Journal of Universal Computer Science, vol. 17, no. 8 (2011), 1161-1182
submitted: 1/6/10, accepted: 1/2/11, appeared: 28/4/11 © J.UCS

ORPMS: An Ontology-based Real-time Project Monitoring System in the Cloud

Hai Dong
(Digital Ecosystems and Business Intelligence Institute,
Curtin University of Technology, Perth, WA 6845, Australia
hai.dong@cbs.curtin.edu.au)

Farookh Khadeer Hussain
(Digital Ecosystems and Business Intelligence Institute,
Curtin University of Technology, Perth, WA 6845, Australia
farookh.hussain@cbs.curtin.edu.au)

Elizabeth Chang
(Digital Ecosystems and Business Intelligence Institute,
Curtin University of Technology, Perth, WA 6845, Australia
elizabeth.chang@cbs.curtin.edu.au)

Abstract: Project monitoring plays a crucial role in project management, which is a part of every stage of a project's life-cycle. Nevertheless, along with the increasing ratio of outsourcing in many companies' strategic plans, project monitoring has been challenged by geographically dispersed project teams and culturally diverse team members. Furthermore, because of the lack of a uniform standard, data exchange between various project monitoring software becomes an impossible mission. These factors together lead to the issue of ambiguity in project monitoring processes. Ontology is a form of knowledge representation with the purpose of disambiguation. Consequently, in this paper, we propose the framework of an ontology-based real-time project monitoring system (ORPSM), in order to, by means of ontologies, solve the ambiguity issue in project monitoring processes caused by multiple factors. The framework incorporates a series of ontologies for knowledge capture, storage, sharing and term disambiguation in project monitoring processes, and a series of metrics for assisting management of project organizations to better monitor projects. We propose to configure the ORPMS framework in a cloud environment, aiming at providing the project monitoring service to geographically distributed and dynamic project members with great flexibility, scalability and security. A case study is conducted on a prototype of the ORPMS in order to evaluate the framework.

Keywords: Ontology, Product Quality Evaluation, Project Management, Project Monitoring, Project Progress Assessment
Categories: D.2.8, D.4.7, I.2.4

1 Introduction

Project management, as an important part of organizational innovation, has evolved from a management process with restricted functions to a business process having an overall impact on organizational survival [Kerzner 2009]. As a critical part of project management, project monitoring ensures that project managers can quickly and

efficiently control the progress of projects, risks, product quality and changes, which occur at every stage of a project's life-cycle [Crawford and Bryce 2003]. However, project monitoring is suffering from the impact of globalization. On the one hand, project monitoring requires high-quality knowledge sharing between employees in project organizations; on the other hand, along with the development of economic globalization, outsourcing probably becomes inevitable in many companies' strategic plans in terms of companies' perennial goal of pursuing lower costs. This outsourcing trend necessarily leads to the dynamism of project environments, the complex structure of project organizations, the geographical dispersal of project teams and the cultural diversity of team members, which consequentially has a negative influence on the knowledge sharing activities within project organizations [Chinowsky and Goodman 1996, Dong et al. 2007a, Pich et al. 2002].

Cloud computing is the next generation of service delivery paradigm, which aims at providing IT services with lower costs, fewer resources, higher security etc., through data centers or servers [Armbrust et al. 2009]. Cloud services include software (SaaS), platform (PaaS), and infrastructure (IaaS). In this paper, we present the framework of an Ontology-based Real-time Project Monitoring System (ORPMS), in order to address the ambiguity issue in the project monitoring processes. It is proposed that this system be set up in a cloud computing environment, which enables the provision of internet-based services to the geographically dispersed and dynamic project team members in terms of enhanced flexibility, scalability and security as well as transparent data backup. The core elements of the ORPMS include a project monitoring ontology, a project management ontology, and a set of metrics for assessing project completion status and product quality. This methodology aims to solve the ambiguity issue in project monitoring processes and help the management personnel of organizations to better control the progress of projects and to timely recognize potential risks and solve problems.

The rest of the paper is organized as follows: in Section 2, we survey the existing literature in the field of ontology-based project management approaches and analyze the research issue that needs to be addressed; in Section 3, we propose an overview of the ORPMS framework and briefly introduce its workflow; in Section 4, we introduce the project monitoring ontology which is designed to enhance project monitoring, knowledge capturing, storage and sharing, and the project management ontology which is designed to solve the ambiguity problem in project knowledge sharing activities; in Section 5, we respectively describe the metrics for assessing project completion status and for evaluating product quality, with the purpose of enabling management of project organizations to better control projects; in Section 6, we present a prototype for validating the ORPMS framework; the conclusion is drawn and the future work is presented in the final section.

2 Related Work

Extensive products for project management have been developed, according to our understanding, for project monitoring (e.g., Microsoft Office Project Server, Oracle Project Portfolio Management, etc.), with the purpose of improving knowledge sharing within project organizations and thus enhancing the efficiency and quality of project monitoring. Nevertheless, because of the lack of a uniform data standard and

insufficient disambiguation support, there arises an ambiguity issue within the project data exchange between different products, and therefore project teams with different products rarely collaborate to realize multi-site project monitoring. Ontology is regarded as an efficient notational system for representing domain knowledge and enhancing knowledge exchange among different agents [Bhatt et al. 2004a, Bhatt et al. 2006, Bhatt et al. 2004b, Esposito et al. 2008, Flahive et al. 2005, Flahive et al. 2009, Martin et al. 2008, Martino 2009, Vega-Gorgojo et al. 2008]. In order to address the ambiguity issue within the knowledge sharing activities in project management, many ontology-based project management approaches have been developed.

First of all, in 2005, Clark [Clark 2005], the project manager for the Project Management Institute (PMI) (http://www.pmi.org/), provided a position paper concerning the rules for building project management ontologies. The PMI developed an Organizational Project Management Maturity Model (OPM3), which is based on the Project Management Body of Knowledge (PMBOK) [Project Management Institute 2004], as a standard for ontology design. In the same year, Bullinger et al. [Bullinger et al. 2005] proposed an ontology-based project management approach for the acceleration of innovation projects. An innovation-process ontology was developed in order to express complex interrelations between essential elements in innovation processes and to turn the processed information into knowledge. In addition, rules were designed in order to determine excessive time consumers in innovation processes. Simultaneously, Chan and Chang [Cheah and Chang 2005] initiated an ontology-based approach for managing multi-site system development. The ontology was proposed in order to cover multiple standards and define the terms, issues, risks and workflow requirements of multi-site project management. Dustdar initiated an activity-based knowledge management approach which links domain knowledge (e.g. project documents) with project management activities performed by human actors, which reconciles knowledge management and workflow management in project management [Dustdar 2005]. In 2006, Abels et al. [Abels et al. 2006] proposed a project management ontology (PROMONT) in order to regulate the data standard for the exchange of project data between different project management software. The PROMONT summarizes the typical concepts and their meanings for project planning, which provides project participants with a shared vocabulary. By means of this ontology, terms used by different project management software can be formally defined, and different project plans, management methods and data formats can be integrated. In 2009, Aramo-Immonen [Aramo-Immonen 2009] focused on using a project management ontology to explore project learning. This ontology is a classification of management disciplines for project managers. The objective of the research is to help system integrators to manage the evolution of projects during their life-cycles in terms of this ontology.

The common defects of the existing ontology-based project management approaches are that none of them proposes a means for project monitoring and they all ignore the ambiguity issue in project monitoring. As a crucial part of project management, project monitoring guarantees that management of project organizations will obtain the latest information regarding project completion status, product quality, cost, risks and so on, in order to ensuring project outcomes to completely fulfill stakeholders' expectations. Nevertheless, as presented in the Introduction, the

ambiguity issue obstructs the progress of project monitoring activities, and the contemporary ontology-based approaches all ignore this issue.

3 System Architecture

In order to address the problem of the lack of ontology-based project monitoring approaches and the ambiguity issue in project monitoring, in this paper, we present the framework of an ontology-based real-time project monitoring system (ORPMS). This system is designed in order to realize the following functions:

1. Real-time project monitoring – enabling the management of a project organization (e.g., project board chairs or project managers) and relevant stakeholders to monitor project completion status and product quality among geographically dispersed project teams in real time.
2. Project knowledge sharing – enabling employees of project organizations to share the knowledge regarding the project monitoring and addressing the ambiguity issue in the knowledge sharing.

It needs to be noted that the scope of this research primarily focuses on designing methodologies for project completion status monitoring and product quality monitoring, in spite of the technical details of other methodologies, such as risk assessment and so on.

Fig. 1 shows the system overview of the ORPMS. The system primarily consists of three components – a Web interface, a project monitoring knowledge base and a project monitoring database. We introduce their functions below.

- Web interface. The Web interface is responsible for interacting with any internal users (employees of project organizations) and external users (stakeholders) in order to realize the proposed functions of the ORPMS.
- Project monitoring knowledge base. The project monitoring knowledge base is used to store the project monitoring knowledge and the project management knowledge, which can respectively be represented by two ontologies as follows:
    - Project monitoring ontology. This provides the knowledge for realizing project monitoring and for producing monitoring reports, including basic terms, workflows and protocols, which are introduced in Section 4.
    - Project management ontology. This is designed to represent project management knowledge, which provides a shared vocabulary for addressing the ambiguity issue in knowledge sharing in project monitoring activities, which is introduced in Section 4.
- Project monitoring database. This is designed to store the data obtained from the project monitoring process for the purpose of backup. The data are, namely, the instances of the concepts in the project monitoring ontology.

All the three components are supposed to be located in a cloud environment, which allows both internal and external project members to access the system and to

use their provided services with great flexibility, efficiency, scalability and security. In addition, the centralized cloud environment eases the task of software update.

The principle of the ORPMS is to use the project monitoring ontology to create online forms in order to record the real-time status of projects, and to use several metrics to quantitatively express the completion status of projects and the quality of products resulting from the projects. We introduce the workflow of the system as follows:

- **Step 1.** In the initial stage of a project, relevant employees (e.g. project managers) need to input the detailed project plan into the system, including responsible employees and their roles, schedule (stages and tasks), timing, budget, and resources. More importantly, they need to design domain-dependent criteria for assessing project completion status and product quality based on stakeholders' expectation.
- **Step 2.** Once the criteria have been designed by project organizations, stakeholders need to evaluate the clarity and importance of the criteria.
- **Step3.** Once the criteria have been reviewed by stakeholders, relevant employees need to revise the criteria according to the stakeholders' evaluation result. Step 1 to Step 3 is a recursive process until all the criteria are mutually agreed to between project organizations and stakeholders.
- **Step 4.** Once all the criteria have been mutually accepted by both sides, the project can start.
- **Step 5.** During the course of the project, relevant employees (e.g. team managers) need to regularly record the progress of a task into the system.
- **Step 6.** During the course of the project, authorized people (e.g. project managers) can gain access to the system to obtain real-time project completion status data. We design a set of metrics for the project completion status assessment, which are described in Section 5.1.
- **Step 7.** Once a product has been generated by a task, relevant employees (e.g. project managers) need to evaluate the quality of the product based on the designed criteria and then enter the assessment result into the system. We design a set of metrics for the product quality evaluation, which are introduced in Section 5.2.
- **Step 8.** During the course of the project, relevant employees need to identify (e.g. team managers), approve (e.g. project managers), and review risks (e.g. project assurance officers) and submit risk logs into the system. Relevant employees (e.g. project assurance officers) also need to assess the impacts of risks and log them into the system.

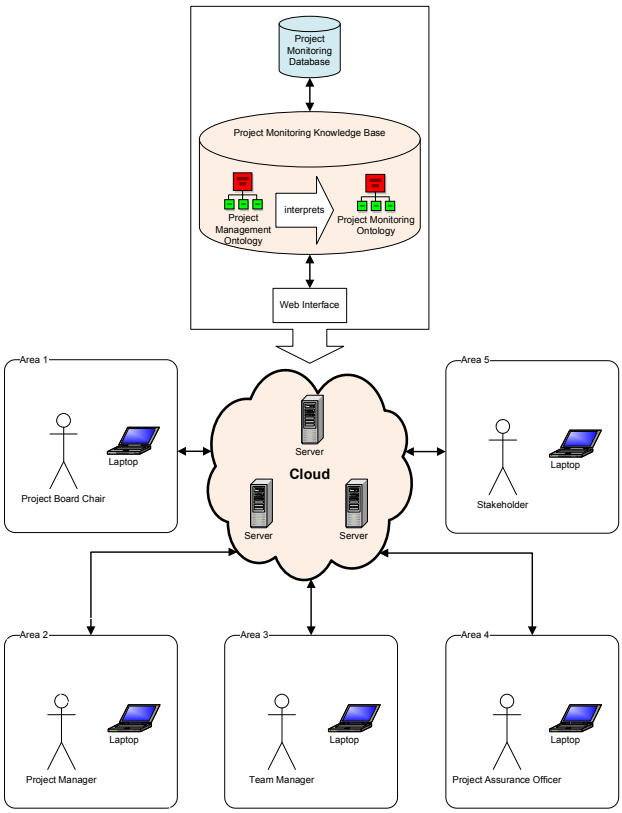

Figure 1: System overview of the ORPMS

4 Project Monitoring Ontology and Project Management Ontology

Ontologies represent the knowledge shared by specific domains [Gruber 1995]. In this section, first of all, we present a project monitoring ontology, as the key element of the proposed real-time project monitoring system, in order to represent the knowledge in project monitoring processes and facilitate the capture, storage and sharing of knowledge in these processes. An abbreviated view of the project monitoring ontology is displayed in Figure 2. The project monitoring ontology primarily comprises a number of concepts which can be described as follows:

- Project. A project is "*a temporary endeavor undertaken to create a unique product, service, or result*" [Project Management Institute 2004]. A project can be divided into several stages.
- Stage. A stage refers to a phase of a project according to a project plan. A stage involves several tasks.

- Task. A task is an activity that needs to be accomplished within a defined period of time.
- Cost. Cost refers to the amount of money that has been expended on a project/stage/task. The sum of costs of all tasks or all stages of a project is equal to the cost of this project. In a project plan, the cost of a project and its stages and tasks are controlled by budget.
- Resource. Resources in a project/stage/task refer to the equipment or facilities used to carry out the project/stage/task. Similar to cost, the number of resources used in a project is equal to the sum of the number of resources used in all stages or all tasks in this project. Additionally, the use of resources in a project is controlled by a project plan.
- Timing. Timing refers to the start time and the end time of a project/stage/task. The time interval of a project is equal to the sum of the time intervals of its involved stages. The timing of a project/stage/task is controlled by a project plan.
- Completion Status. Completion status refers to the degree of completion of a project/stage/task, which is controlled by a series of domain-specific task completion assessment criteria. We present a set of assessment metrics in Section 5.1.
- Product. Product refers to the material outcome of a task. The quality of products is controlled by domain-dependent product quality evaluation criteria. We present a set of quality evaluation metrics in Section 5.2.
- Progress. Progress is used to describe the detailed information regarding the completion status of a task.
- Criteria. There are two kinds of criteria in this ontology, which are task completion status assessment criteria and product quality evaluation criteria, with the purpose of quantitatively assessing task progress and evaluating product quality. The task completion status assessment criteria are task-dependent, and the product quality evaluation criteria are product-dependent. Their relevant metrics (correlation, importance and clarity) are introduced in Section 5.1 and Section 5.2.
- Risk. Risk is a concept that denotes the likelihood of the occurrence of an event [de Landa et al. 2003], whose primary properties we define as follows:
    - Nature of Risk. Nature of risk is the brief description of a risk posed to a project, including threats to it and its life-cycle.
    - Trigger. Trigger is an event that would incur a risk, which is expressed in terms of time events.
    - Current Status. Current status refers to the status of whether or not a risk has been avoided.
    - Cause for Changes in Status. Cause for changes in status is used to describe the cause of a revision to the status of a risk.
- Impact. Impact is the negative outcome of a risk, which needs to be assessed by employees. We define the key properties of an impact as follows:
    - Impact Level. Impact level is the class of an impact, such as critical, severe, moderate or limited.
    - Probability. Probability is the percentage indicating whether or not a risk is likely to occur.

  - Risk Exposure. Risk exposure is the product of the impact and the probability of a risk, which can be classified into unacceptable, critical, significant, minor or area of concern.

- Employee. An employee is a member of a project team. An employee has a role in a project team, such as Member of Project Board Chair (Project Executive, Senior User or Senior Supplier), Project Assurance Officer (Business Assurance Officer, Specialist Assurance Officer or User Assurance Officer), Project Manager, Team Manager, and so on. An employee is responsible for certain aspects of a project and may be responsible for stages and tasks in the project plan. S/he may also need to take part in the design of criteria for assessing the completion status of a task and the quality of a product. S/he may also be responsible for detecting, proving and reviewing a risk in the project, or assessing the impact of a risk, and so on.
- Stakeholder. A stakeholder is anyone who has an expectation regarding the outcome (benefit/functionality/products etc.) of a project. A stakeholder has the right to review the criteria designed for assessing project completion status and product quality.

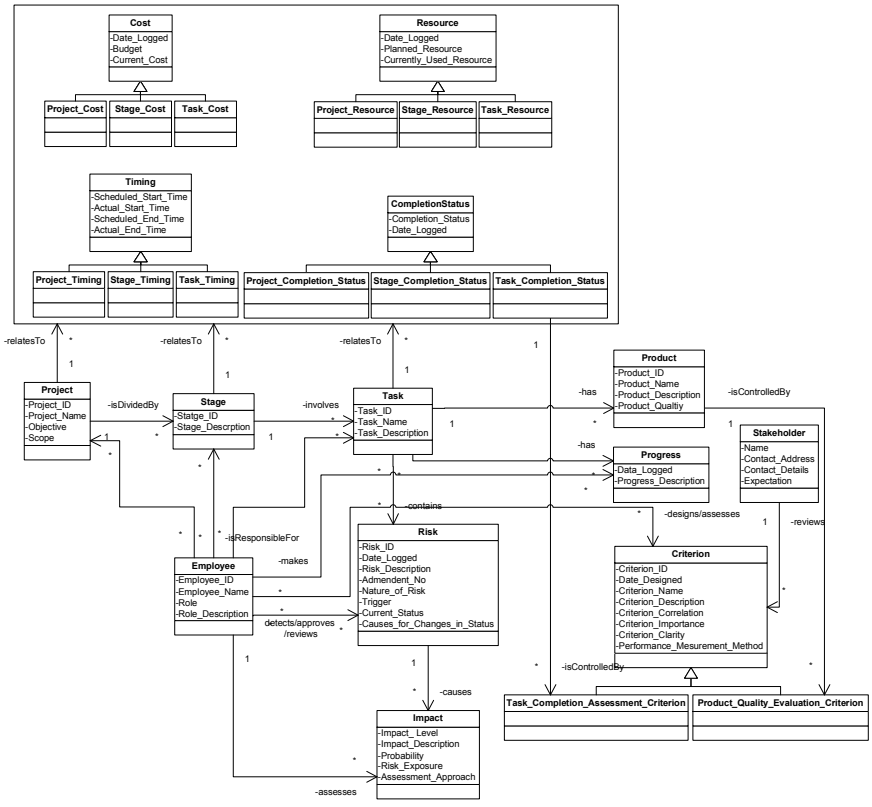

Figure 2: Abbreviated view of the project monitoring ontology

Subsequently, we present a project management ontology which provides a shared vocabulary for explaining the terms used in project monitoring processes. An example of the project management ontology can be seen in Figure 3. It can be found that the ontology consists of a group of project management concepts, where each concept represents a specific term utilized in the project management field. Each concept is defined by three properties as follows:

- Definition. Definition is a datatype property, which gives a detailed definition of a project management concept.
- Synonym. Synonym is a datatype property, which provides a group of terms semantically similar to a concept.
- RelatesTo. RelatesTo is an object property, which connects two relevant concepts.

The project management ontology can be used to provide formal definitions, synonyms and relevant terms for the terms used in project monitoring processes, in order to solve the ambiguity problem caused by cultural or project management software differences, which enhances the effectiveness and efficiency of knowledge sharing in project monitoring processes.

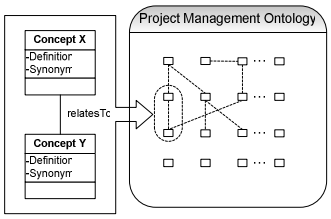

Figure 3: Example of the project management ontology

5 Assessment Metrics

In this section, we respectively present the metrics for assessing the project/stage/completion status and product quality, which are both extended from the theory of CCCI Metrics (Correlation of Interaction, Correlation of Criterion, Clarity of Criterion, and Importance of Criterion).

The theory of CCCI Metrics originates from Chang and Hussain's works [Chang et al. 2005a, Chang et al. 2005b, Hussain et al. 2004, Hussain et al. 2006], which is a methodology that allows service customers to quantitatively assess the trustworthiness of service providers in the field of logistics. The workflow of this theory includes:

1. A service customer and a service provider collaboratively design multiple mutually agreed criteria for assessing the trustworthiness of the service provider in a service transaction.
2. The service customer evaluates the service provider's performance on each criterion (0-6) in the service transaction, clarity of each criterion (0 or 1) and importance of each criterion (0-2).

3. The trustworthiness value (0-6) of the service provider in this service transaction can be obtained by aggregating the service customer's evaluation scores on all criteria.

It can be observed that there is a resemblance between the service providers' trustworthiness assessment and project completion status and product quality monitoring, which means that, on the one hand, employees in a project team need to design multiple criteria to monitor the completion status of tasks and the quality of products in a project; on the other hand, the criteria designed for a project must be in accordance with the stakeholders' expectations of the project outcome. Therefore, the criteria in a project can be regarded as having been mutually agreed upon by both project team members and project stakeholders. This establishes the foundation for applying the CCCI Metrics in the project monitoring process.

5.1 Project/Stage/Task Competition Status Assessment Metrics

As stated in Section 3.2, the criteria for assessing the task completion status are designed by responsible employees (e.g. project managers) and reviewed by stakeholders to achieve an agreement regarding the clarity and importance of each criterion at the beginning of a project. The project/stage/task completion status can be assessed by responsible employees (e.g. project managers) in the progress of the project whenever needed. In this section, we respectively introduce the project/stage/task completion status assessment metrics, which can be found below.

Completion status of a project. Since a project is divided into several stages, the completion status of a project relies on the completion status of its various stages. Therefore, we define the completion status of a project as the arithmetic mean of the completion status of its particular stages, which can be represented by Equation (1).

$$CompletionStatus_{\text{Project}} = \left\lfloor \frac{\sum_{k=1}^{l} CompletionStatus_{Stage_k}}{l} \right\rfloor \tag{1}$$

Completion status of a stage ($CompletionStatus_{Stage}$). Since a stage involves several tasks, the completion status of a stage depends on the completion status of its tasks. Hence, we define that the completion status of a stage is equal to the arithmetic means of the completion status of its involved tasks, which can be represented by Equation (2).

$$CompletionStatus_{Stage} = \left\lfloor \frac{\sum_{j=1}^{m} CompletionStatus_{Task_j}}{m} \right\rfloor \tag{2}$$

Completion status of a task. Here we use the theory of CCCI Metrics to measure the completion status of a task. Prior to the assessment, many specific criteria

need to be designed to measure the completion status of a task. In other words, once all criteria have been fulfilled, the task for which these criteria were designed is considered to have been completed.

The CCCI Metrics for the task completion status assessment contain four metrics which are:

Definition 1. Correlation of a task ($Corr_{Task}$)

$Corr_{Task}$ in the context of task completion assessment is defined as a metric that expresses the degree of comparison between the actual completion status of a task ($ActualCompetion_{Task}$) and the mutually agreed completion status of a task ($MutuallyAgreedCompletion_{Task}$) [Dong et al. 2007a, Dong et al. 2007b], which can be mathematically represented by Equation (3).

$$Corr_{Task} = \frac{ActualCompletion_{Task}}{MutuallyAgreedCompletion_{Task}} \tag{3}$$

Definition 2. Correlation of a criterion ($Corr_{Criterion}$)

$Corr_{Criterion}$ in the context of task completion assessment is defined as a metric that qualifies the actual completion extent of a task according to a given criterion [Dong et al. 2007a, Dong et al. 2007b]. The scope of the $Corr_{Criterion}$ has two levels as follows:

0 – None/ partially completed

1 – Fully competed

Definition 3. Clarity of a criterion ($Clear_{Criterion}$)

$Clear_{Criterion}$ in the context of task completion assessment is defined as a metric that qualifies the extent to which a criterion has been mutually agreed upon by criteria designers and stakeholders [Dong et al. 2007a, Dong et al. 2007b]. The $Clear_{Criterion}$ has two levels as follows:

0 – This criterion is not mutually agreed upon by both sides.

1 – This criterion is mutually agreed upon by both two sides.

Definition 4. Importance of a criterion ($Imp_{Criterion}$)

$Imp_{Criterion}$ is defined as a metric expresses the importance of a criterion [Dong et al. 2007a, Dong et al. 2007b]. The $Imp_{Criterion}$ has three levels as follows:

0 – Not important

1 – Important

2 – Very important

Thus, the equation for assessing the task completion status is as follows:

$$
\begin{aligned}
CompletionStatus_{Task} &= Corr_{Task} \\
&= \frac{ActualCompletion_{Task}}{MutuallyAgreedCompletion_{Task}} \\
&= \left\lfloor \frac{6 \times \sum_{i=1}^{n} Corr_{Criterion_i} \times Clear_{Criterion_i} \times Imp_{Criterion_i}}{\sum_{i=1}^{n} MACorr_{Criterion_i} \times Clear_{Criterion_i} \times Imp_{Criterion_i}} \right\rfloor
\end{aligned} \tag{4}
$$

where $MACorr_{Criterion}$ in the context of task completion assessment is defined as the mutually agreed completion status of a task according to a given criterion, and $MACorr_{Criterion} = 1$.

The project/stage/task completion status has seven levels as follows:

0 – Ignorance
1 – Uncompleted
2 – Mostly uncompleted
3 – Half completed
4 – Mostly completed
5 – Nearly completed
6 – Completed

5.2 Product Quality Evaluation Metrics

In this section, we use the CCCI Metrics in the field of product quality assessment. Analogous to the previous section, the criteria for assessing the quality of each product need to be designed in advance according to stakeholders' expectations. Stakeholders then need to review the criteria in order to reach an agreement regarding the clarity and importance of each criterion. Responsible employees (e.g. project managers) then evaluate the quality of each product based on the product-dependent criteria in the progress of the project. Similarly, the CCCI Metrics for the product quality evaluation include four metrics as follows:

Definition 5. Correlation of a product ($Corr_{Product}$)

$Corr_{Product}$ is defined as a metric that expresses the degree of comparison between the actual quality of a product ($ActualQuality_{Product}$) and the mutually agreed quality of a product ($MutuallyAgreedQuality_{Product}$) [Dong et al. 2008, Dong et al. 2009], which can be mathematically expressed in Equation (5).

$$
Corr_{\text{Product}} = \frac{ActualQuality_{\text{Product}}}{MutuallyAgreedQuality_{\text{Product}}} \tag{5}
$$

Definition 6. Correlation of a criterion ($Corr_{Criterion}$)

$Corr_{Criterion}$ in the context of product quality evaluation is defined as a metric that qualifies the extent of the examiners' satisfaction with the quality of a product based on a given criterion [Dong et al. 2008, Dong et al. 2009]. The $Corr_{Criterion}$ has seven levels as follows:

0 – Ignorance

1 – Very dissatisfied
2 – Dissatisfied
3 – Neutral
4 – Partially satisfied
5 – Satisfied
6 – Very satisfied

The definitions and scopes of the metrics for clarity of a criterion ($Clear_{Criterion}$) and importance of a criterion ($Imp_{Criterion}$) are the same as those for the context of task completion status assessment.

Therefore, the equation for assessing the quality of a product is given by:

$$\begin{aligned} Quality_{\text{Product}} &= Corr_{\text{Product}} \\ &= \frac{ActualQualtiy_{\text{Product}}}{MutuallyAgreedQuality_{\text{Product}}} \\ &= \left\lfloor \frac{6 \times \sum_{i=1}^{n} Corr_{Criterion_i} \times Clear_{Criterion_i} \times Imp_{Criterion_i}}{\sum_{i=1}^{n} MACorr_{Criterion_i} \times Clear_{Criterion_i} \times Imp_{Criterion_i}} \right\rfloor \end{aligned} \tag{6}$$

where $MACorr_{Criterion}$ in the context of product quality evaluation is defined as the mutually agreed quality of a product according to a given criterion, and $MACorr_{Criterion} = 1$.

The quality of a product has seven levels as follows:
0 – Ignorance or unknown
1 – Unacceptable quality
2 – Poor quality
3 – Average quality
4 – Good quality
5 – Very good quality
6 – Excellent quality

It can be observed that we configure both the value spaces for the project completion status level and the product quality level between 0 and 6, which allows users to easily and clearly learn about the current situation of projects.

6 System Implementation and Evaluation

In this section, we build a prototype of the proposed ORPMS framework and then conduct a functional testing in order to thoroughly assess the feasibility of the methodology.

First of all, the prototype implementation can be divided into three subtasks according to the architecture of the ORPMS, which are described as follows:

- The first subtask is to implement a Web portal for the knowledge sharing between geographically dispersed employees of project organizations. We

make use of Java, Java Serverlet, JavaServer Pages (JSP), JavaScript and Asynchronous JavaScript and XML (AJAX) to construct an ORPMS web portal, which can be seen from Figure 4 to Figure 7.

- The second subtask is to implement the project monitoring knowledge base, which primarily contains two ontologies – a project monitoring ontology and a project management ontology. We build the two ontologies by means of Protégé-OWL. It needs to be noted that building a project management ontology is an on-going process which needs to reference the terms used by multiple project management software and methodologies. Currently, the project management ontology is built by referencing the terms and definitions from the PMBOK [Project Management Institute 2004], which contains 230 concepts to date.
- The third subtask is to build the project monitoring database. We make use of MySQL to implement this subtask.

In design science, an approach for evaluating an artefact is functional testing, namely running the artefact to detect its defect or failure [Hevner et al. 2004]. In order to evaluate the ORPMS framework, apart from implementing a prototype, we need to test its major functions: 1) project/stage/task completion status monitoring, and 2) product quality monitoring.

Before we start the functional testing, we need to configure the authority levels of users for the ORPMS prototype. We assume that there are five authority levels for the ORPMS Web portal, which are:

- Project Board Chair. A project board chair is authorized to appoint a project manager and to approve the project plan at the beginning of a project. S/he also needs to approve any change of plan as the project progresses.
- Project Manager. A project manager is responsible for entering the whole project plan into the ORPMS, including information regarding objectives, scope, budget, resources, schedule, products, and criteria for task completion assessment and product quality evaluation etc... S/he also needs to appoint team managers and configure their roles and responsibilities. S/he needs to approve the risks identified by team managers and s/he can also change the project plan to avoid risks as the project progresses. S/he needs to assess the quality of a product when a product is generated. The project manager also has the authority to assess the completion status of tasks. S/he is responsible for monitoring the actual cost, timing and resources of the project.
- Team manager. A team manager is authorized to appoint his/her team members at the beginning of a project. As the project progresses, s/he needs to regularly (e.g. weekly) submit the progress reports of his/her responsible tasks and log detected risks to the ORPMS. S/he also needs to report to the project manager when a product is generated from the tasks for which s/he is responsible.
- Project Assurance Officer. A project assurance officer is responsible for analysing the nature, trigger, current status and reasons for changes in status of detected risks. S/he also needs to assess the impact of risks and enter the information about risks and their impact into the ORPMS. S/he needs to review risks once a change has been executed.

- Stakeholder. A stakeholder is authorized to submit his/her expectation of the project into the ORPMS and to evaluate the clarity and importance of the criteria for task completion status assessment and for product quality evaluation.

Subsequently, we execute a case study in the Web portal in order to test the function of project/stage/task completion status monitoring. Figure 4 shows a scenario of the project/stage/task completion status assessment. In this scenario, a project manager (User ID: MCC0027) logs in to the ORPMS Web portal and s/he needs to know the completion status of a task (Task 2). To this end, s/he retrieves the task by SPARQL, and then starts the assessment. The task completion status assessment interface is then displayed in the Web portal (Fig. 4). It can be found that there are four criteria (Criterion 8 to Criterion 11) for assessing the progress of the task. The detailed information regarding these criteria can be observed by clicking their links. The project manager can click the link "*Progress Report*" to check the latest progress of the task entered by a responsible team manager. Based on the latest task progress report, the project manager can assess the completion status of the task. The assessment is to assign the $Corr_{Criterion}$ value (0 or 1) for each criterion, which is determined by whether or not the progress record fulfils a criterion. The $Clear_{Criterion}$ value (0 or 1) and $Imp_{Criterion}$ value (0-2) of each criterion have already been decided by stakeholders. Here the project manager determines the $Corr_{Criterion}$ values for all the criteria as "1". Therefore, according to Equation (4), the completion status of Task 2 can be calculated by

$$\begin{aligned} CompletionStatus_{Task_2} &= \left\lfloor \frac{6\times(Corr_{Criterion_8}\times Clear_{Criterion_8}\times Imp_{Criterion_8}+...)}{MACorr_{Criterion_8}\times Clear_{Criterion_8}\times Imp_{Criterion_8}+...} \right\rfloor \\ &= \left\lfloor \frac{6\times(1\times1\times2+1\times1\times2+1\times1\times2+1\times1\times2)}{1\times1\times2+1\times1\times2+1\times1\times2+1\times1\times2} \right\rfloor \\ &= 6 \end{aligned}$$

Therefore, the completion status of Task 2 is "*Completed*" according to the task completion status level. Figure 5 shows the completion status of the task and the updated completion status of its belonged stage (Stage A) and project (Project A). According to Equation (2), the completion status of Stage A can be calculated by

$$CompletionStatus_{Stage_A} = \left\lfloor \frac{4+6}{2} \right\rfloor = 5$$

Consequently, the completion status of Task A is "*Nearly Completed*" according to the stage completion status level. According to Equation (1), the completion status of Project A can be calculated by

$$CompletionStatus_{Project_A} = \left\lfloor \frac{5+0}{2} \right\rfloor = 2$$

Hence, the completion status of Project A is "*Mostly Uncompleted*" according to the project completion status level.

The second test is to validate the function of product quality monitoring. When a project manager receives notice of the generation of a product (Product A) from a team manager, s/he needs to retrieve the product from the ORPMS Web portal and then start the evaluation. The product evaluation interface is then shown in the Web portal (Fig. 6). The project manager can evaluate the quality of the product based on the criteria (Criterion 0 to Criterion 3) mutually agreed upon by the project manager and stakeholders. The evaluation is accomplished by assigning the $Corr_{Criterion}$ value (0-6) to each criterion, which is based on the extent to which this product fulfils a criterion. Moreover, the project manager needs to upload a product quality evaluation report to the system in order to explain the reasons for these evaluation values. Similar to the project/stage/task completion status assessment, the $Clear_{Criterion}$ value (0 or 1) and $Imp_{Criterion}$ value (0-2) of each criterion have been decided by the stakeholders. After clicking the "*Evaluation*" button, the evaluation result is submitted. It needs to be noted that usually each product can be evaluated only once, and any amendments to a product can be regarded as a new product, different from the project/stage/task completion status assessment.

The product quality evaluation result is shown in Figure 7. According to the Equation (6), the quality of Product A can be calculated by

$$\begin{aligned} Quality_{Product_A} &= \left\lfloor \frac{Corr_{Criterion_0} \times Clear_{Criterion_0} \times Imp_{Criterion_0} + ...)}{MACorr_{Criterion_0} \times Clear_{Criterion_0} \times Imp_{Criterion_0} + ...} \right\rfloor \\ &= \left\lfloor \frac{3\times1\times2+5\times1\times2+4\times1\times2+6\times1\times2}{1\times1\times2+1\times1\times2+1\times1\times2+1\times1\times2} \right\rfloor \\ &= 4 \end{aligned}$$

Hence, the quality of Product A is evaluated as "*Good Quality*" according to the product quality level.

The above two experiments show us the basic functions and workflows of the project/stage/task completion status monitoring and the product quality monitoring. In terms of the demonstration and explanation, it can be observed that the proposed functions are basically realized in the case study. Therefore, it can be deduced that we preliminarily prove the primary functions of our ORPMS framework by means of the functional testing.

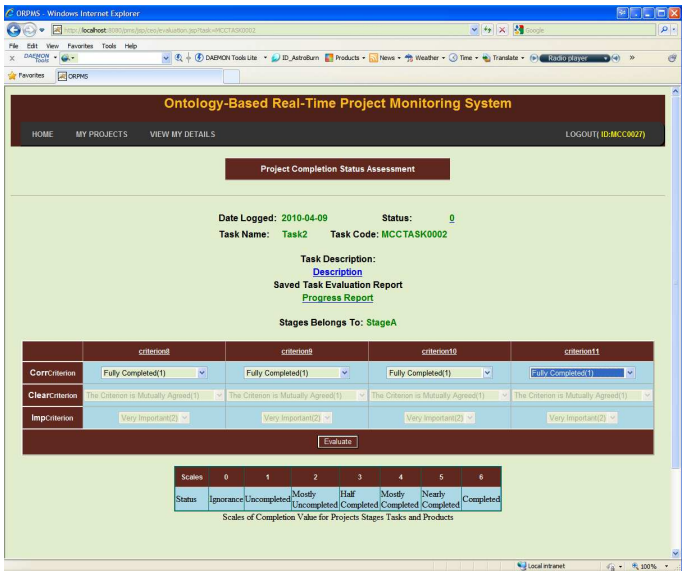

Figure 4: Screenshot of the task completion status assessment interface in the ORPMS web portal

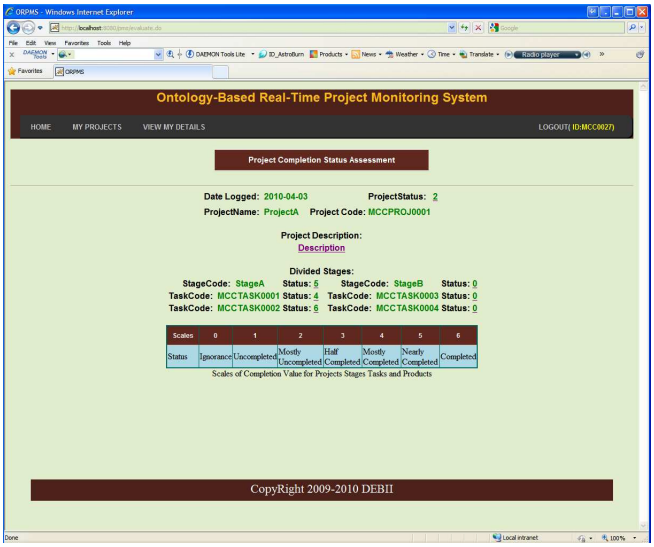

Figure 5: Screenshot of the task completion status assessment result in the ORPMS web portal

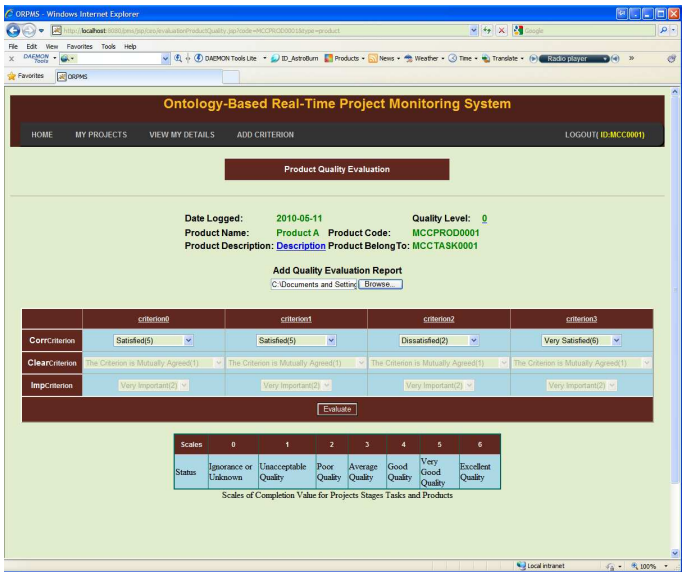

Figure 6: Screenshot of the product quality evaluation interface in the ORPMS web portal

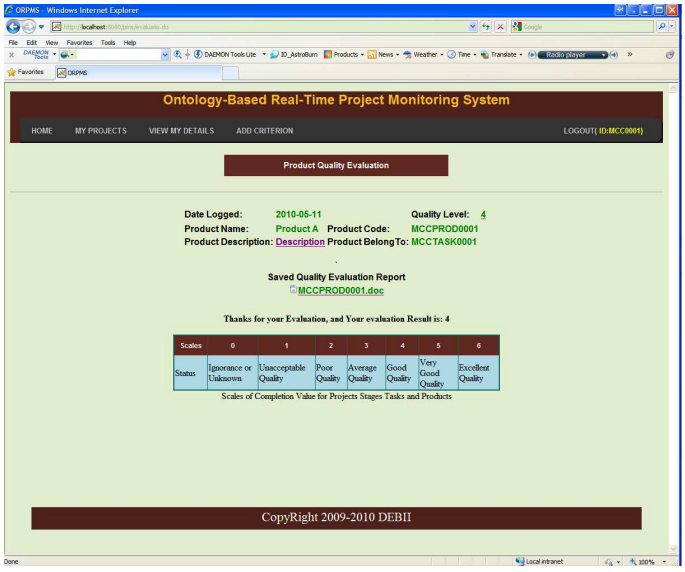

Figure 7: Screenshot of the product quality evaluation result in the ORPMS web portal

7 Conclusion and Future Work

In this paper, we present an ontology-based real-time project monitoring approach, in order to address the issue of the ambiguity in knowledge sharing in project monitoring processes and the issue of the omission of project monitoring from contemporary ontology-based project management approaches. The core components of the proposed ORPMS framework include:

1. A project monitoring ontology which represents project monitoring knowledge and facilitates knowledge capture, storage and sharing in project monitoring processes;
2. A project management ontology which represents project management knowledge and provides a shared vocabulary to disambiguate terms used in project monitoring processes;
3. A set of project/stage/task completion status assessment metrics which allows management of a project to timely learn about the latest progress of a project;
4. A set of product quality evaluation metrics which allows management of a project to timely learn about the quality of products generated by the project.

In order to validate the feasibility of the ORPMS framework, we use the functional testing approach to build a prototype – an ORPMS Web portal, and conduct a case study based on the prototype. The testing result preliminarily proves the feasibility of the proposed ontology-based project monitoring approach.

Our future research work will focus on the following areas:

1. We will install the Web portal in a real cloud environment and test its scalability.
2. We will test our Web portal in a real project organization by surveying users' satisfaction.
3. We will extend the vocabulary of the project management ontology by referencing more project management documents.
4. We will try to incorporate the project management ontology with existing commercial project monitoring software in order to solve the ambiguity issue in the project data exchange between different project monitoring software.

Acknowledgements

We would like to express our appreciation to our web designer – Wei Liu who is responsible for the ORPMS Web portal implementation. We also would like to express our gratitude to any anonymous reviewers of our manuscript for their valuable suggestions to improve this paper.

References

[Abels et al. 2006] Abels, S., Ahlemann, F., Hahn, A., Hausmann, K. and Strickmann, J.: "PROMONT – A project management ontology as a reference for virtual project organizations"; On the Move to Meaningful Internet Systems 2006: OTM 2006 Workshops, Springer, Berlin, Heidelberg (2006), 813-823.

[Aramo-Immonen 2009] Aramo-Immonen, H.: "Project management ontology - The organizational learning perspective"; Doctor's thesis, Tampere University of Technology, Tampere, Finland (2009).

[Armbrust et al. 2009] Armbrust, M., Fox, A., Griffith, R., Joseph, A.D., Katz, R., Konwinski, A., Lee, G., Patterson, D., Rabkin, A., Stoica, I., and M. Zaharia: "Above the clouds: A berkeley view of cloud computing"; UC Berkeley Reliable Adaptive Distributed Systems Laboratory, Berkeley, CA (2009).

[Bhatt et al. 2004a] Bhatt, M., Wouters, C., Flahive, A., Rahayu, J. W. and Taniar, D.: "Semantic completeness in sub-ontology extraction using distributed methods"; Computational Science and Its Applications - ICCSA 2004, Springer-Verlag, Berlin, Germany (2004a), 508-517.

[Bhatt et al. 2006] Bhatt, M., Flahive, A., Wouters, C., Rahayu J. W. and Taniar, D.: "MOVE: A distributed framework for materialized ontology view extraction"; Algorithmica, 45, 3 (2006), 457-481.

[Bhatt et al. 2004b] Bhatt, M., Flahive, A., Wouters, C., Rahayu, J. W., Taniar , D., and Dillon, T. S.: "A Distributed Approach to Sub-Ontology Extraction"; The 18th International Conference on Advanced Information Networking and Applications (AINA 2004), IEEE Computer Society, (2004b), 636-641.

[Bullinger et al. 2005] Bullinger, H.-J., Warschat, J., Schumacher, O., Slama, A., and Ohlhausen, P.: "Ontology-based project management for acceleration of innovation projects"; From Integrated Publication and Information Systems to Information and Knowledge Environments, Berlin, Heidelberg, Springer (2005), 280-288.

[Chang et al. 2005a] Chang, E., Dillon, T. S,. and Hussain, F.: "Trust and Reputation for Service Oriented Environments-Technologies for Building Business Intelligence and Consumer Confidence"; John Wiley & Sons, West Sussex, England (2005a).

[Chang et al. 2005b] Chang, E., Hussain, F. K., and Dillon, T. S.: "CCCI Metrics for the measurement of quality of e-service"; The 2005 IEEE/WIC/ACM International Conference on Intelligent Agent Technology (IAT'05), IEEE Computer Society, Compiegne, France (2005b).

[Cheah and Chang 2005] Cheah, C. and Chang, E.: "Multi-site project management ontology system development methodology"; The 2005 3rd IEEE International Conference on Industrial Informatics (INDIN 2005), IEEE, Perth, Australia (2005), 836-840.

[Chinowsky and Goodman 1996] Chinowsky, P. S. and Goodman, R. E.: "Managing interdisciplinary project teams through the Web"; Journal of Universal Computer Science, 2, 9 (1996), 597-609.

[Clark 2005] Clark, D.: "Position paper for rules concerning project management ontologies"; W3C Workshop on Rule Languages for Interoperability, W3C, Washington, D.C., USA (2005).

[Crawford and Bryce 2003] Crawford, P. and Bryce, P.: "Project monitoring and evaluation: a method for enhancing the efficiency and effectiveness of aid project implementation"; International Journal of Project Management, 21, 5 (2003), 363-373.

[de Landa et al. 2003] de Landa, F. L., Travassos, G. H. and Rocha, A. R.: "Managing organizational risk knowledge"; Journal of Universal Computer Science, 9, 7 (2003), 670-681.

[Dong et al. 2007a] Dong, H., Hussain, F. K. and Chang, E.: "Project track and trace ontology"; The Second International Conference on Internet and Web Applications and Services (ICIW '07), IEEE Computer Society, Le Morne, Mauritius (2007a), 52-52.

[Dong et al. 2007b] Dong, H., Hussain, F. K. and Chang, E.: "Application of Protege and SPARQL in the field of project knowledge management"; The Second International Conference on Systems and Networks Communications (ICSNC '07), IEEE Computer Society, Cap Esterel, France (2007b), 74-74.

[Dong et al. 2008] Dong, H., Hussain, F. K. and Chang, E.: "Quality of service (QoS) based service retrieval engine"; The 6th International Conference on Advances in Mobile Computing and Multimedia (MoMM '08), ACM, Linz, Austria (2008), 405-408.

[Dong et al. 2009] Dong, H., Hussain, F. K. and Chang, E.: "A QoS-based service retrieval methodology for digital ecosystems"; International Journal of Web and Grid Services, 5, 3 (2009), 261-283.

[Dustdar 2005] Dustdar, S.: "Reconciling knowledge management and workflow management systems: The activity-based knowledge management approach"; Journal of Universal Computer Science, 11, 4 (2005), 589-604.

[Esposito et al. 2008] Esposito, A., Tarricone, L., Vallone, L. and Zappatore, M.: "A proposal for an electromagnetic ontology framework"; International Journal of Web and Grid Services, 4, 3 (2008), 284-302.

[Flahive et al. 2005] Flahive, A., Rahayu, J. W., Taniar, D. and Apduhan, B. O.: "A distributed ontology framework in the Semantic Grid environment"; The 19th International Conference on Advanced Information Networking and Applications (AINA 2005), IEEE Computer Society, Taipei, Taiwan (2005), 193-196.

[Flahive et al. 2009] Flahive, A., Taniar, D., Rahayu, J. W. and Apduhan, B. O.: "Ontology tailoring in the Semantic Grid"; Computer Standards & Interfaces, 31, 5 (2009), 870-885.

[Gruber 1995] Gruber, T.: "A translation approach to portable ontology specifications"; Knowledge Acquisition, 5, 2 (1995), 199-220.

[Hevner et al. 2004] Hevner, A. R., March, S. T., Park, J. and Ram, S.: "Design science in information systems research"; MIS Quarterly 28, 1 (2004), 75-105.

[Hussain et al. 2004] Hussain, F. K., Chang, E. and Dillon, T. S.: "Trustworthiness and CCCI metrics in P2P communication"; International Journal of Computer Systems Science and Engineering 19, 3 (2004), 173-190.

[Hussain et al. 2006] Hussain, F. K., Chang, E. and Dillon, T. S.: "Trust issues in Service Oriented Environment"; SOLI 2006, IEEE, Shanghai, China (2006), 790-793.

[Kerzner 2009] Kerzner, H.: "Project management: A systems approach to planning, scheduling, and controlling "; John Wiley & Sons, New Jersey, USA (2009), xxi-xxiv.

[Martin et al. 2008] Martin, B., Mitrovic, A. and Suraweera, P.: "ITS domain modelling with ontology"; Journal of Universal Computer Science, 14, 17 (2008), 2758-2776.

[Martino 2009] Martino, B. D.: "Semantic web services discovery based on structural ontology matching."; International Journal of Web and Grid Services, 5, 1 (2009), 46-65.

[Pich et al. 2002] Pich, M. T., Loch, C. H., and d. Meyer, A.: "On uncertainty, ambiguity, and complexity in project management "; Management Science, 48, 8 (2002), 1008-1023.

[Project Management Institute 2004]: "A guide to the project management body of knowledge: PMBOK guide"; Project Management Institute, Newtown Square, Pennsylvania, (2004).

[Vega-Gorgojo et al. 2008] Vega-Gorgojo, G., Bote-Lorenzo, M. L., Gómez-Sánchez, E., Asensio-Pérez, J. I., Dimitriadis, Y. A. and Jorrín-Abellán, I. M.: "Ontoolcole: Supporting educators in the semantic search of CSCL tools"; Journal of Universal Computer Science, 14, 1 (2008), 27-58.