

©2007 IEEE. Personal use of this material is permitted. However, permission to reprint/republish this material for advertising or promotional purposes or for creating new collective works for resale or redistribution to servers or lists, or to reuse any copyrighted component of this work in other works must be obtained from the IEEE

# Project Track and Trace Ontology

Hai Dong  
Curtin University of  
Technology

Farookh Khadeer Hussain  
Curtin University of  
Technology

Elizabeth Chang  
Curtin University of  
Technology

## Abstract

*It is well-known that ontology is utilized as an effective methodology to share domain-specific knowledge in multidisciplinary fields. In the field of project management, due to the characteristic of project organizations in which project members are geographically dispersed and from different cultural background, senior management would feel difficulty when they attempt to know about the detailed project completion status from dispersed project groups. Thus, the objective of this paper is to propose an automated project track and trace methodology through the use of ontology technology, to challenge the knowledge sharing issues in project organizations. By means of extending CCCI Metrics into the field of project management and introducing a new ontological notation system, we deliver the project track and trace ontology.*

## 1. Introduction

It is well-known that ontology is utilized as an effective methodology to share domain-specific knowledge in multidisciplinary fields [1] [4]. In the field of project management, one characteristic of project organizations is that people in the organizations are geographically dispersed [3]. With the increase of project outsourcing, project groups and its members are probably located in different areas, from different cultural background and even speaking with different languages [2]. These issues challenge the administration of senior management on project completion status. In addition, until now there is not an existing methodology for tracking and tracing project procedures in project organizations.

Against the above issues, this paper is to propose project track and trace methodology by means of ontology, due to the advantage of ontology which is helpful to share knowledge [4]. From Elizabeth, Dillon and Hussain's works, we observe that CCCI Metrics is

a proper methodology to measure the project completion status. In addition, we introduce a notation system instead of UML (Unified Modeling Language) to represent our ontological model. Finally we present the ontology to realize the function of automated project track and trace.

## 2. Utilizing CCCI Metrics for project track and trace

CCCI Metrics originates from the works of Chang et al., which is a quantitative methodology to assess trustworthiness of logistic service providers [1]. The essence of CCCI Metrics theory is to measure the trustworthiness value of the service providers by means of designing various criteria for the providers and quantitatively evaluating the correlation, the clarity and the importance of each criterion. In this paper we extend the CCCI Metrics and apply it to the field of project management to enable ontology based knowledge sharing.

CCCI Metrics for project track and trace is utilized to measure the completion status of a project. A project is viewed as being composed of many criteria. Each status or the completion status of each criterion is individually tracked in order to determine the status of the project. In other words, once all criteria have been completed, the project in turn is regarded as complete.

CCCI Metrics for project track and trace comprise four metrics as shown below:

**Correlation of a project ( $Corr_{Project}$ )** – Degree of Comparison between the actual status of the project completion ( $ActualCompletion_{Project}$ ) and the mutually agreed status of the project completion ( $MutuallyAgreedCompletion_{Project}$ ) (1).

$$Corr_{Project} = \frac{ActualCompletion_{Project}}{MutuallyAgreedCompletion_{Project}} \quad (1)$$

**Correlation of a criterion ( $Corr_{Criterion}$ )** – A metric qualifies the extent of criterion completion in a project.

Extent: 0 – None/ Partially Completed  
 1 – Fully Completed –  $\text{MaxCorr}_{\text{Criterion}}$

**Clarity of a criterion ( $\text{Clear}_{\text{Criterion}}$ )** – A metric qualifies the extent whether a criterion is mutually agreed between the evaluating person and the evaluated person or not.

Extent: 0 – This criterion is not mutually agreed between two sides.

1 – This criterion is mutually agreed between two sides.

**Importance of a criterion ( $\text{Imp}_{\text{Criterion}}$ )** – A metric expresses the importance of a criterion.

Extent: 0 – Not important

1 – Important

2 – Very important

Thus, the equation of project completion status is drawn as (2).

$$\text{Project Completion Status} = \text{Corr}_{\text{Project}} \quad (2)$$

$$\begin{aligned} &= \frac{\text{ActualCompletion}_{\text{Project}}}{\text{MutuallyAgreedCompletion}_{\text{Project}}} \\ &= \frac{(\text{Corr}_{\text{Criterion1}} \times \text{Clear}_{\text{Criterion1}} \times \text{Imp}_{\text{Criterion1}} + \dots}{\text{MaxCorr}_{\text{Criterion1}} \times \text{Clear}_{\text{Criterion1}} \times \text{Imp}_{\text{Criterion1}} + \dots} \\ &\quad \dots + \text{Corr}_{\text{CriterionN}} \times \text{Clear}_{\text{CriterionN}} \times \text{Imp}_{\text{CriterionN}}) \times 6 \\ &\quad \dots + \text{MaxCorr}_{\text{CriterionN}} \times \text{Clear}_{\text{CriterionN}} \times \text{Imp}_{\text{CriterionN}} \end{aligned}$$

The scope of project completion status includes:

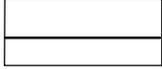
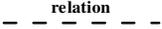
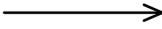
- 0 – Ignorance
- 1 – Completely unfinished
- 2 – Unfinished
- 3 – Minimally Finished
- 4 – Partially Finished
- 5 – Finished
- 6 – Completely finished

### 3. Notation system for ontology representation

Before we represent the ontology model for project track and trace, a notation system is introduced in this section. The notation system utilized in the ontological representation is based on Chang et al. [1]’s work, which consists of three basic notations as Table 1. Although in the past we usually used to employ UML to represent ontology model, due to its complex symbols categories, UML cannot efficiently help people better understand the shared knowledge. This notation system simplifies the symbols and its symbols

are closer to the principle of ontology which is the combination of shared concepts and relationships between concepts [4].

**Table 1. Ontology notation system**

Ontology Notation	Semantics of the Notation
	Double-field Box represents the Ontological Concepts.
	A dotted line represents Ontology Concept Association Relation which represents a Concept is closely related to another concept. The relationship name can be noted above the dotted line.
	Open-arrow line represents Composition and Aggregation or Part-of relationship between Upper Ontology Concept and Lower Ontology Concept.

### 4. Hierarchy of project organization domain concepts

In a Project Organization Domain, the Project Organization concept can be seen as a combination of Employee concept and Project concept. Employee also consists of:

**CEO** who is responsible for managing all projects in Project Organization.

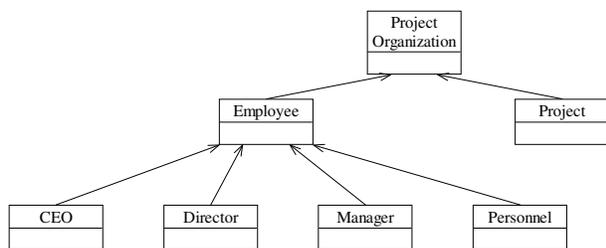
**Director** who is responsible for managing the projects which belong to his/her department in the Project Organization.

**Manager** who is responsible for managing the projects which belong to his/her division in each department.

**Personnel** who are responsible for the implementation of arranged projects.

On the other hand, according to the theory of CCCI Metrics, Project is divided into different Criteria which are in correspondence with tasks involved in the Project.

The graphical view of hierarchy of project organization domain concepts is shown in Fig. 1 through the use of the ontology notation.

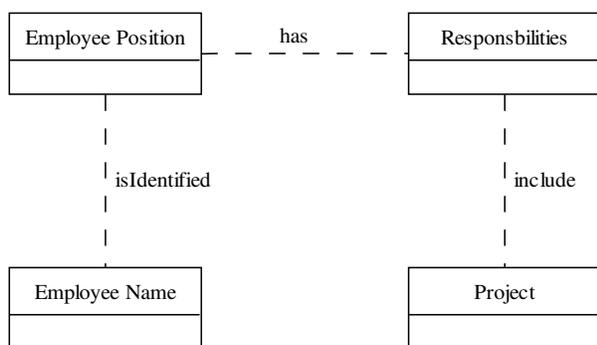


**Figure 1. Project organization domain concepts hierarchy**

## 5. The ontology of employee and its sub-compositions

### 5.1. Employee ontology

In a project organization, the Employee Ontology is defined as the conceptualization of the Employee who has an Employee Position in the organization and is identified by an Employee Name as well as has Responsibilities which include some Projects. (Fig. 2)



**Figure 2. Employee ontology**

We present the Employee Ontology as the combination of the ontology name and a tuple where the elements of the tuple can be complex elements as defined below:

Employee [Employee Position, Employee Name and Responsibilities] where:

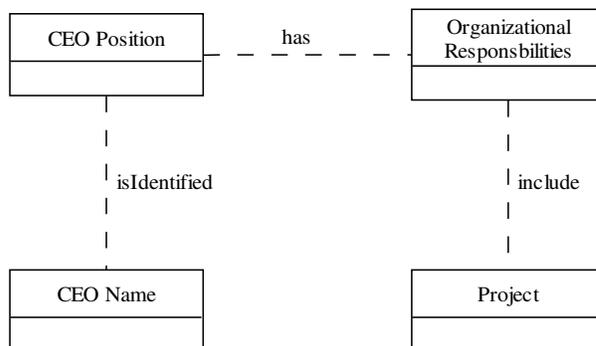
‘**Employee Position**’ is a unique identification of Employee in a project organization.

‘**Employee Name**’ is a unique identification of Employee Position in a project organization.

‘**Responsibilities**’ is an aggregation of Projects which Employees should take part in. Different Employee Positions are in correspondence with different Responsibilities.

### 5.2 CEO ontology and other employee’s Lower-level ontologies

In project organization environments, the CEO Ontology is defined as the conceptualization of the CEO who has a CEO Position in the organization and is identified by a CEO Name as well as has Organizational Responsibilities which include all Projects in the organization. (Fig. 3)



**Figure 3. CEO ontology**

We present the CEO Ontology as the combination of the ontology name and a tuple where the elements of the tuple can be complex elements as defined below:

CEO [CEO Position, CEO Name and Organizational Responsibilities] where:

‘**CEO Position**’ is a unique identification of CEO in a project organization.

‘**CEO Name**’ is a unique identification of CEO Position in a project organization.

‘**Organizational Responsibilities**’ is an aggregation of all Projects involved in a project organization that a CEO manages.

The other three sub-compositions of Employee Ontology – Director Ontology, Manager Ontology and Personnel Ontology inherit all the relations from Employee Ontology and the only difference is the scopes of the inherited concepts’ properties.

### 5.3 The relationships between employees

In a project organization, a well-conditioned management structure is beneficial to task distribution and progress evaluation. Here the management structure namely relationships between Employees are described to clarify the management structure in project organizations. (Fig. 4)



**Figure 4. Employee relationships**

In project organizations, the CEO manages all directors in the project organization. Then each director supervises at least one given manager and every manager manages at least one given personnel. On the other hand, except for CEO who is not managed by anyone, each member in the project organization has been administrated by the only one.

Owing to the differences of management scopes to different level of Employee, the associations are distinct, which are:

**CEO's management scope** is limited in all directors in the Project Organization domain.

**Directors' management scope** is the given managers in their departments.

**Managers' management scope** is the given Personnel in their divisions.

## 6. Project ontology and criterion ontology

### 6.1 Project ontology

In a project organization, the Project Ontology is defined as the conceptualization of the concept of Project that is identified by Project Code, is shown Date Started, is responsible to Employee and is evaluated by Project Status. (Fig. 5)

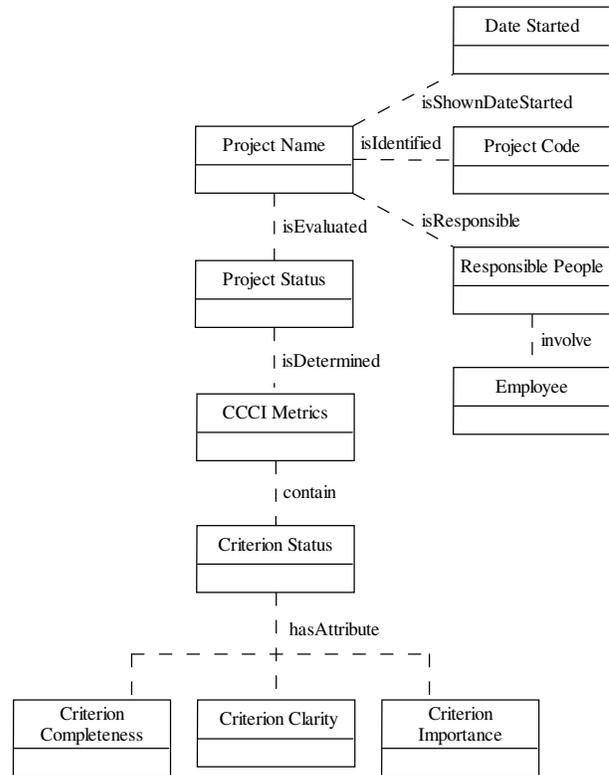


Figure 5. Project ontology

We represent the Project Ontology as the combination of the ontology name and a tuple where the elements of the tuple can be complex elements as defined below:

Project [Project Name, Project Code, Date Started, Responsible People, Project Status and CCCI Metrics] where:

**'Project Name'** usually refers to a Project itself. In project organization environments, a Project Name is seen as a unique identification for Project.

**'Project Code'** is the mixture of numerical symbols and alphabetic symbols, which also can be seen as the unique identification for Project. The use of Project Code mainly focuses on the storage of Projects' records in databases, which is beneficial to the pick-up and the storage of Projects' documentations.

**'Date Started'** refers to the date when a Project begins to implement. In project track and trace, Date Started can be utilized as a means to measure the length of a Project period which can be evaluated as an important quality aspect and a Criterion of Project.

**'Responsible People'** is an aggregation of Employees who are relevant to a Project.

**'Project Status'** can be substituted as the concept of Project Status Value. Based on the theory of CCCI

Metrics, the scope of Project Status Value is from 0 to 6, which means the different level of Project Status.

## 6.2 Criterion ontology

In project organization environments, the Criterion Ontology is defined as the conceptualization of the concept of Criterion that is identified by Criterion No., is shown Date Logged, is responsible to Employee and is determined by Criterion Status which has the attributes of Criterion Completeness, Criterion Clarity and Criterion Importance. (Fig. 6)

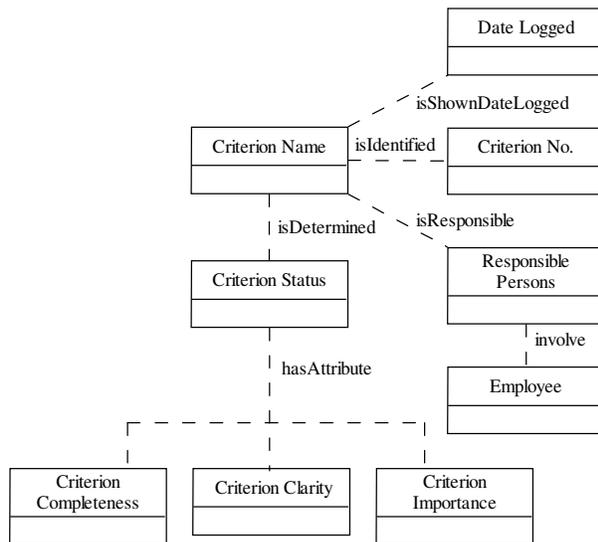


Figure 6. Criterion ontology

We represent the Criterion Ontology as the combination of the ontology name and a tuple where the elements of the tuple can be complex elements as defined below:

Criterion [Criterion Name, Criterion No., Date Logged, Responsible Persons, Criterion Status, Criterion Completeness, Criterion Clarity and Criterion Importance] where:

‘**Criterion Name**’ usually refers to a Criterion itself. In project organization environments, a Criterion Name is seen as a unique identification for Criterion.

‘**Criterion No.**’ is the mixture of numerical symbols and alphabetic symbols, which also can be seen as the unique identification for Criterion. The use of Criterion No. mainly focuses on the storage of Criteria’s records in databases, which is beneficial to the pick-up and the storage of Criteria’s documentations.

‘**Date logged**’ refers to the date when a criterion has been mutually agreed between an evaluating person and an evaluated person.

‘**Responsible Persons**’ is an aggregation of Employees who are relevant to a Criterion.

‘**Criterion Status**’ is a sub-tuple of the Criterion tuple, which uses quantitative means to determine the extent to which a criterion has been completed or delivered up on the mutually agreed Criterion. It consists of three elements – Criterion Completeness, Criterion Clarity and Criterion Importance.

‘**Criterion Completeness**’ is an element of Criterion Status, which qualifies the extent of task completion according to its corresponding Criterion.

‘**Criterion Clarity**’ is an element of Criterion Status, which qualifies the extent whether a Criteria is mutually agreed between an evaluating person and an evaluated person or not in a Project. Its scope is as below:

‘**Criterion Importance**’ is an element of Criterion Status, which expresses the importance of a Criterion in a Project.

## 6.3 The relationship between project and criterion

As explained earlier, a Project can be divided into several Criteria which are in correspondence with tasks or quality aspects of the Project (Fig. 7).

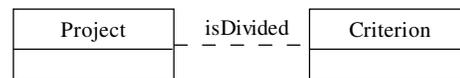


Figure 7. Relationship between project and criterion

## 7. Conclusion and future works

In this paper, against the issues in project track and trace, we propose an ontology-based methodology to assist senior management to better understand the current status of the projects under their administration, and for the further objective – to promote the automated and simplified project track and trace in project organisations. By means of extending the theory of CCCI Metrics into the field of project management, we attempt to adapt the quantitative methodology to evaluate the completion level of projects. Finally we borrow the ontology notation system from Chang, Dillon and Hussain’s works to create the project track and trace ontology.

The benefits of this project are concluded as below:

It realizes the function of managing the project status from the perspective of project management, which is to promote knowledge sharing between senior management and actual executors.

It can be utilized to distinctly define the tasks of each member in projects, and thus avoiding the confusion of members' understanding to own responsibilities.

It can be utilized to distinctly define completion criteria for each task, the importance and the clarity of each criterion, which is efficient to assist members fully understand their responsibilities.

It adopts quantitative methodology to measure the project completion status, which is effortlessly understood by organizational management.

The limitations of the project are concluded as below:

The ontology is not tested in practice, and thus we cannot validate its actual contribution to knowledge sharing activities in project organizations.

On account of the limitation of the time, we have not designed the API to guide users to use and test this system, which could be proposed in the future.

Therefore, in the future works, we will design the user interfaces by Java Language and implement the ontology-based system in client/server networks or peer-to-peer networks in project organizations and we will survey users' satisfaction status to evaluate the

system. In addition, we will attempt to expand our research scope to other project management activities in project organizations.

## 8. References

- [1] E. Chang, T.S. Dillon and F.K. Hussain, *Trust and Reputation for Service Oriented Environments-Technologies for Building Business Intelligence and Consumer Confidence*, John Wiley & Sons, Brisbane, 2005.
- [2] J. J. J. Kasvi, M. Vartiainen and M. Hailikari, "Managing knowledge and knowledge competences in projects and project organizations," *International Journal of Project Management*, New York, 2003, pp. 571-582.
- [3] I. Ruuska and M. Vartiainen, "Characteristics of knowledge sharing communities in project organizations," *International Journal of Project Management*, New York, 2005, pp.374-379.
- [4] N. Guarino, "Ontology-Driven Conceptual Modeling," presented at *21st International Conference on Conceptual Modeling*, Tampere, 2002, p.10.