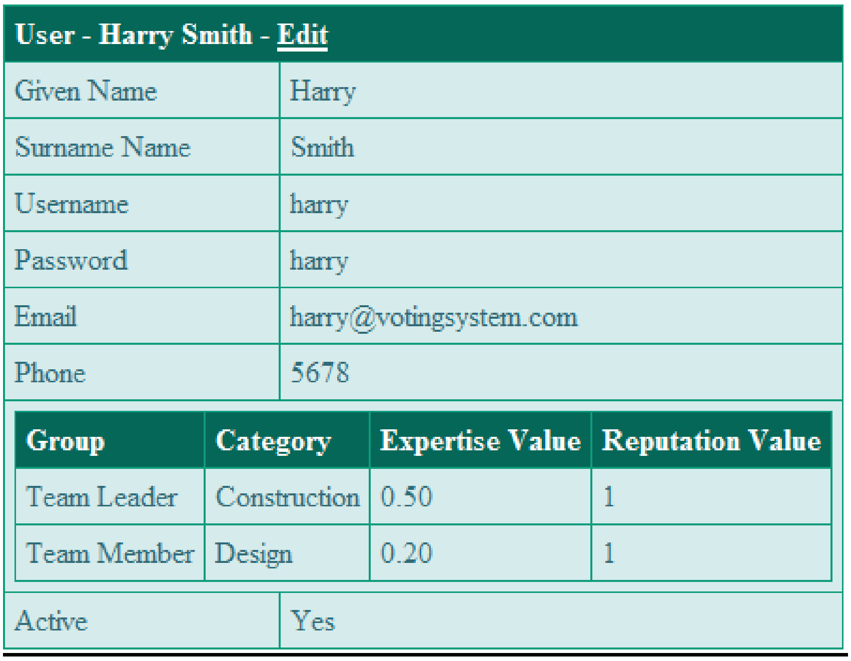


Add User

Name	Email	Active	Action
Ahmed Aseeri	ahmed@gmail.com	Yes	View Edit Delete
Tino Hingins	tino@gmail.com	Yes	View Edit Delete
john smith	john@gmail.com	Yes	View Edit Delete
harry smith	harry@gmail.com	Yes	View Edit Delete

Figure 6-32: List User

The first feature within the List User page enables the viewing of the user's details. The information includes name, a username, a password, email address and contact number. Also, it allows the user to click on the edit link if there are details that are required to be changed as shown in Figure 6.33. On the screen, the name of the group to which the user belongs, his/her expertise value and reputation value are also displayed. These values are extracted using the calculation described in the previous section.



User - Harry Smith - Edit

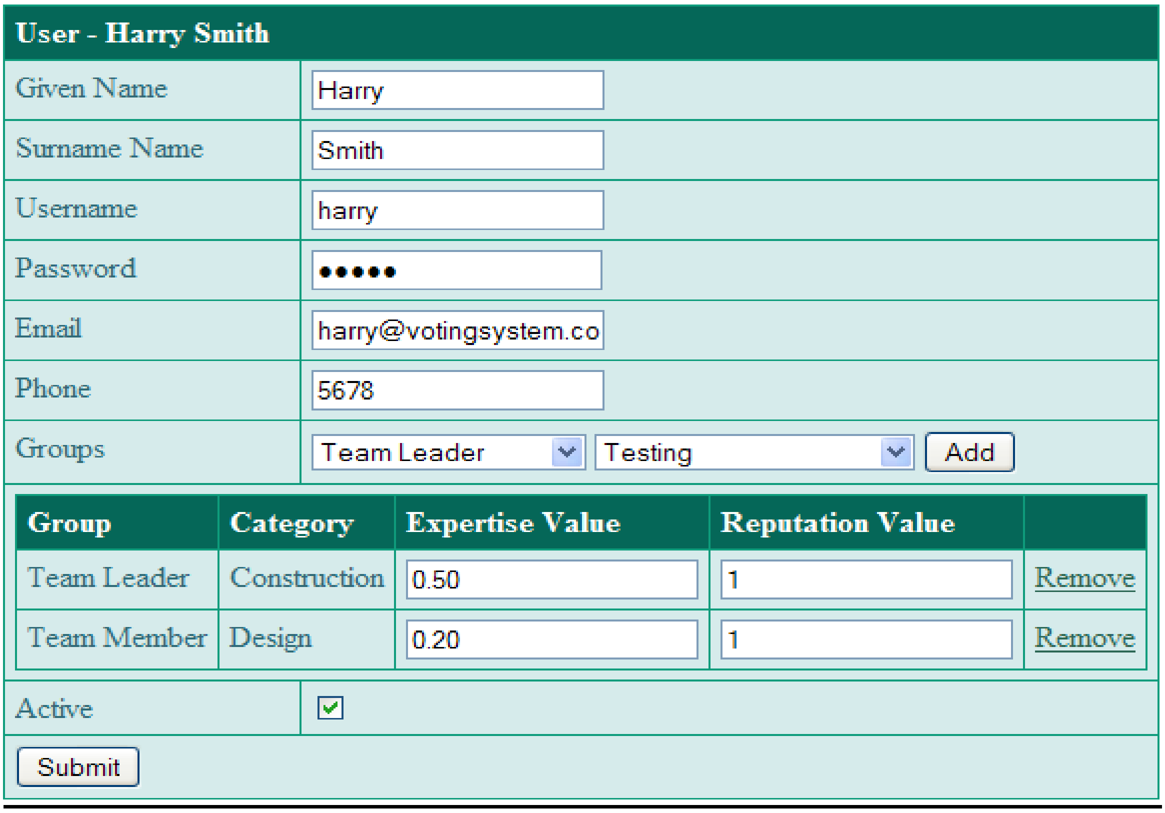
Given Name	Harry
Surname Name	Smith
Username	harry
Password	harry
Email	harry@votingsystem.com
Phone	5678

Group	Category	Expertise Value	Reputation Value
Team Leader	Construction	0.50	1
Team Member	Design	0.20	1

Active	Yes

Figure 6-33: View User

Once the user clicks on Edit, all the details will be editable except for the expertise and reputation values as these are calculated automatically based on the user's voting history and the group to which s/he is assigned. The edit feature is shown in Figure 6.34.



User - Harry Smith

Given Name	Harry
Surname Name	Smith
Username	harry
Password	●●●●●
Email	harry@votingsystem.co
Phone	5678
Groups	Team Leader ▾ Testing ▾ Add

Group	Category	Expertise Value	Reputation Value	
Team Leader	Construction	0.50	1	Remove
Team Member	Design	0.20	1	Remove

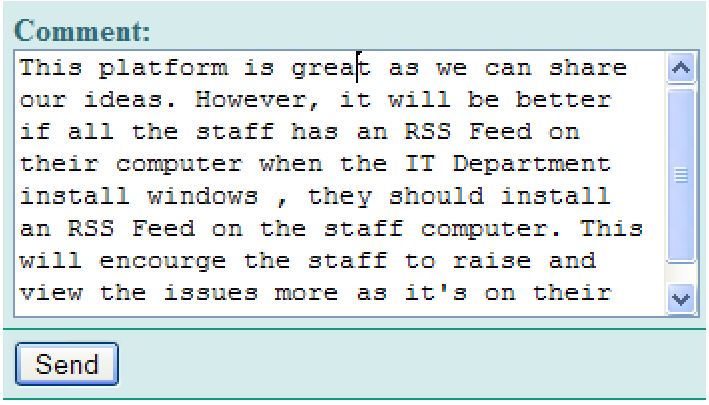
Active	☑

Figure 6-34: Edit User

Another important feature in Figure 6.34 is the "Remove" field. The administrator is responsible for adjusting users' status of membership in accordance with the new raised issue using the "Remove" button. Once the administrator clicks on that particular button, information under fields of "Group", "Expertise Value" and "Reputation Value" are deleted.

6.3.7 Feedback

All the users will be able to provide feedback regarding the platform. This includes ideas on how to improve or add a new feature within the platform as shown in Figure 6.35.



This platform is great as we can share our ideas. However, it will be better if all the staff has an RSS Feed on their computer when the IT Department install windows , they should install an RSS Feed on the staff computer. This will encourge the staff to raise and view the issues more as it's on their

Figure 6-35: Send Feedback

6.4 Discussions

In the previous section, the platform for Lightweight Community Support was presented. As illustrated in Chapter 4, communications among staff from geographically diverse locations are routed through a central hub, which will then order these communications using an ontological hierarchy to determine the significance and value of the information. Individuals and Groups can engage in communications which promote personal, group, and organisational concerns. To ensure the ontological hierarchy remains of highest use to the organisation, this ontology must be managed to keep information current and to designate appropriate use. Management of the ontological hierarchy can be attained through discussions on various ontology evolution issues and the established roles performed by persons active in management roles within appointed Groups. The distribution of responsibility between knowledge management (i.e. Contributor Group) and administration of communication (i.e. Admin Group) has contributed to well-organised discussions on ontology of a particular issue within the oil and gas industry.

Use of this platform will encourage staff to actively initiate and participate in a discussion of issues on a specific topic.Users will be presented with opportunities to acquire knowledge and act upon it in a methodical manner. Users with limited understanding of a particular ontology might benefit from the platform as it will allow them to confirm their comprehension and perception through dynamic dialogue with their peers. The integration of both formal and informal communications will serve as a hub for social networking, which might promote deeper interpersonal discussions and informed decision-making among Contributor Group members. It is also possible that such discussion can be applied by the Admin Group to inform the selection and management of information relevant to the ontological hierarchy which governs communications.

The value of informal and formal social networking cannot be understated, especially in an industry in which participants might be separated by both space (location) and time (zone). The culture and the philosophy of the organisation might be encouraged through allowing Contributor Group members to voice their concerns and receive timely feedback from their administrators or team leaders. Rapid communication between team members and team leaders can also reduce the likelihood of misunderstanding about the ontological framework, and can help reduce conflict among staff members by allowing then to voice discrepancies and concerns. Asstaff have different backgrounds and perceptions on issues, a Lightweight Community Support platform might foster compatibility amongst staff and encourage staff to be responsible for the improvement of ontology. Aside from improved ontology, the platform presents opportunities for staff to enhance their knowledge as well as increase their astuteness regarding particular issues. As staff are encouraged to be involved in domain-based communication in a systematic manner, they acquire the necessary expertise that will be valuable for their subsequent work.

The collaborative discussion which has occurred resulted in a decision that affects the oil and gas ontology which allows it to evolve. The ontology evolution platform provides a way to better understand ontology. By presenting the ontology online using Web-Protégé, staff are exposed and gradually gain knowledge of the ontology.

The obvious problem with the oil and gas industry is related to information predicament between staff located in different areas. The immense distance between staff is obviously immense. There is a need to reduce this distance in order to ensure timely and reliable exchange of information as offered by a platform. This allows staff to conduct discussions virtually. Through this discussion, the ontology of an issue is collaboratively reviewed and improved where needed.

6.5 Conclusion

In this chapter, the lightweight community platform and the ontology evolution platform have been validated and verified. The chapter started with the application of a sample of oil and gas ontology on prototype of platforms that serve Lightweight Community Support and ontology evolution. Furthermore, the ontology evolution platform describes users' involvement in the ontology evolution process and the functionalities of the platform are presented. Finally, the chapter discusses the results and presents a conclusion.

Chapter 7 – Recapitulation and Future Work

7.1 Introduction

Throughout previous chapters, issues related to ontology evolution in a business setting are identified and the solution has been presented in detail. The three key principles of the Lightweight Community-driven Approach are elaborated on in the last three chapters. These include the conceptual framework (Chapter 4), implementation (Chapter 5) and validation and verification (Chapter 6) of the approach. The choice of the oil and gas industry as a case study is substantiated by the industry's distinctive feature of having the business operated from diverse locations situated throughout petroleum-rich regions. The dispersed offices in onshore and offshore branches present a significant challenge to administrators and leaders, who need to ensure that data is transferred between branches and ontology is evolved in a timely and efficient manner. This requires systematic collaboration amongst the relevant staff to allow knowledge captured in ontology to evolve, which is actualised through the ontology evolution platform. The developed platform incorporates the results of knowledge sharing between users from which evolution benefits in terms of reduced errors, cost savings, and increased involvement of users.

In this chapter, research issues and areas are encapsulated. These are the lack of users' involvement in the ontology evolution process and the lack of an ontology evolution platform. Next, the proposed solution which includes the Lightweight Community Support and Ontology Evolution Platform will be presented. The chapter ends by discussing the practical scope of the solution and proposals for future works.

7.2 Issues Faced in Ontology Evolution

Throughout this thesis, the issues faced by the business community regarding ontology evolution have been comprehensively explained. These issues concern the means with which to bring together scattered users who are working in offices located in dispersed areas in order to exchange information. The significance of overcoming the geographical challenge amongst users in different locations emerges from an increasing demand to improve the relevant ontology in a timely and accurate manner. It is apparent that a failure to organise and manage the knowledge captured in ontology will have economic consequences for the business. Within this logical framework, and applying this to oil and gas industry, there are two issues that need to be attended to:

- **Lightweight Community Support**
- **Ontology Evolution Platform**

7.2.1 Lightweight Community Support

A failure to include relevant users in the communication process will reduce the suitability of evolved ontology. Within this context, the focus is on the legitimacy of the ontological structure and what is contained therein, rather than the number of participants. It is important to foster and incorporate feedback from as many future users of the stated ontology as possible to ensure that a broad and diverse range of feedback is incorporated directly into the development and the ongoing maintenance of the ontological system. Taking into account that users have various types of expertise, i.e. technical and managerial, feedback from different users might contribute different perspectives to the ontology.

Historically, the task of updating an ontological system has usually been assigned to the ontology engineer, while the knowledge lies with domain expert. In the oil and gas industry, this traditional approach to ontological communication is not sufficient to meet industry demand. To illustrate this point, it is important to recognize that in the oil and gas industry, all offshore staff are confronted with functional problems and performance issues on a daily basis, while onshore staff's exposure to hands-on complications is limited. Thus, offshore staff have an experiential-based subject matter knowledge. This domain knowledge should be incorporated in order to ensure an updated ontology and to drive an ongoing ontological evolution that incorporates the needs of users. Attainment of this goal must incorporate all stakeholders and participants within the organisation, as a thriving, evolving ontological system emerges from collaboration and a shared willingness to improve communications.

Another issue related to Lightweight Community Support is to provide equal opportunity for users to participate in the ontology evolution process. The significance of presenting all users with options to voice their views relates to the different understanding and discernment that users, in this case offshore and onshore staff, possess. Information exchange among staff is needed not only to upgrade the ontology but more importantly to allow the exchange of a range of ideas and opinions based on which individual and business knowledge management occur. The fast changing working environment – especially on offshore sites – needs to be integrated with information that is available on hand. Without a participatory systematically constructed decision, information will be dispersed and incoherent, which tends to lead to unproductive business activities.

7.2.2 Ontology Evolution Platform

In order to ensure that the evolved ontology is appropriate for users, there is a need to justify the updated ontology. Taking into account the different expertise and knowledge of users, it is essential that this wide range of credentials be acknowledged by the ontology evolution process. Recognition and incorporation of managerial skills as well as domain proficiency are intended to increase the credibility of the updated ontology. Therefore, a platform for ontology evolution should embed the weighted opinions into the actual updating process.

At present, there is no ontology evolution platform available for the oil and gas industry. As discussed previously, offshore and onshore staff have different knowledge and tasks related to their daily operational activities. This division of responsibility between the two sites has impeded the evolution of ontology. The onshore staff are responsible for documentation of ontology including ontology updates. However, these staff are not directly involved with the practicalities of the oil and gas business where the actual drilling and extraction of natural resources are performed. The different roles and responsibilities in the ontology evolution process have posed a challenge in terms of ways to not only integrate the efforts to update ontology, but also to ensure the appropriateness of the result.

7.3 Addressed Research Areas

Issues within this research have been established in the previous section. These issues are consistent with the research areas of:

- **Lightweight Community Support**
- **Ontology Evolution Platform**

7.3.1 Lightweight Community Support

Lightweight Community Support is one research area isolated from the original social network concept. In Lightweight Community Support, participatory discussion and organisation of massive amounts of information are hallmarks of the ontological structure. Within this research, the Lightweight Platform is used to serve the informal nature of the discussion. Participants are able to start a discussion by introducing a topic of their own choice and/or enter another discussion and provide feedback on a particular topic. A unique characteristic of this communication is the autonomous engagement of each member. Participation is not compulsory and it is performed at the members' discretion with regard to time and place. In the end, it presents a collaborative interaction among members regardless of the size of the organisation and the location of its offices.

One type of Lightweight Community-driven platform is the Community Forum. This forum is categorised by topics. Therefore, it meets the need for systematic communication. Within the oil and gas industry, this means each virtual forum is formed every time a specific issue is nominated for further discussion by discussion participants. In addition, knowledge sharing between offices and staff is made possible through Informal Community Support that presents opportunities for offshore staff to be involved in discussions regarding improvement of relevant ontology. This means that Community Forums are also an appropriate tool to engage more people in sharing the knowledge they have on a topic that is of interest to them.

Regardless of their many benefits, Community Forums do not support ontology and lack a problem solving dimension. This ability to produce a decision is vital to this research issue since updated ontology is the desired outcome. In order to overcome this limitation, a Ticketing Support System is introduced. This system enables any staff within the Contributor Group to efficiently raise an issue deemed to be significant for improvement. Once the collaborative communication has been initiated, the platform will automatically manage the process until a reliable decision is reached.

The system proposed in this research acknowledges the expertise and knowledge that have been constructed over the years. Thus, it allows different values to be assigned to participants accordingly, whereby higher recognition is given to the members with higher levels of expertise. The overall method contributes to the appropriateness of the evolved ontology.

7.3.2 Ontology Evolution Platform

The absence of an Ontology Evolution Platform in the oil and gas industry has been established in the previous section. As part of the automation process, tools have been developed to ease the ontology update process; these are commonly known as the Ontology Editor. OilEd, OntoEdit and Protégé are examples of such an editor (Bechhofer, Horrocks et al. 2001; Sure, Erdmann et al. 2002; Gennari, Musen et al. 2003; Tudorache, Vendetti et al. 2008). Web-Protégé is used in this research to perform the ontology revision as the need arises. The justification of Web-Protégé is its user friendly feature. The application is able to mechanically distinguish users according to their preferred role and provide each role with appropriate permission. Thus, users are encouraged to participate consistently in another issue related to their role without reservation. This straightforwardness contributes to a reliable ontology evolution process and outcome. Furthermore, users are also able to view and suggest revisions to the relevant ontology within a given time frame, resulting in a methodical update process.

Besides their practical use in updating ontology, Ontology Editors do not present a stage or venue to discuss the related ontology and do not support a collective effort that leads to ontology evolution. In other words, these applications are useful for performing the actual revision of ontology, but are futile in the preceding process. Taking into account the objective of this research, the process leading to the evolution is very significant. Therefore, it is necessary to have a platform that enables everyone involved to be well-informed about the evolution stages. The symmetric information among users depicts shared knowledge which eventually results in appropriate updated ontology.

7.4 Recapitulation of the Proposed Solution

The previous sections have reviewed research issues and related areas. This section will focus on recapitulating the proposed solution of:

- **Lightweight Community Support**
- **Ontology Evolution Platform**

In addition to this, the validation and verification of the solution is summarized.

7.4.1. Lightweight Community Support

The venue for Lightweight Community Support is instigated from the logical framework provided by the Lightweight Community-driven Approach and is beneficial in reducing the distance gap between dispersed offices, in increasing the involvement of staff and in increasing the level of staff knowledge. The Platform ultimately resolves problems that are acknowledged in the previous section, namely users' disengagement with the ontology evolution process.

Within the platform, staff in scattered offices are able to have a virtual discussion on an agreed issue. The participants are assigned different responsibilities depending on their roles. The roles that have the most impact on the evolution of the respective ontology are those of domain experts, ontology engineer, team leader and team Member, which are assembled in the Contributor Group. The domain experts provide the body of knowledge regarding the issues, while the ontology engineer supports the evolution through his/her distinctive expertise regarding ontology. The team leader and team members augment the content of the discussion with respective working and hands-on experiences.

The proposed System classifies the issues raised according to: the Ontology Element of Classes, Data Properties, Object Properties, Quantifier Restrictions, Has Value Restrictions, Universal Restrictions, Cardinality Restrictions and Individual. Upon its cut-off date, the System begins the calculation of votes that are submitted by participants of the discussion. The different weightings allotted to votes reflect the acknowledgement of technical knowledge and contributes to the result.

Aside from ensuring the reliability of the subject matter, the System also offers a way to undertake a well-organised virtual discussion. The roles that are responsible for an efficient process are administrator and system auditor, which are assembled in the

Admin Group. The administrator is authorised to manage the access of users and the relevant groups to which they belong, while the system auditor maintains the validity of users' authentication granted by the administration. In brief, the system auditor performs the work of an internal auditor to users of the System. Although it has a lesser role regarding the content of the issue raised, the Admin Group does contribute to the quality of the collaborative communication process.

7.4.2 Ontology Evolution Platform

The Ontology Evolution Platform distinctively incorporates a community-driven feature. The feature ensures the systematic involvement of users – a necessary requirement for the updating of the relevant ontology. In addition, the proposed system is designed to assign different weighting to votes cast by users based on users' relevant expertise and knowledge on the specific ontology. This particular Platform ultimately resolves problems that have been acknowledged in the previous section – that is, the need to have shared knowledge amongst employees within the oil and gas industry.

7.4.3 Validation and Verification (V & V) of Proposed Solution

Validation and verification of the Platform for Lightweight Community Support involves the development of a prototype of the proposed system. A small sample of oil and gas ontology is used to extensively illustrate how the platform can be used in order to involve users in the ontology evolution process. The ontology evolution process is evident through the existing of Current, Temporary and Archived Ontology. The Current Ontology is updated by incorporating the results of collaborative discussions amongst staff.

7.5 Scope of Use

This platform can be used by the oil and gas industry in terms of both location and knowledge. With regard to location, the platform has no boundaries. It is operated by offshore staff, and onshore staff who support the offshore staff. This includes offshore personnel who work in very remote areas as well as those who work in an office in the middle of the city. However, with regard to knowledge, the use of the platform is limited to staff with domain knowledge. Although this presents an opportunity for anyone with relevant expertise to participate and voice his/her opinions on any specific issue, staff with general administrative tasks will not be able to participate. Examples of such tasks are those related to human resources (e.g. recruitment) and finance (e.g. accounting).

7.6 Future Work

This research has identified key issues stemming from the updating of reliable data crucial for business operations, as exemplified by the oil and gas industry in this research. The transfer of distinctive and yet complementary knowledge between onshore and offshore staff is pivotal to resolving these issues. An ontology evolution platform has been developed in order to improve the interactions between the staff, thereby increasing staff' involvement, and allowing the ontology to evolve further. However, there are improvements that can be made based on the work completed in this research. Future endeavours should essentially focus on a more organised exchange of knowledge and ideas within this particular business entity that ensure automated ontology evolution. In this section, a proposed solution which can be considered for future work is discussed. This includes: (i) integration with Wiki, and (ii) integration with Web-Protégé.

7.6.1 Integration with Wiki

In this thesis, a platform has been created with scope for enhancing the communication between onshore and offshore staff. It focused on a lightweight communication model that allows the staff to create and add their ideas to the current knowledge base of organisation. However, these new ideas need to be integrated in a more formal way so as to be accessed and assessed in order to be applicable to the organisation where the idea originated. This can be applied by integrating Wiki to allow further control and a formalised way to retrieve the information from the ontology evolution platform. It will be retrieved by obtaining the outcome of the voting options that were discussed by all the contributors.

The formal feature offered by Wiki is emulated from its renowned performance in the field of education. Wikis' significant contributions to enhancing collaborative learning have been well-documented (Novacek, Laera et al. 2007). The discussions cover the various benefits of Wiki, in particular with regard to the construction as well as the evolution of knowledge among its participants. Subsequently, research has been extended to knowledge incorporating collaborators that exist beyond classrooms (Bloehdorn, Haase et al. 2006). These latter studies focused on online and distance education, where the limitations of Wiki were addressed and adjusted to better serve the purpose.

One of the advantages of Wiki technology is its systematic organisation of knowledge. Shared thoughts are categorised by topic, and then followed by sequence. This provides contributors with the opportunity to focus on and follow the progress of the discussion exhaustively. In the context of the oil and gas industry, discussions on several topics can take place simultaneously. Taking into account the dynamic nature of the Domain Expert and the necessity to update information in a timely manner within this particular industry, members of different Domain Expert groups can effectively respond to relevant issues. For example, a Domain Expert who is based in a Western Australia office can provide feedback at the same time as another Domain Expert who is working in Saudi Arabia regardless of the time difference. Using Wiki technology, the feedback will be recorded sequentially, thus allowing other contributors to follow the discussion.

Another advantage in applying Wiki to online collaboration is the existence of a central repository. Although the updating process is carried out in a dispersed fashion with staff located in different offices and regions, the ontology database is developed and maintained as a single server. It is crucial to preserve the historical data of evolving ontology for accuracy and reliability purposes. A shared knowledge recognises the influence of the educational and professional background of each contributor; thus, subjectivity tends to play a role. Systematic documentation of ontology evolved over time will help to overcome any ambiguity in meaning that may occur due to different times and contexts. In addition, it bridges the various levels of knowledge that staff have, which finally leads to the exact intended result.

Semantic Wiki presents additional benefits to the lightweight community-driven approach since it incorporates semantic technologies within Wiki applications by way of annotations. This improves the utility of the proposed system since users are not bounded by technical prerequisites. Members of Domain Experts are knowledgeable and qualified people in their domain, but they are most likely lacking in technical expertise; thus, a user-friendly system is beneficial for them. In other words, Semantic Wiki provides a platform whereby members with variable technical ability can engage in collaborative discussion.

7.6.2 Integration with Web-Protégé

The development of this thesis has focused on creating a separate platform to allow staff to raise, discuss and vote on particular issues. The proposed system does not require all staff to have some knowledge of ontology, even though the end product of this collaborative communication is an evolved ontology. This means that staff with domain knowledge and expertise can engage in discussion on particular issues on one platform, while the revision (if any) of ontology will be executed by Ontology Engineers on another platform, based on the result of the discussion,.

However, this research has not examined the integration of the platform with the ontology editor, Web-Protégé. Such incorporation will allow a business entity to have a single system that will be used by all staff. In recent years, a research team at Stanford University (Tudorache, Vendetti et al. 2008) has been developing a Collaborative Protégé, which is intended to assist the process of ontology evolution carried out by a group working in the biomedical field. Although it serves a different domain than the industry focused on in this thesis, some features are applicable to the oil and gas industry.

The Web-Protégé is currently in the development stage; hence, there are still bugs within the Web-Protégé platform that makes the communication not as reliable as it should be. Furthermore, Web-Protégé does not provide an effective voting option which helps the users to make decisions about issues and allow the ontology to evolve. The basic concept of Collaborative Protégé is based on the notion that it takes a collective effort to improve ontologies, through annotating and voting mechanisms. The developed system enables multiple users to work on the same ontology simultaneously, and multiple users can work on the same ontology consecutively. The former is known as the multi-user mode, while the latter is called the standalone mode. The Stanford team has the development of Web-Protégé as their next endeavour, which started in 2009, utilising some features of Collaborative Protégé. This will allow ontology to be evolved openly without requiring specific software.

The integration of the platform with Web-Protégé presents the business with both monetary and non-monetary advantages. An entity can maintain the integrated system with fewer staff and less time, thereby enabling a cost-effective system to be put in place. This ease of maintenance also means that more time can be devoted to actually improving the system for the future needs of the organisation, rather than providing support to it. The organisation may require specific features that are currently not available but can be appended to the system in future.

Regardless of the aforementioned advantages, the integration requires one significant precondition that has to be satisfied. All participants in the collaborative communication have to have a certain level of knowledge of ontology. Without this minimum level of understanding, the system will not work effectively. Although the currently developed Web-Protégé provides a solution for this by creating typed comments, this presents a drawback with regards to sharing knowledge. Participants are bounded to predefined comments rather than being able to express their individual opinions and ideas.

In order to address the above shortcoming, in particular within the oil and gas industry, the system proposed in this thesis has allowed the Ontology Engineer to be a part of the discussion process. Nevertheless, in the future, the industry has to devote time and effort to ensure that staff – at least those who are at one point or another included in Domain Expert – have some knowledge of ontology. Capital investment in this area will benefit the organisation in the long term as established in previous chapters.

Furthermore, this also can be considered by developing a plug-in within Web-Protégé. However, due to the immaturity of Web-Protégé (alpha phase) at the time of writing this thesis, the development of a plug-in or integration with the platform is not feasible at the present time. Therefore, this can be considered as a future direction to further develop and enhance the process.

7.7 Conclusion

This research has constructed a platform which has the fundamental function of enhancing the users’ involvement in the ontology evolution process. Using the oil and gas industry as a case in point, the issue is to ensure that onshore and offshore staff interact with each other to update the ontology. This includes encouraging them to provide ideas or raise any issue identified in the course of their daily work. Further, the platform helps to evolve the ontology further, for this leads to a reduction in the operational error costs within an organisation and an increase in financial benefits.

7.8 References

Bechhofer, S., I. Horrocks, et al. (2001). OilEd: A Reason-able Ontology Editor for the Semantic Web. KI 2001, Vienna, Austria

Bloehdorn, S., P. Haase, et al. (2006). Ontology Evolution, John Wiley & Sons.

Gennari, J. H., M. A. Musen, et al. (2003). "The evolution of Protégé : an environment for knowledge-based systems development". International Journal of Human-Computer Studies **58**(2003): 89 - 123.

Novacek, V., L. Laera, et al. (2007). Semi-automatic integration of learned ontologies into a collaborative framework. IWOD.

Sure, Y., M. Erdmann, et al. (2002). OntoEdit: Collaborative Ontology Development for the Semantic Web. International Semantic Web Conference 2002 (ISWC'02), Sardinia, Italy

Tudorache, T., J. Vendetti, et al. (2008). Web-Protégé: A Lightweight OWL Ontology Editor for the Web. OWL: Experiences and Direction (OWLED) Workshop, Karslruhe, Germany.

Every reasonable effort has been made to acknowledge the owners of copyright material. I would be pleased to hear form any copyright owner who has omitted or incorrectly acknowledged.

APPENDIX - (Conference Papers)

Towards Social Network based Ontology Evolution Wiki for an Ontology Evolution

Ahmed Aseeri, Pornpit Wongthongtham, Chen Wu, Farookh K Hussain
Curtin University of Technology
GPO Box U1987 Perth WA 6845 Australia
{A.Aseeri, P.Wongthongtham, C.Wu, F.Hussain}@cbs.curtin.edu.au

ABSTRACT

There is a lack of well-maintained ontologies thus ontology evolution now becomes an important filed of ontology research. The evolution may reflect new categories of systems being evaluated on broader and different understandings of certain concepts and relations. Alternatively ontologies evolve because the conceptualization improves. For ontology evolution, we focus in this paper a social network based approach in which the user community has direct control over the evolution of the ontologies. Ontologies can be enriched, learnt, and obtained from social network users using various empirical techniques. In this paper, we ground the social network based approach on the philosophy of wikis so called ontology Evolution Wiki.

1. INTRODUCTION

There is a lack of well-maintained ontologies thus ontology evolution now becomes an important field of ontology research. Ontologies inevitably evolve over time. The evolution may reflect new categories of systems being evaluated on broader and different understandings of certain concepts and relations. Alternatively ontologies evolve because the conceptualization improves. For ontology evolution, we focus, in this paper, on a social network based approach in which the user community has direct control over the evolution of the ontologies. Ontologies are developed for community thus the community shall have control over it. Ontologies can be enriched, learnt, and obtained from social network users using various empirical techniques. In this paper, we ground the social network based approach on the philosophy of wikis so called ontology Evolution Wiki. In rapidly evolving domains the ontology Evolution Wiki is important because the community users can keep up with the pace of changes. Ontologies are used to reach consensus on a view of the world and agreed upon by the community users. Due to ontologies used by the community users, they play an important role in the ontology evolution. Maintaining the ontologies is then a joint effort by the community users benefiting from the ontologies.

Once reached to the ontology Refined Wiki which is the version all agree upon, knowledge and agreement are explicit and shared not only among members, users, and domain experts but also among software systems. Human as well as machine (i.e. in the form of software applications) can make use of the explicitly knowledge and agreement. At this stage developing and evolving ontologies are tasks that depend on human intelligence as a source of community users and domain expertise in producing a consensual conceptualization and resolving inconsistencies. The developed ontologies can be deployed by both human and software agents.

In the next section, we review chosen social network techniques including media wiki and its extension, semantic media wiki. In section 3, we present the ontologies and their evolution. In section 4, we discuss our approach, ontology Evolution Wiki and conclude the work in section 5.

2. LITERATURE REVIEW

MediaWiki [1] is the engine for Wikis that have been developed so that everyone can collaborate in order to achieve certain goals. More recently, Wikis have been widely used in many organizations and institutions as a way to communicate, share, or explain a specific topic which made it as a knowledge base for everyone.

An application of MediaWiki is the Wikipedia [2] which has been developed in 2001 and made the Wiki concept much more widely used around the universe. Also, there has been a development of the Swiss Experiment – Tagging within Wiki [3] which is an extension for wiki that support personalized access to documents and specialized user group. Also, there has been other Wiki's that has been developed such as Platypus Wiki [4], OntoWiki [5].

Semantic MediaWiki [6] is an extension for MediaWiki which uses the Semantic Web [7] concept on Wiki which allows the computers to understand the content in Wikis instead of only displaying them for human interaction.

All the media Wikis and the semantic media wikis that have been developed thus far were using the new features that it provides. However, none of the Semantic Wikis are aimed at developing a Semantic MediaWiki that allows the users and experts to discuss the ontology in an Evolution Wiki for a specific domain and allow only the experts to modify in a different Wiki (Refined Wiki) after the discussion has been completed.

iiWAS2008, November 24–26, 2008, Linz, Austria.
(c) 2008 ACM 978-1-60558-349-5/08/0011 $5.00.

3. ONTOLOGIES

Ontology is a widely accepted, state-of-the-art way for knowledge representation. The ontology term can be referred to a wide range of formal representations to detailed logical specification of a domain. Basically its details depends ontologists who describe a domain and on a number of factors, for example, the domain itself, its uses, etc. Ontologies are used in many industrial and academic applications e.g. concept-based search, interoperability support, constraint specification, semantic web applications, etc. [8].

Formally, ontologies are formal, explicit specifications of a shared conceptualization of a domain [9][10][11] with following properties:

- machine-process-able semantics
- explicitly defined
- consensual knowledge
- abstract model

Ontology elements are concepts or ontology classes, relationships hold among concepts, constraints or restrictions, and instances. Changes in these elements are inevitable. Changes to any of those elements can cause changes in the ontologies. Different things may imply different views on the domain and consequently a different conceptualization [12]. The evolution may reflect new categories of systems being evaluated on boarder and different understandings of certain concepts and relationships. Alternatively ontologies evolve because the conceptualization improves.

4. ONTOLOGY EVOLUTION WIKI

Ontologies inevitably change over time [12]. Wiki is one of the solutions that allow collaborations amongst different people to reach an agreement. Ontology Evolution Wiki is then a solution to control, track, and trace the changes in which different users are involved in the changes in a form of either a proposal submission or proposal discussion. Figure 1 shows an overview of the systems architecture. There are three types or levels of users to give control over the systems i.e. general users, community members, and domain experts. Each one will have different privileges and roles in the ontology Evolution Wiki. This will help to further control the ontology Refined Wiki, the version everyone agree upon, and will still give the members and experts the ability to discuss and raise issues that they believe are required to be evolved. After all, the general users or even software agents are able to access and use the refined ontology.

The ontology Evolution Wiki is where the members and experts propose the changes through proposal submission and discussion. The ontology Refined Wiki is the version that the ontology Evolution Wiki get updated / modified. The ontology Refined Wiki will be modified by the experts after brainstorming and discussion in the ontology Evolution Wiki. By having two separate wikis, it will give a further control for the domain. Additionally, it will provide consensus information that everyone agree upon.

As from Figure 1, the experts and the members import the ontology file (an OWL file) to choose the domain to work on. The chosen ontology will be shown in a tree-like structure of concepts (concepts hierarchy) as well as relationships hold concepts together and constraints. This will give a clear understanding for the members and experts of how the domain is described and how the concepts are related to each other. If there is any issue over them, the members and the experts then raise the issue through proposal submission. Once the proposal has been submitted, the experts and the members will start to discuss over the submitted proposal to reach to a point of agreement. After automatic text analysis and pre-processing, these change requests will be transferred to the ontology Refined Wiki. If changes are required, the expert will be responsible to update within the ontology Refined Wiki. Everyone can then follow the new update in the ontology Refined Wiki if these changes are finalized. Eventually, the changes will be propagated to the Ontology Evolution wiki through the refreshing mechanism.

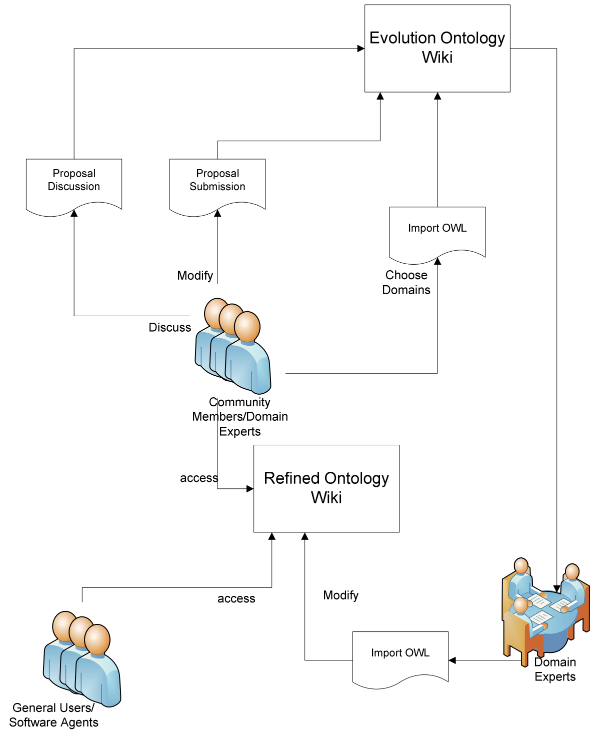


Figure 1. Overview of systems architecture

5. ONTOLOGY EVOLUTION WIKI IMPLEMENTATION

MediaWiki, Semantic MediaWiki and TreeAndMenu are essential to set up the environment for ontology Evolution Wiki. Both Semantic MediaWiki and TreeAndMenu are extensions for MediaWiki. Ontology Evolution Wiki is a plug-in allowing user to install it as an extension to the Semantic MediaWiki. The ontology Evolution Wiki displays the ontology into a tree-like structure as shown in Figure 2.

Figure 2 illustrates the Evolution Wiki for the software engineering ontology [13] as an example. The user is able to click on the concept to inspect relationships, related concepts, properties, constraints, instances, etc. Related concepts will be displayed in other pages in more details. Ontology Evolution

Wiki allows the users to view the tree-like structure and display each concept details and its associated properties.

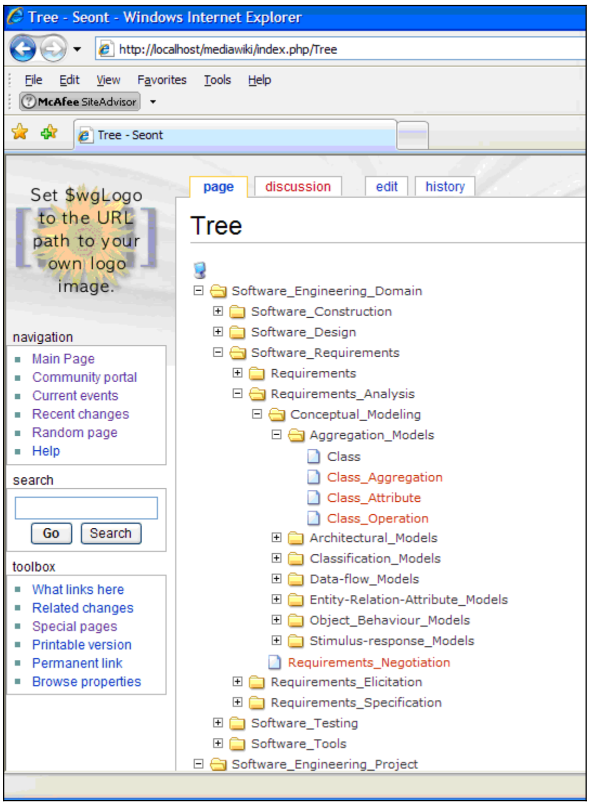


Figure 2, Tree-like structure of the software engineering ontology

6. CONCLUSION AND FUTURE WORK

We have proposed a social network-based approach towards ontology Evolution Wiki for an ontology evolution. This is a step forward in the ontology research. There will be considerable amount of improvement for the future work. In our future work, we aim to have a methodology and a prototype for community ground evolution of ontologies. We also aim to develop key functionalities that software agents can involve in ontology evolution on behalf of human agent. This will combine human and machinery power to support users in achieving ontology evolution.

7. REFERENCES

[1] MediaWiki. MediaWiki Home Page (October 14 2008): http://www.mediawiki.org/wiki/MediaWiki

[2] Wikipedia. Wikipedia Home Page.(October 14 2008) http://www.wikipedia.org

[3] Singh, A., Wombacher, A., Aberer, K, 2007, Personalized Information Access in a Wiki Using Structured Tagging, On the Move to Meangful Internet Systems 2007: OTM 2007 Workshop.

[4] Wang, C., Zhang, G., Zeng, X, 2007, Creating and Managing Ontology Data on the Web: A Semantic Wiki Approach, Web Information Systems Engineering – WISE 2007.

[5] Soren, A., Sebastian, D., Thomas, R. 2006. OntoWiki – A Tool for Social, Semantic Collaboration. International Semantic Web Conference.

[6] Semantic MediaWiki: Semantic MediaWiki Homepage (October 14 2008) http://www.semantic-mediawiki.org

[7] Celino, I., Della Valle, E., 2005, Multiple Vehicles for a Semantic Navigation Across Hyper-environments, European Semantic Web Conference.

[8] McGuiness, D.L. 2001, Ontologies Come of Age. The Semantic Web: Why, What, and How. D. Fensel, J. Hendler, H. Lieberman and W. Wahlster, editors, MIT Press.

[9] Gruber, T.R, 1993, translation approach to portable ontology specification, Knowledge Acquisition.

[10] Borst, W, 1997, Construction of Engineering Ontologies, Centre of Telematica and Information Technology, University of Tweenty, Enschede, The Netherlands.

[11] Studer, R., VR, B., Fensel, D, 1998, Knowledge Engineering: Principles and Methods, IEEE Transactions on Data and Knowledge Engineering.

[12] Noy, N., Klein, M, 2004, Ontology Evolution: Not the Same as Schema Evolution, Knowledge and Information Systems.

[13] Wongthongtham, P. 2006. A methodology for multi-site software distributed software development. PhD thesis Curtin University of Technology

Community-Driven Ontology Evolution Based on Lightweight Social Networking in Oil and Gas Domain

Ahmed Aseeri
Digital Ecosystems and Business Intelligence
Curtin University
Perth, Australia
ahmed@aseeri.com

Dr Pornpit Wongthongtham
Digital Ecosystems and Business Intelligence Institute
Curtin University
Perth, Australia
P.Wongthongtham@cbs.curtin.edu.au

Abstract— **Ontology evolution is one of the problems facing ontology users today. One of the problems is that making changes to one concept of ontology might affect other concepts, thereby leading to inconsistencies. Moreover, this creates a lot of work for ontology engineers who have to update the ontology manually. The process of changing the ontology requires work from the domain experts. No cost effective communication method exists that enables discussion and agreement on these changes. In this paper, we propose to use a lightweight Social Networking approach in order to enhance the communication between different participants. This will produce a more efficient and effective method of communication and decision-making.**

Keywords-component; Ontology Evolution, Social Network, Web-Protége, Oil and Gas Ontology

I. Introduction

Communication between staff within the Oil and Gas industry, whose sites are in different geographical locations, is crucial and at the same time problematic. There is a need to have information on issues regarding day-to-day activities in a timely manner which will affect productivity of the company. In addition, the different operation sites, such as offices both onshore and offshore or located in different countries, produce other concerns related to delayed transfer of information due to different time zones. Within this context, the updating of Oil and Gas Ontology will increase productivity and reduce operational errors and maintenance costs [1].

Staff members may use the Oil and Gas Ontology, but they are not involved in the development process. Most existing ontologies including the Oil ontology are designed by individuals or a small group of experts, not actual ontology users. An effective means of improving existing ontology is to secure the active involvement of relevant personnel during the process of creating and maintaining ontology. Thus, we raised the question of how and what will be the most effective and efficient way to improve communications between different users to address the aforementioned need.

In this paper, the Lightweight Community Driven Approach is proposed to provide a platform to enhance communication between different participants. This systematic discussion will produce a more reliable outcome since every staff member is given an opportunity to be involved in the knowledge sharing process by providing feedback on a particular issue. The platform incorporates a voting system on which the updated ontology is based. Therefore, this research provides a tool for creating a new opportunity to discuss issues and allow the ontology to evolve.

The Lightweight Community Driven Approach supports the company in three ways. First, it reduces the distance gap between the staff by connecting them virtually. Second, the approach increases the commitment and responsiveness of staff which leads to higher quality of communication. Third, collaborative discussion encourages the learning process amongst staff throughout the ontology updating process. Accordingly, the contribution of the proposed system is twofold: (1) to present an approach that enhances the collaborative communication between different participants within the oil and gas industry; and (2) to present a platform that allows a greater number of staff to participate in the evolution of oil and gas ontology so that staff is more aware of changes and more actively involved in the change process.

The rest of this paper is organized as follows. In Section 2, we discuss the existing work available regarding this field. We present the conceptual framework applied to this research in Section 3 and implementation of our proposed framework in Section 4. We present and discuss our findings in Section 5. In Section 6, we conclude our research and make recommendations for future work.

II. Related Work

Most of the research undertaken previously was concerned with developing plug-ins or extensions for Ontology Evolution such as Protégé. Other researchers have developed mathematical formulas to support their ontology evolution theory or use a visualization presentation. These theories are presented and described below:

ISBN: 978-1-4577-0872-5 (c) 2011 IEEE

A. *Protégé Plug-Ins*

Protégé Plug-ins have been developed where each plug-in has been designed for a specific purpose. This includes plug-ins which were developed specifically for Ontology Evolution. These are described in the following:

The Change Management plug-in provides many functions to Protégé. Firstly, it records the changes as an instance in the Change and Annotation Ontology (CHAO) [2] with the timestamp and author of the change. Therefore, once the plug-in is enabled in Protégé, all the changes are monitored for any change. Furthermore, users can gain access to CHAO either in Protégé or the Protégé knowledge-base, API. Secondly, the change management plug-in provides two views – a detailed view and a summary view. The detailed view shows the low-level changes to each ontology, while the summary view shows the changes as groups [3].

PROMPT Plug-in provides some of the ontology management functionalities, including four main functionalities: firstly, it compares the current ontology with a different ontology; secondly, frames can shift between the current project and the included project; thirdly, ontologies can be merged; and finally, a portion of ontology can be extracted and added to the current project [4]. The comparing section is the most relevant in the context of ontology evolution since it checks whether or not the instance of CHAO exists, and follows different methods depending on the finding.

Evolva starts with information discovery, data validation, ontological changes, evolution validation, and ends with evolution management. It starts by comparing the information from the external domain and validates it through a set of heuristic rules, then using a gradual matching technique that starts from the simplest and quickest and proceeds to the more complex and time consuming. If no relations are found, it could possibly ask for a manual check. The final stage consists of evolution validation and management where a consistency and duplication check is done and records for any tracing or roll backs are obtained [5].

B. *Web-Protégé*

Web-Protégé is one of the protégé clients used to interact with the Protégé Server and supports the evolution process of ontology. Web-Protégé is a collaborative platform to allow users to update and view instances and concepts. It is also able to segregate users' privileges by providing them with an appropriate permission. This will also help other users to see and track the changes immediately [6].

Web-Protégé contains a server and client side. The server provides access through Ontology API. Collaboration API provides a method for the users to comment on or change a specific ontology. Also, it manages the conflict if different clients make a change to the same ontology. On the other hand, the client is the user interface where the user interacts with the server. It contains Remote Procedure Calls (RPC), which is a module for communication with the server. However, the current version of Web-Protégé has some issues and further development of the next version is intended to overcome these issues. This includes the lack of support for OWL 2.0 with the current infrastructure, enforcing access policies for browsing and editing ontology and the scalability, as a large ontology might be an issue with the limitations of Java Script [6].

C. *Ontology Evolution in a Visual Representation*

Visual Presentation has been one of the approaches for presenting ontology evolution. This includes developing a plug-in for Protégé or other applications that support this process. Moreover, a list of requirements has been identified to visualize changes in ontology evolution [7]. However, visual presentation is a challenging task given its expressive nature and the series of graphs produced over time [7].

Media is one of the domains that changes very frequently and therefore requires an instant update of the instances in ontology. Hence, ontology evolution is a key factor in media, and visualizing the changes will help further the ontology evolution process. Several studies have been undertaken with a sample of six month's data in order to visualize and semi-automate ontology evolution [8]. These studies have also classified the ontology into three terminologies, beginning with core to extended domains and ending with the peripheral. The three different types of relationships, and how the same topic can change from one terminology or relationship to another when the topic's priority changes, are then discussed.

D. *Oil and Gas Ontology*

The use of Oil and Gas Ontology is becoming more popular in these industries. Statoil in Norway is one of the Oil and Gas industries that has been investigating the benefits of using Oil and Gas Ontology for the AKSIO project [9] and Norwegian Continental Shelf [1].

The AKSIO project is concerned with developing a knowledge management system to support operations in offshore oil fields. Major operators, such as Statoil, have about 20 drilling projects around the world. This has created issues regarding monitoring, analysing or making decisions between projects which cost the drilling process downtime around $0.5 – $1 million/day. Therefore, it is vital to have a process for knowledge transfer between drilling projects through documented experience best practices and expert references [9].

The AKSIO system has two main functionalities - ontology-based annotation and contextual ontology-driven retrieval of content. These functionalities are created around two use cases. First, capture and qualify knowledge gained in the drilling operation; and second, supply relevant and timely knowledge to the planning of new wells [9]. These use cases have resulted from issues arising from the current approach.

The demand for oil and gas has increased in the last decade. This has led to the need to investigate a more efficient way of drilling and completion to reduce the pressure and protect the environment further. Therefore, an Integrated Operation project on the Norwegian Continental Shelf has been researched and implement within Norwegian Oil and Gas industries (e.g. Statoil). This is to further "support operational decisions about offshore installations by onshore control centre, developing common standards, integrated solutions, and new technologies" [1].

Integrated Operation (IO) consists of two generations: "(IO G1) is to integrate processes and a person offshore and onshore to improve onshore's ability to support offshore operations." [1]. IO G2 helps operators to utilize the vendors' competences and services more efficiently than today" [1]. IO G2 is harder to implement as it requires a high level technology, which includes the development of an Oil and Gas Ontology (OGO) [1].

Offshore Oil and Gas industries have massive data which has been gathered over decades. However, these data can be used in a more organized way to increase the productivity and further protect the environment during Drilling and Completion process. Therefore, the availability of real-time data would be a valuable asset within these industries [1].

Hence, an automated analysis and response as well as prediction of the near future via machines will provide great benefits for the drilling team. This can be implemented by using Semantic Web, which requires an Oil and Gas Ontology. It will further enable the transference of real-time data from offshore to onshore and vice versa, and will automate key work processes [1].

E. Social Network

Social Networks have created a new world where participating users share their knowledge with others. Furthermore, most of the top websites on the internet are Social Network websites or have an embedded Social Network feature [10]. This includes new technologies such as Wiki, Blog, and community forums. It has also helped to provide massive levels of information for the community to draw upon. However, because of the growth of information ontology, and the semantic web becoming a key factor in organizing and analyzing the information into a more useful structure and advancing the automated decision making process, combining these technologies can open new doors for innovation and create an intelligent way to solve some issues faced by ontology evolution. In this paper, three types of Social Networking are discussed in detail as it is the most related to this research.

Wiki is one of the technologies from which Ontology Evolution can benefit in order to evolve ontology more efficiently. This can also be combined with folksonomy as another tool to tag on to the data in order to provide comments and suggestions. Moreover, Wiki allows users to add, delete, and modify ontology based on discussions occurring within wiki. This provides an open environment for the particular community to discuss the change in ontology. Furthermore, the user will have an opportunity to edit the ontology more freely which will further improve the ontology maintenance process. This will also provide an opportunity for a voting system to involve the ontology evolution and to use Wiki as a collaborative approach discussing the changes in ontology [11]. Thus, Wiki can be a valuable tool for enhancing the ontology evolution process by means of collaborative editing.

Ontology Evolution also faces other challenges. This includes the formality of specifying ontologies, the difficulties of the world understanding the meaning of concepts and relations, and the fact that most of the ontologies are constructed by a group of engineers which means that users have no direct control over the evolution of the ontology. Therefore, Wiki can be used as a portal in order to involve different users in the ontology evolution process [12]. However, tracking the changes of ontology evolution can be beneficial in Wiki in order to analyze what has been changed. since it allows users to undo and examine previous versions of the ontology [13].

Weblogging has become a popular tool in the last few years within the cyber world. Weblogging users can be categorized into different groups: habitual, active, personal and blogging lurkers. This ranges from very addicted users to the least addicted users respectively [10].

A Single Weblog can have many features. Firstly, each blog has its own permanent link, "permalink", which users can refer to when they try to access a blog. This can be either an individual webpage or a section in a webpage. Blogspace is where users communicate [14]. It is also a type of repository that represents the knowledge by sharing it within online hyperlinks. Within the Blogspace, the blogger reads and refers to topics on the sidebar, which are referred to as Blogrolls. They can also comment or tag each others' postings and build further relationships based on their common interests [15]

Some applications such as Wordpress, Blooger, and Egblog provide further opportunities and encouragement for users to create and manage their own blog more easily. Also, numerous plug-ins have been developed for users to choose from for these applications. This gives users the choice of selecting any extra feature they prefer to have in their blog. It has also created more scope for innovative new ideas which can be shared with other users.

An Online Forum is a virtual communication that starts with a thread containing information, questions or requests. It is then followed by further replies to the message. These messages are usually sorted in descending order. Each community forum specializes in discussing one area and is established for different purposes. This includes information sharing, coordination, and emotional support. A forum can be either public or private. The public forums can usually be accessed by anyone. A private forum can be created and accessed within an organization's internal network [16]. Users can be linked by different roles which may be administrator, moderator, or member with reading and/or writing permission [17].

Community forums usually target specific types of people who have common interests. However, these communities exist on the boundaries of their interests of the sub-category. This starts with the common crossposts phenomenon. Crossposts are posts or articles that the user thinks might be of interest to another community or sub-group. Therefore, traditional social network analysis (SNA) will fail to identify that the person knows the other person because they are invisible to each other. Thus, in order to find a solution which allows people to link with one another via their common interests, additional network structures are required. This includes the domain specific knowledge of the user and, by using ontology, a link can be established between people with similar interest in a particular topic [18].

III. Conceptual Framework

Developing an Ontology Evolution Platform for Oil and Gas industries is vital. In this section, a Community-Driven Ontology Evolution Platform for the Oil and Gas Domain is presented. The platform is divided into three layers, as described below.

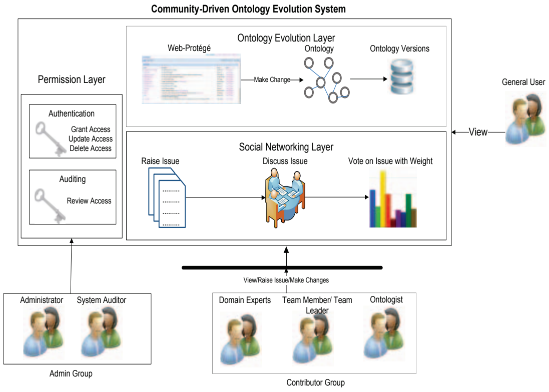


Figure 1: Conceptual Framework of the Ontology Evolution Platform

As shown in Figure 1 below, the platform includes Team Members/Leaders, Domain Experts, Ontologist, Administrator, and System Auditor. These users are divided into the different groups: the Admin Group and the Contributor Group. The Admin Group includes the Administrator and System Auditor. The Contributor Group consists of Domain Experts, an Ontologist, Team Members and Team Leaders. This will firmly establish the separate role of each user.

Furthermore, the Platform is divided into three layers: (i) Social Networking Layer, (ii) Permission Layer, and (iii) Ontology Evolution Layer. The Social Networking Layer is where the Domain Experts, Ontologist and Team Member/Leader are able to raise, discuss and vote on an issue. The Permission Layer allows the Administrator to manage the access of the users and groups. Finally, the Ontology Evolution Layer allows all the users to view the ontology by using Web-Protégé [6].

IV. Implmentation For Ontology Evolution Platform For Oil And Gas Domain

As explained briefly above, two groups are involved in the ontology evolution process, namely the Contributor Group and the Administrator Group. Members of the Contributor Group are those who are actively participating in the discussion by raising an issue, providing feedback, and voting on an issue. This group is responsible for the quality of the discussion outcome which includes reliability of the ontology. The Admin Group is tasked with the managerial aspects of discussion to ensure the eligibility of every participant.

Any staff member is able to raise an issue that s/he finds in Current Ontology. Current Ontology is the existing ontology that is used by employees in a particular oil and gas company. For example, the Team Leader proposed that "hasExhausted" be added to the Current Ontology. The Team Leader recommends a modification of the existing ontology to accommodate the possibility of an oil field being exhausted, i.e. there is no oil available for extraction.

Once an issue is raised, a deadline is set in order to limit the discussion process. While the discussion takes place, other participants are encouraged to provide feedback and comments on the proposed ontology. Every staff member participating in the discussion is free to ask the Ontologist to create a Temporary Ontology based on his/her proposed ontology.

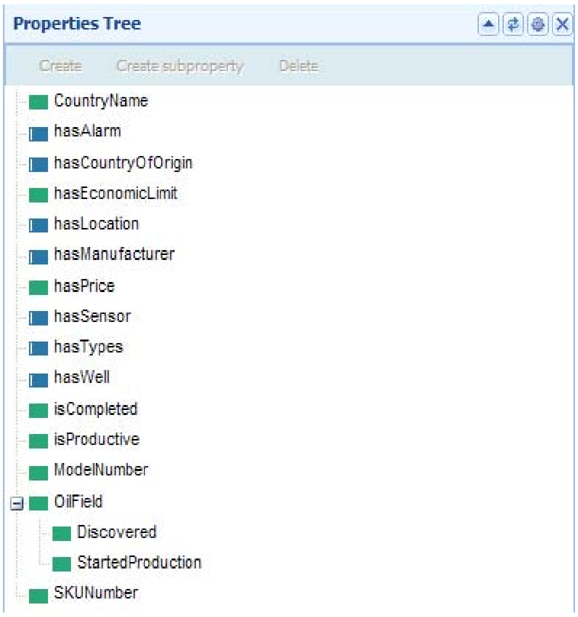
Each member of the Contributor Group then votes on the ontology that s/he considers is the most appropriate for the issue. The platform establishes a policy of one man, one vote and single entry mode. This means that once a vote is cast, it cannot be revised, even though the issue is still open to discussion. Thus, voters have to be confident about their decisions before they opt for any vote.

There are two values which this platform takes into account to weight the submitted vote: expertise and reputation value. The expertise value is assigned in a way that reflects users' knowledge. Users with higher value are those who have greater experience and/or knowledge. This starts with the highest expertise value for Domain Expert (0.8) followed by Ontologist (0.7), Team Leader (0.5) and Team Member (0.2).

On the other hand, the reputation value is calculated based on past reputation points which the users have accumulated on different projects that resulted in 1 or 2. This value will increase/decrease based on the performance of the votes. The reputation value always starts with the value of 1. This encourages the staff to participate in order to increase their reputation value in accordance with the category of the issues.

The staff vote on one of three options which are: (i) Unresolved Issue, (ii) Viewpoint Issue and (iii) Modify Issue. Unresolved issue occurs when the users do not find any solutions to the raised issue. Within this context, participants in the discussion concur that the issue with the ontology remains. However, the voters have not come to an agreement about the actions that should be applied to the ontology. When an issue is classified as a Viewpoint issue, this means that the ontology should remain the same. The last option, Modify issue, means that active action such as revision to the ontology is demanded. Below is an illustration of the collaborative communication that leads to ontology evolution. Supposedly, the Team Leader finds that one of the many oil fields the company operates, Burgan oil field, has been experiencing decreasing oil productivity as much as 30% within the last five years. This alarming information is crucial since Burgan oil field is considered as one of the biggest oil fields in the world and one in which the company has a lot at stake. From the business perspective, the company has two options, either to allocate substantial resources dedicated to renewal program of this oil field or to develop an exit strategy. If the company chooses to carry on with the latter option, the future situation of an exhausted oil field has to be included in the ontology.

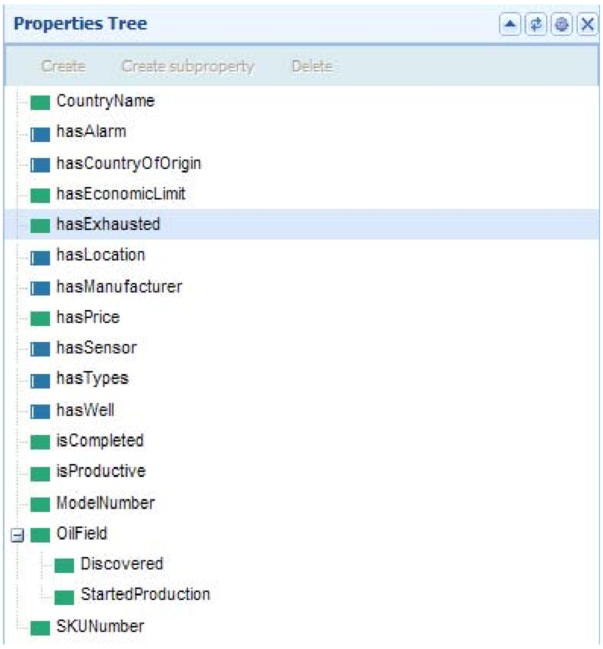
The Team Leader recommends a modification of existing ontology to accommodate the possibility of an oil field being exhausted, as stated previously. For this research, the ontology is created using Protégé version 3.4.4 which is compatible with Web Protégé 0.5 alpha build 300. Figure 2 illustrates Properties within Current Ontology which contains "isProductive", but does not reflect the extent of resource exhaustion. The Team Leader is free to ask The Ontologist to create a Temporary Ontology based on his proposed ontology where "hasExhausted" is added to Properties as one of the Datatype Property.



- CountryName
- hasAlarm
- hasCountryOfOrigin
- hasEconomicLimit
- hasLocation
- hasManufacturer
- hasPrice
- hasSensor
- hasTypes
- hasWell
- isCompleted
- isProductive
- ModelNumber
- OilField
  - Discovered
  - StartedProduction
- SKUNumber

Figure 2: Current/Archived Ontology

While the discussion takes place, other participants are encouraged to provide comments on the proposed ontology. As stated earlier, these participants can also propose their own version of ontology which means there are several Temporary Ontologies. Each member of the Contributor Group votes on the ontology that s/he considers as the most appropriate for the issue. When a "Modified Issue" earns the highest voting point, the existing ontology needs to be corrected. If the ontology proposed by The Team Leader is the most favored throughout the discussion process, the Ontologist adds "hasExhausted" as one of Datatype Property as well as the related Individuals. The updated Current Ontology is presented in Figure 3, while Figure 2 becomes Archived Ontology.



- CountryName
- hasAlarm
- hasCountryOfOrigin
- hasEconomicLimit
- hasExhausted
- hasLocation
- hasManufacturer
- hasPrice
- hasSensor
- hasTypes
- hasWell
- isCompleted
- isProductive
- ModelNumber
- OilField
  - Discovered
  - StartedProduction
- SKUNumber

Figure 3: Temprary Ontology

V. Discussion

In the previous section, the platform for Lightweight Community Support is presented. As illustrated, our approach enables staff from dispersed geographic locations to conduct discussions on various ontology issues and maintain different roles within appointed Groups. The distribution of responsibility between knowledge management (i.e. Contributor Group) and administration of communication (i.e. Admin Group) has contributed to well-organized discussions on ontology of a particular issue within the oil and gas industry.

The platform encourages staff to actively initiate and participate on a discussion of issues on specific topic. Therefore, users are presented with opportunities to acquire knowledge and act upon it on a methodical manner. Users with limited understanding of a particular ontology are benefitted from the platform as it allows them to verify their comprehension and perception through dynamic dialogue with their peers.

The communication reduces misunderstandings about the ontology and minimizes discrepancies in staff understandings and perspectives since staff members have different backgrounds and perceptions on issues. The platform will promote compatibility amongst staff and encourage staff to be responsible for the improvement of ontology. Aside from improved ontology, the platform presents opportunities for staff to enhance their knowledge as well as awareness of particular issues. As staff are encouraged to become involved in domain-based communication in a systematic manner, they acquire the necessary expertise that will be valuable for their subsequent work.

The collaborative discussion which occurs, results in a decision that affects the Oil and Gas ontology which allows it to evolve. The ontology evolution platform provides a way to better understand the ontology that is applied by a particular oil and gas industry. By presenting the ontology online using Web-Protégé, staff are exposed to and gradually gain knowledge of the ontology.

The challenge for any domain-based discussion is to ensure reliability of the results. Based on the demonstrations of the voting feature, this has been overcome by the different weight assigned to the voters. This avoids the validity issue that quantity prevails over the discussion outcome. Taking into account the number of Team Members in the Contributor Group, this potentially affects the voting count but not the voting points since Domain Experts, Ontologist, Team Leader and Team Member have from the highest to the lowest expertise value. Each user accumulates a different reputation value which represents voters' trust level related to providing suggestions to the raised issue. Of the two values, expertise has more significance which means the platform reasoning is sound.

VI. Conclusion And Future Work

This research focuses on a means of improving communication between different participants for the purpose of discussing the current issues arising from the evolution of ontology. This is performed by using a social networking tool to discuss these issues and to make decisions based on the discussion, thereby allowing the ontology to further evolve. This research can be further developed by Integrating Wiki within the lightweight community driven approach to allow further control and formalization of the way to retrieve the information. In addition, with the slow development of Web-Protégé there has not been a feasible way to build/integrate it with other tools or plug-ins; hence, it cannot be applied to all scenarios. Therefore, the integration of Web-Protégé with other tools is vital to allow further expansion of this promising software.

References

[1] Thorsen, K. and C. Rong. Data Integration in Oil and Gas at Norwegian Continental Shelf. in 22nd International on Advanced Information Networking and Applications - Workshop. 2008. Okinawa, Japan.

[2] Noy, N.F. and M. Klein. A component-based framework for ontology evolution. . in In Workshop on Ontologies and Distributed Systems at IJCAI-03. 2003. Acapulco, Mexico.

[3] Noy, N.F., et al. A Framework for Ontology Evolution in Collaborative Environments. in 5th International Semantic Web Conference 2006. 2006. Athens, USA.

[4] Liang, Y., H. Alani, and N. Shadbolt. Ontology Change Management in Protégé. in AKT DTA Colloquium. 2005. Milton Keynes, United Kingdom.

[5] Zablith, F. Dynamic Ontology Evolution. in International Semantic Web Conference (ISWC 2008). 2008. Karlsuhe, Germany.

[6] Tudorache, T., J. Vendetti, and N.F. Noy. Web-Protégé: A Lightweight OWL Ontology Editor for the Web. in OWL: Experiences and Direction (OWLED) Workshop. 2008. Karslruhe, Germany.

[7] Blundell, B. and S. Pettifer. Abstract: Graph Visualisation to Aid Ontology Evolution in Protégé. in 7th International Protégé Conference 2004. Bethesda, Maryland.

[8] Weichselbraun, A., et al., Capturing Ontology Evolution Processes by Repeated Sampling of Large Document Collections, in On the Move to Meaningful Internet Systems 2007: OTM 2007 Workshops. 2007: Vilamoura, Portugal.

[9] Norheim, D. and R. Fjellheim, AKSIO - Active knowledge management in the petroleum industry, in ESWC2006 - 3rd European Semantic Web Conference 2006. 2006: Budova, Montengro.

[10] Fun, R. and C. Wagner, Weblogging: A study of social computing and its impact on organizations. Decision Support Systems, 2007. 45(2008): p. 242 - 250.

[11] Gendarmi, D. and F. Lanubile. Community-Driven Ontology Evolution Based on Floksonomies. in OTM Workshop. 2006. Montpellier, France.

[12] Siorpaes, K. Lightweight Community-Driven ontology Evolution. in 6th International and 2nd Asian Semantic Web Conference. 2007. Busan, Korea.

[13] Aseeri, A., et al. Towards Social Network based Ontology Evolution Wiki for Ontology Evolution. in 10th International Conference on Information Integration and Web-based Applications & Services. 2008. Linz, Austria.

[14] Searls, D. and D. Sifry, Building with Blogs. Linux Journal 2003. 2003(107): p. 4.

[15] Zhou, Y. and J. Davis. Analysis of Weblog Link Structure - A Community Perspective. in WEBIST 2005/2006. 2007. Miami, USA.

[16] da Cunha, J.V. and W.J. Orlikowski, Performing catharsis: The use of online discussion forums in organizational change. Information and Organization, 2008. 18(2): p. 132-156.

[17] Harth, A., J. Breslin, and I. Ohmukai, Linking Semantically Enabled Online Community Sites. 2004, National Universiyt of Ireland: Galway, Ireland.

[18] Malzahn, N., S. SZeini, and A. Harrer, Ontology Facilitated Community Navigation – Who Is Interesting for What I Am Interested in? . Modeling and Using Context, 2005. 3554(2005): p. 292-303.