Developing PMIS for Business Projects Based on Social Science Research Findings and Ontology Modelling

Siaw Ming Wong, Jean-Yves Lafaye, Patrice Boursier

DEVELOPING PMIS FOR BUSINESS PROJECTS BASED ON SOCIAL SCIENCE RESEARCH FINDINGS & ONTOLOGY MODELLING

Siaw Ming Wong
Open University Malaysia / Université de La Rochelle
Jalan Tun Ismail, 50480 Kuala Lumpur, Malaysia

Jean-Yves Lafaye
Université de La Rochelle
Avenue Michel Crépeau, 17000 La Rochelle, France

Patrice Boursier
Université de La Rochelle
Avenue Michel Crépeau, 17000 La Rochelle, France

ABSTRACT

The purpose of the research is to obtain a better understanding of the essential components of business project management and capture the new gained knowledge in a machine readable format to facilitate software engineering. This paper presents how this interdisciplinary link between social science and computer science can be established through ontology modelling; and demonstrates how the resulting model can be used by the requirements analysis or conceptual modelling phase in the development of a Project Management Information System (PMIS) for business projects.

KEYWORDS

Project Management, Modelling, Ontology, Unified Modelling Language.

1. INTRODUCTION

Companies today are entrusting their business projects to certified project managers, in acknowledgement of the positive causal relationship between project management effectiveness on project performance (Bryde, 2003). Nonetheless business project success rates remain low where one in three projects fails to deliver on time or within budget; and 60% fails to deliver the benefits as set out in business case (PIPC Global Project Management Survey, 2005).

The development of the project management research can be divided into 3 strands (1) hard system model which emphasizes the planning and control dimensions; (2) organization design which focuses on the organization structure as a means to achieve integration and task accomplishment; (3) broader view of projects which emphasizes context, front-end work, learning and managing exogenous factors (Winter et al, 2006). What is still lacking is approaching the problem with project performance from the holistic approach by assessing the collective influences of the various factors and demonstrating the relevance of project management in business (Morris, 2003). The closest work found was the investigation into project management effectiveness by considering organizational structure, technical competency, leadership ability and the characteristics of an effective project manager (Hyväri, 2006). The selection of these variables however, was motivated by the discrete gaps identified by previous researches with emphasis on the factors influencing the ability of the project managers in their project delivery. The path of examining project performance from an organizational perspective remains largely unexplored.

The objective of the social science research therefore, is to (1) identify the essential components of business project management based on a comprehensive understanding of collective influences of organizational variables on project success; (2) captures the findings into the model as a reference to guide the business project management practice in order to increase chances of project success, while its representation in a machine readable format expedites the injection of this new domain knowledge into a software solution. This paper will touch on all aspects of the research with focus on point (2), by outlining how the social science research findings can be transformed into a machine readable ontology model using Unified Modelling Language (UML); and provides an example of how the resulting model can be used as the foundation to support the development of an integrated Project Management Information System (PMIS) for business projects.

2. ONTOLOGY MODELLING

The first task of the modelling exercise is to decide on the format which could meet the stated objectives [of representing knowledge in a model that can be fed directly into a software development cycle] and ontology is chosen as the starting point because of its origin as the centre piece in the Knowledge Representation (KR) paradigm. Defined as "an explicit specification of conceptualization" (Gruber, 1992), an ontology is virtually always the manifestation of a shared understanding of a domain that is agreed between a number of parties which in turn, facilitates inter-operability, reuse and sharing (Uschold, 1996).

Since ontology can take a variety of forms, it is important to understand the key dimensions of which they may differ. One of such dimensions is the degree of formality or explicitness in the vocabulary of terms and the specifications of their meanings that comprise the ontology (Ushold & Gruninger, 1996). While knowledge specified in the form of informal ontology may be contradicting the definition of ontology as an explicit specification and cannot be leveraged for automated reasoning, formal ontology can be difficult to maintain especially if the process involves non-technical users. A balance between the two extremes is therefore desired and one resolution is the use of conceptual model. This idea is not new as highlighted by Gómez-Pérez et al (2006), formal ontologies should be reverse-engineered for their underlying conceptual models so that a better understanding of the existing ontologies could be obtained before they are updated with new learning or merged/aligned with other ontologies to form the new ones. In addition, the use of conceptual model during the conceptualization phase of an ontology engineering process helps to address issue caused by the limited capacity of the formal language in capturing the problem domain (Nagypal, 2007). Last but not least, conceptual models are expressed in a form comprehensible by both the domain experts and the ontology engineers.

In the context of software engineering, ontology expresses semantics and provides reasoning, querying and navigating facilities; whilst conceptual model represents aspects of a domain to be incorporated into an information system. In which case, ontology (its semantics in particular) can be used to facilitate conceptual modelling but due to their differences in formats, the desired reusability between ontologies and conceptual models can only be achieved through indirect means, e.g. transformation through meta-model mapping (Vasilescas et al, 2006) and enriching the conceptual models with ontology-based semantic annotation (Zouggar, 2006). The ideal alternative would be to develop a foundation model based on which both ontologies and conceptual models can be built upon.

The proposal for this research therefore, is to develop the Business Project Management (BProjM) ontology model using Unified Modelling Language (UML). The choice of UML specification was driven by both the need to adopt a pseudo technical standard as well as to leverage on the Model Driven Architecture (MDA) approach where the desired ontologies and information systems can be developed based on transformation of the resulting ontology model in stages. Figure 1 below indicates the positioning of the ontology model in this research as well as in relations to the software engineering and ontology engineering paths. For the software development path on the left, the developed ontology model could be fed directly into the conceptual modelling / requirement specification phase to support the development of an integrated Project Management Information System (PMIS) for business projects. For the ontology engineering path on the right, the UML *Class Diagram* can be translated to OWL ontology to allow further inference and exploitation of the ontology in semantic web applications. Based on OMG's Ontology Definition Metamodel (ODM) specification [http://www.omg.org/spec/ODM/1.0/], a canonical ontology can be built automatically and then appropriately enriched via annotation and model weaving techniques (Faucher et al, 2008).

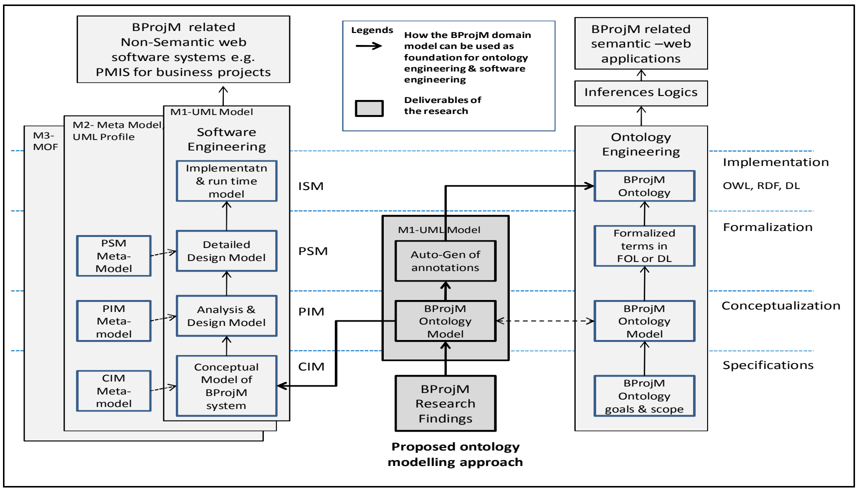

Figure 1. Positioning of the ontology model in the context of this research

3. RESEARCH METHODOLOGY

The research adopts the deductive research strategy and applies the systems school of management theory in support of the holism theme where "Organization can be viewed as open systems which take input from the environment (outputs from other systems) and through a series of activities transform or convert these inputs into output (input to other systems) to achieve some objectives" (Mullin, 2007). By examining the relevancies of the sub-systems which must work together to carry out the "series of activities" in the context of business projects, a theoretical framework with 4 constructs was developed and 3 hypotheses were derived namely (1) Project Management Competency has a direct positive impact on business project success; (2) Comprehensiveness of the PMIS support places a positive impact on the way project management competency contribute towards business project success; (3) The level of integration between the project organization and the parent organization places a positive impact on the way project management competency contribute towards business project success. The case study approach is adopted to validate these hypotheses as it offers the opportunity for a holistic view of a process (Patton & Applelbaum, 2003), thus permitting the attainment of a richer understanding on a well researched topic from a fresh angle.

The key component bridging the two disciplines is a consolidated form of the social science research findings which lists the essential concepts and their inter-relationships. This can be achieved by updating the original theoretical framework to reflect only the key attributes of each construct which contributes to a successful business project based on the case study findings. Given this edited theoretical framework, the following steps of developing the ontology model can then be carried out systematically:

- Define the scope and translating the research findings into requirements
- Identify related research work in the past and examine whether they can be reused
- Build the model based on the stated requirements by leveraging on existing work as appropriate
- Test the completeness and integrity of the model using the case study data.

4. CASE STUDY FINDINGS

Three companies have participated in the case study. The anchor case study is an international airline which had successfully completed a business turnaround programme; the second case study is an operator of satellite pay television services in the region whose corporate wide business transformation programme is currently on a recovery path; and the third case study is a large local banking group which has just initiated their transformation programme due to a recent investment from a foreign bank.

In summary, the case study findings had ascertained the hypotheses, in addition, it was found that (1) Business project success should also be measured at the organizational level. This findings is in agreement with Turner & Müller (2006) that project is a multidimensional strategy concept and Faisal (2006)'s view that project success should be evaluated in terms of production of the expected deliverables and whether these deliverables achieve the intended organization goals; (2) The project management competencies required of the business project managers are less comprehensive and technical in nature and these "Core Business Project Management Competencies" are integration management, scope management, time management and communication management. In other words, business project managers are not expected to be the master of the trade but one who facilitates the work of the domain masters; (3) An integrated programme management function is essential in maintaining on-going alignment between the project output and the organizational goal. Although the role of programme office /programme management office in marshalling projects and resources to achieve the desired strategic and/or synergy benefits is not new (Pellegrinelli, 2002), the emphasis has been placed on its configurations rather than its effectiveness especially in terms of performing on-going close-loop monitoring; (4) The use of IT in support of project work needs to be elevated to the organization level. Project learning for example, would add more value to future corporate planning if each project implementation is captured as an episode in the organization's chronology with causes and effects.

Based on the above findings, an underlying business project management theory i.e. "Business projects must be managed as an integral part of the organization" is developed. Such integration clearly demands the enablement of seamless information exchange between the project and the permanent organization, which is a system support feature less relevant to the traditional construction or engineering projects. By updating the original theoretical framework with these research findings (see figure 2), the modelling exercise is now ready to commence.

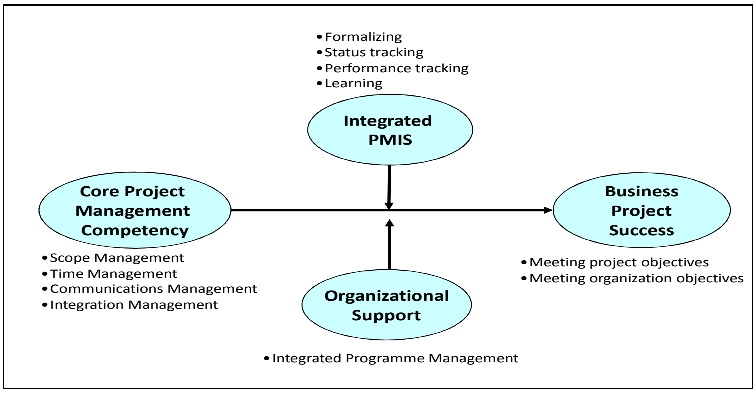

Figure 2. Social science research findings expressed in the form of theoretical framework

5. DEFINING SCOPE & REQUIREMENTS

The scope of the ontology model is provided by the edited theoretical framework where all its stated components must be represented in the ontology model. But in order for the model to be a meaningful foundation for a specialised PMIS, each of these components must be expanded with secondary level details. This is achieved by adopting relevant standard project management terms in the Project Management Body of Knowledge (PMBOK, 2004) published by the Project Management Institute (PMI) which is chosen due to its large member base and wide acceptance by the international communities. To illustrate how this is done, "Communication Management" in the "Core Business Project Management Competency" for example, can be expanded with supporting terms "communication management plan", "project records" and "project reports".

Finally, based on the developed theory that the business project is an integral part of the enterprise, the ontology model must also show how these business project management components interact with their counterparts in the parent organization. As such, the consolidated requirements of the ontology model are to capture (a) essential business project management terms and their inter-relationships; (b) their interfaces with the other components in the parent organization.

6. REUSING EXISTING WORK

Given the intent to leverage on MDA approach to transform the resulting model into an implementable equivalent, the Business Project Management (BProjM) ontology model is built based on OMG's Business Motivation Model (BMM), Business Process Definition MetaModel (BPDM) and Organization Structure MetaModel (OSM) which collectively provide a comprehensive set of meta-classes that define the fundamentals of an enterprise.

As with the related work in the project management domain, no existing project management model in a reusable form is found. Existing project and project management ontologies on the other hand, are designed to serve very specific purposes (Garcia et al, 2003; Gómez-Pérez et al, 2006; Dumbill 2004) and thus, only PROMONT (PROject Management ONTology) developed by Abels et al (2006) can be adopted. Although it is originally developed to support product development projects in distributed virtual project environment, PROMONT summaries all major project management standards including PMBOK. It is therefore still generic enough to serve as a starting point for this research.

7. DEVELOPING THE ONTOLOGY MODEL USING UML

In compliance with UML standards, the research uses *Class Diagram* for structural specifications and *Communications Diagram* in conjunctions with *State Machine Diagram* to specify the behavioural aspect.

7.1 Structural specifications

The structural specification starts with defining *Packages* to group the terms by their origins namely BMM (a separate *Package* was defined for BMM Place Holder in accordance with OMG's BMM specification), BPDM, OSM and the to-be developed BProjM. Each essential business project management terms [in the *BProjM Package*] are then transformed into *Classes* and further grouped into sub-*Packages* based on their nature and purpose. *Core Package* for example, contains basic components of a project/programme initiative such as *Phase*, *Task*, *Activity* by leveraging on PROMONT's terms; whilst *Project Results Package* contains terms related to project performance and the measures for its success.

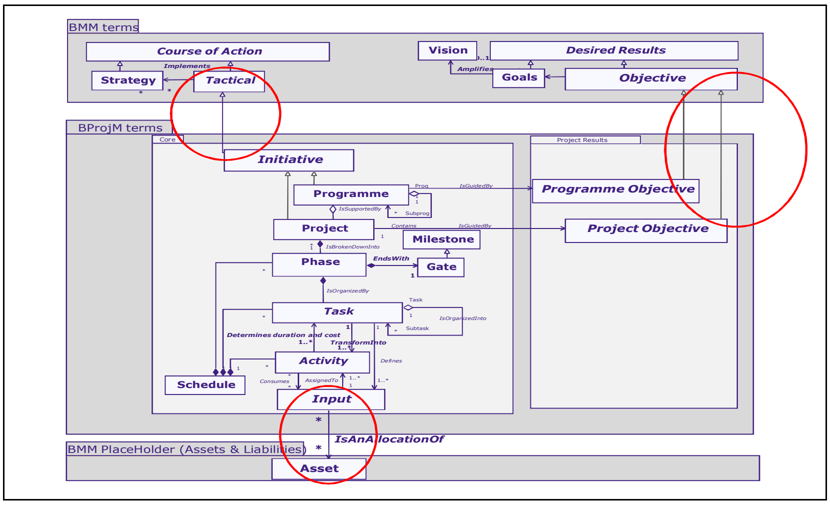

Figure 3. BProjM ontology model – sample class diagram

As illustrated by figure 3, the next step is to define the *Associations* among the *Classes* within the *BProjM Package* to reflect the inter-relationships which justify their logical groupings. This is followed by defining *Associations* among *Classes* across *Packages*, e.g. *Programme* and *Project* in the *Core Package* are linked to *Programme Objective* and *Project Objective* respectively in the *Project Results Package.*

The next important step will be to fulfil the second requirement by establishing linkages between each *Class* in the *BProjM Package* with the meta-classes in BMM, BPDM and OSM. Subject to the nature of the terms, this can be achieved in two ways by defining either (a) Inheritances or (b) Associations. Riding on the same example and highlighted by the circles in figure 3, *Initiative* which is a generalization of *Programme* and *Project,* is now tied to *Tactical* in *BMM Package* through an inheritance link, reflecting its nature as a special type of tactical *Course Of Action*. *Programme Objective* and *Project Objective* are also linked to *Objective* in the same manner. An example of linkage by *Association* on the other hand, is demonstrated by the relationship where the "*Input" Class* in the *BProjM Package "IsAnAllocationOf" the "Asset" Class* in the *BMM PlaceHolder Package*.

The final step of structural specification is to substantiate the *Classes* and *Associations* with more explicitness by defining *Attributes, Operations* and *Constraints.* The "*Input" Class* for example, is defined with (1) *People, Equipment* and *Materials* as its *Attributes* to reflect the type of assets which may be allocated to the project; (2) *AllocateInput()* and *UpdateConsumption()* as its *Operations* in support of the *Association "AssignedTo" and* "*Consumes*" respectively; (3) a *Constraint* indicating that during *AllocateInput()*, *People / Equipment / Materials* assigned to the project must first be available in *Assets*.

7.2 Behavioural specifications

To represent the dynamic aspect of the model, the exercise begins by using the *State Machine Diagram* to describe the behaviour of the "ruling" C*lass,* i.e. a *Class* of which its evolution will trigger different instantiations of other classes at different times. In the case of business project management, its dynamism is induced by the transition of states in the business project life cycle. The "ruling" *Class* for BProJM ontology model is therefore *Initiative* which is the parent *Class* of *Programme* and *Project*. By using *State Machine Diagram,* the advancement of *Initiative* from one state to another as well as the conditions that govern the transition can be clearly spelt out. In a similar context but a more refined coverage, *State Machine Diagram* is also used to elaborate on the more sophisticated *Classes,* e.g. *Issue* and *Change Request* to reflect their "ruling" status in Change Management during the *executing* state.

Finally, for each state specified in the *State Machine Diagram,* a *Communication Diagram* is created to demonstrate the interactions among the C*lasses* as well as the sequence of their occurrences, through invoking the predefined *Operations*. In addition, dependencies among *Classes* can also be indicated. The best example would be the communications among *Classes* triggered by the update of activity status during a project's execution state as illustrated by figure 4.

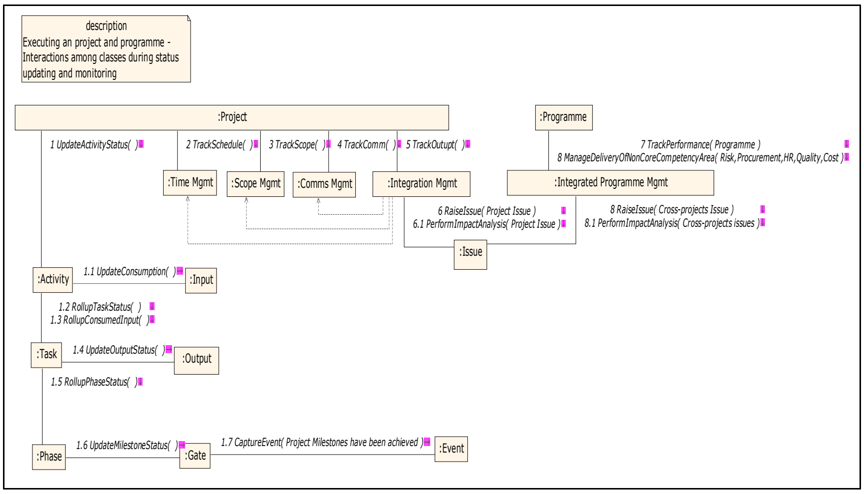

Figure 4. BProjM ontology model - sample communications diagram

7.3 Testing the developed ontology model

The model is tested for completeness by instantiating the *Classes* using *Object Diagram* based on the details gathered during the case study. It is then refined based on test results and the cycle repeats itself until every aspect of the business project management practice of the participating company can be reflected in the model. One of the discrepancies uncovered in an earlier version of this model was that programme may be an *Aggregation* of projects as well as sub-programmes. The model has since been revised and *Programme* is now added with another *Association* that points to itself.

8. USING THE ONTOLOGY MODEL

Leveraging on the predefined *Classes, Operations* etc. in the model, the user requirements of a PMIS catering specifically for the need of business project management can now be easily captured using (a) *Use-Case Diagram* to describe the requirements of the different stakeholders and to provide an overview of the required system functionalities in fulfilment of those requirements; (b) *Sequence Diagram* to denote the series of actions that must be carried out in order to deliver the required system functionality, where existing *Communication Diagrams* with predefined interactions among *Classes* can be used as a basis and build upon with additional features and processing logics as necessary.

The *Use-Case Diagram* in figure 5 presents a scenario where the PMIS captures the team members' update of activity status; based on which derives the status for the task, deliverables, phase and milestones automatically; and finally formats the information according to the expectations of different stakeholders.

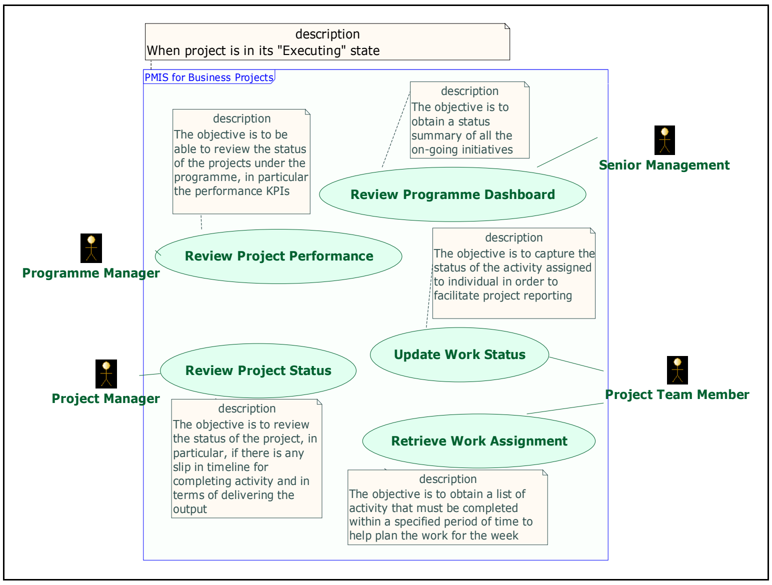

Figure 5. BProjM ontology model – sample use case diagram

Using "Update Work Status" as an example (see figure 6), the *Sequence Diagram* can be drawn by first stating the user requirements in a *Note* and associating it to an interface definition between the actor and the system. The software engineer can then detail the series of required actions by referring to the related *Communication Diagram* already defined in the ontology model and sequence the standard interactions accordingly. As the *Sequence Diagram* is being developed, he/she can define new *Classes* to represent the system components, as well as adding new *Operations* into the existing *Classes* as required.

In addition to the reduced effort in detailing user requirements, the ontology model provides the much needed context for the software engineers to better comprehend the different stakeholders' needs, thus facilitating the augmentation from conceptual modelling (i.e. "what the system should do") to requirement engineering (i.e. "why is the system like this") as proposed by Rolland (2006). With a more in-depth understanding on the subject matter, completeness of the resulting conceptual model for the desired PMIS is assured which in turn, leads to the delivery of a quality system.

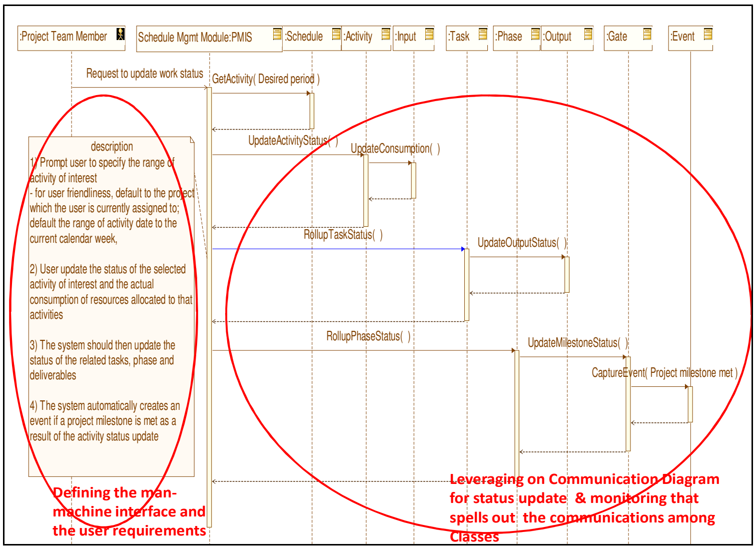

Figure 6. BProjM ontology model – sample sequence diagram

9. CONCLUSION : CONTRIBUTION, ISSUES & FUTURE RESEARCH

The proposed methodology can be applied to other management subject of interest as long as the findings can be expressed in the form of theoretical framework which can be expanded based on recognized standards. In which case, consistent application of this methodology over time would contribute to the development of a multi-facet enterprise ontology model. The availability of this semantic foundation should in turn, place a considerable impact on the existing landscape of software engineering where systems can now be built based on predefined knowledge specifications of the domain, rather than user input from various sources. In other words, comprehensiveness in fulfilling the domain-imposed requirements is no longer a differentiator and the enterprise application systems in the future, would compete primarily in terms of superiority in design and technicality.

The realization of such an enterprise ontology model however, would need support from a central agency to host the convergence of all related management research efforts. OMG would be in the best position to play that role since the ontology model is built upon its standards which also require on-going refinements. This modelling exercise have already identified some *Classes* such as those defined in the *Documentation Package* which should technically be applicable to the enterprise at large but there is no suitable meta-classes in BMM, BPDM nor OSM today where they could be built upon. The central agency will also play an important role in providing the forum to resolve semantic conflicts between new and existing *Classes*, which are expected to occur with the expansion of the ontology model.

The other concern which may potentially hamper the acceptance of the methodology is the lack of UML knowledge and understanding of the object oriented concepts among the social scientists. Thus, until unification of modelling standards across disciplines is achieved, close collaborations in the form of joint researches must be encouraged and promoted.

As an immediate next step, the comprehensiveness of the resulting business project management ontology model can be refined and further generalized by incorporating findings of more case studies. On the technical front, a potential research area would be to supplement the current ODM specifications with a solution that translates *State Machine Diagram* into OWL ontology.

REFERENCES

Book

A Guide to the Project Management Body of Knowledge (PMBOK® Guide) (ed.) 2004. Project Management Institute, Pennsylvania, USA.

Faisal, A., 2006. *Trapped in between: Realities of the project world.* Trafford Publishing, Victoria, Canada.

Gómez-Pérez A. et al, 2006. *Ontology Engineering.* Springer, Berlin, Germany.

Morris, P.W.G., 2003. 'Research trends in the 1990s – the need now to focus on the business benefits of project management' in *The Frontiers of Project Management Research,* PMI, Pennsylvania, Chapter 2.

Mullin, L.J.,(ed.) 2007. *Management and Organizational Behaviour.* Prentice Hall, Great Britain

Nagypal, G., 2007. 'Ontology Development' in *Semantic Web Services,* Springer, Berlin, pp. 107-134.

Turner, R. & Müller, R., 2006. *Choosing Appropriate Project Managers: Matching Their Leadership Style to the Type of Project,* Project Management Institute, Pennsylvania, USA.

Journal

Bryde, D.J., 2003. Modelling project management performance. *International of Quality & Reliability Management*, Vol. 20, No. 2, pp. 229-254.

Faucher C. et al, 2008. Génération d'ontologie à partir d'un modèle métier UML annoté. *Numéro spécial de la Revue des Nouvelles Technologies de l'Information (RNTI) sur la Modélisation des Connaissances*, Cépaduès Editions, Vol. RNTI E-12, pp. 65-84.

Hyväri, I., 2006. Project management effectiveness in project-oriented business organizations. *International Journal of Project Management*, Vol. 24, No. 3, pp. 216-225.

Patton, E. & Appelbaum, S.H., 2003. The case for case studies in management research. *Management Review News,* Vol. 26, No. 5, pp. 60-71.

Uschold, M. & Gruninger, M., 1996. Ontologies: Principles, Methods and Applications. *Knowledge Engineering Review*, Vol. 11, No. 2.

Conference paper or contributed volume

Abels S. et al, 2006. PROMONT – A Project Management Ontology as a Reference for Virtual Project Organizations. *OTM Workshops.* LNCS 4277, pp. 813 – 823.

Gómez-Pérez A. et al, 2006. R&D Project Management with ODESeW. *OTM Workshops.* LNCS 4277, pp. 36 – 37.

Rolland, C., 2006. From Conceptual Modelling to Requirements Engineering. *ER 2006.* LNCS 4215, pp. 5 -11.

Uschold, M., 1996. Building ontologies : towards a unified methodology. *AIAI-TR-197. Proceedings of expert system 1996.* Cambridge, UK.

Vasilecas O. et al, 2006, On approach on enterprise ontology transformation to conceptual model. *CompSysTech'06, International Conference on Computer Systems and Technologies.*

Zouggar, N., 2006. Semantic Enrichment of Enterprise Modelling. *Doctoral Symposium*, *Interoperability for Enterprise Software and Applications (I- ESA) International Conference.*

Electronic Publication

Dinsmore, P., 2006. *Why should project management matters to CEO?.* viewed 12 April 2007*,* <http://www.chiefprojectofficer.com/column/175>.

Dumbill, E., 2004. *Keep project information up-to-date with the DOAP vocabulary*, viewed 18 August 2008, <http://www.ibm.com/developerworks/xml/library/x-osproj.html>

Garcia A.C.B. et al, 2003. *Building a Project Ontology with Extreme Collaboration and Virtual Design & Construction*. CIFE Technical Report #152, Stanford University, <http://cife.stanford.edu/online.publications/TR152.pdf>

Gruber, T., 1992. *What is an Ontology?* viewed 16 March 2007, <http://www.ksl.stanford.edu/kst/what-is-an-ontology.html>.,

PIPC Global Project Management Survey, 2005. PIPC, viewed 15 April 2007, <http://www.pmportal.co.uk/uploads/documents/PIPCSurvey.pdf>