

MV-TMM: A Multi View Traceability Management Method

Abstract— The approach presented in this work aims to guide the companies in their design of requirements traceability models adapted to the context of their projects. This is achieved by allowing the construction of a model based on trace fragments adapted to each phase of the development process or to a specific situation. Furthermore, the approach guides the users to use the traceability model in a requirement management tool. And help them capture and manage the evolution of the traceability data.

Index Terms—Requirement traceability, Requirement Management, Multi perspectives Modeling, Method, Meta model, Process, Project.

I. INTRODUCTION

Requirements traceability (RT) helps organisations to control and manage the evolution of the system requirement within a project. It's a requirements management activity where requirements are traced back to their original higher-level requirement sources. This ensures that all higher-level requirements are being met by detailed requirements. It also ensures that lower-level requirements have a higher-level source and have not been arbitrarily added to the scope of the project.

Gotel and Finkelstein [1] express the concept in a more complete way: RT refers to the ability to describe and follow the life of a requirement, in both a forwards and backwards direction (i.e., from its origins, through its development and specification, to its subsequent deployment and use, and through all periods of on-going refinement and iteration in any of these phases.)

This definition has become the common definition of RT. They make explicit that in order to follow (i.e., to trace) the life of a requirement you have to describe it. Two important aspects of requirements tracing are used to extend their definition. The first one is the ability to capture the traces we want to follow, and the other is the idea that traces should be viewed as natural occurrences, naturally produced.

Previous research work on RT focus on different aspects of this domain [2] [1] [3] [4] [5]. They have been classified on two categories [1] pre-traceability and post-traceability. Pre-traceability refers to those aspects of a requirement's life before it is included in the requirements specification and is focused on enabling a better understanding of the requirement. Post-traceability, on the other hand refers to those aspects of a requirement's life from the point in time when it has been included in the requirements specification and is focused on enabling a better understanding and acceptance of the current

system/software.

The proposed method, capture particular types of traceability information, like links relations, contributions or rationales etc. And, the existing commercial tools focus on the persistent recording and the maintenance of trace information [6] [7] [8]. In contrast to the above research contributions, they define only a few generic information types, which can be specialized by the user of the system [3].

Thus, the companies still find difficulties in integrating the traceability in their project activities. They also consider this activity an expensive exercise compared to the benefit that it generates.

Indeed, the current techniques for tracing requirements are not adapted to the context of a global development project and do not take into account the different disciplines and points of view in a project. This problem is due to (1) the multi perspectives nature of traceability (ie requiring the capture of several categories of information), (2) and the fact that it is linked to various System Engineering disciplines (requirements engineering, design methods, strategy for project management, development, testing and validation, etc.).

If we take for example the approach of [1], the model obtained after the application of its method is focused on the network of the people which takes part in project. The approach distinguishes the various roles and relations between the contributors and the artefacts produced.

The people and the contributors are essential aspects in RT, but it is not sufficient. Thus, a model must take into account the trace of all the deliverables (product and process) as well as other additional information (decision, management, etc). Although, we can re-use the Gotel model as a trace model fragment in order to manage the types of the relations between stakeholders.

We propose in this paper a traceability management method for modelling requirement traceability with Multi perspectives View (MV). Our approach is based on the concept of model integration, because we found that in the traceability domain, models address only a specific aspect of traceability (or perspective). So, to build a generic model we need some existing model best suited for a given situation. The method is composed of a Meta model to describe traceability information which is supported by a process that guides the system engineers in the traceability management.

Section 2 presents the concepts and formalisms used in the method description. It also explains the Meta model of the MV-TMM approach. Section 3 describes the process of the method with an example of an application. Finally, Sect. 4 presents our conclusions and outlines ongoing further work.

II. OVERVIEW OF THE MV-TMM METHOD

A. MV-TMM method

The method MV-TMM is divided on two principle phases: (a) modeling the traceability information in the context of a development project, and (b) guiding the users in the conception and use of a traceability model. In the modeling part, we propose an approach for the identification of traceability needs in a project. The approach allows the consideration of different category of traceability information and also the guidance of users, by a methodological process. The two phases compose our generic approach that defines the life cycle of a traceability management in projects.

The diagram below (Figure 1) gives an overview of the method. The methods engineer begins with the capture of traceability needs of a project. Then, it uses our Meta model MV-TMM to describe the traceability model suitable for a project. The process of construction and description of the model is based on the MAP model (see next section). And to guide the use and maintenance of the resulting traceability model, system engineer applies the process part for capture and use of information traceability.

B. The MAP Formalism

The proposed process in our approach is based on the MAP process meta-model [9] [10].

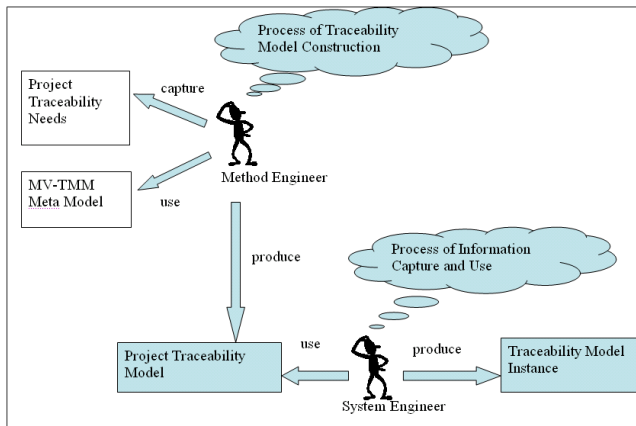


Fig. 1. The MV-TMM traceability method overview

The MAP model allows an intentional representation of processes. It rests on a declarative and flexible ordering of intentions and strategies. An intention is an objective which can be achieved by the execution of one or more process. We add that each MAP model has two particular intentions, Start and Stop, to begin and finish the execution of a process. A strategy is an approach, a manner or a means to carry out an intention. A section is a triplet made up of a source goal, a target goal and a strategy. A section expresses the realization of the target goal by using the strategy once the source goal was carried out.

The MAP chart is represented by a directed and labelled graph. The intentions are represented by the nodes and strategies by the arcs. The directed nature of the graph

translates the flow of the intention source to the target intention via the strategy (for more detail sees the MV-TMM process of our approach).

C. Multi Perspectives Traceability

Our work propose a generic Meta model (MV-TMM) which manages multiple forms of traceability encountered in a project, they also integrates the various concepts used in the existing approaches. The Meta model is multi perspectives, in other words, taking into account the different perspectives associated with requirements traceability, and represents traceability information respecting the peculiarity of each one.

The concept of multi-perspective has been applied in other areas [11] [12] [13] [14]. The Multi-Perspective Modelling (MPM) techniques allow one to present and analyze organizational knowledge from different perspectives, which in turn allows the knowledge to be used for different purposes [15]. In a MPM initiative, several different modeling languages are normally used to describe the different aspects of the same knowledge domain.

Further, the thesis of MPM is that for any “knowledge asset” to be represented adequately, it’s necessary to represent a number of different perspectives on its knowledge - and, possibly, to represent the asset at multiple different levels of decomposition (Figure 2). These ideas are based on those of the Zachman framework [16], and are embodied in various knowledge modelling methods, notably the CommonKADS methodology for knowledge engineering [17].

D. Resulting Meta model

The Figure 3 illustrates the Meta model MV-TMM and its various components using UML language. The following sections provide more details on the different perspectives addressed in the Meta model, as well as the different points of

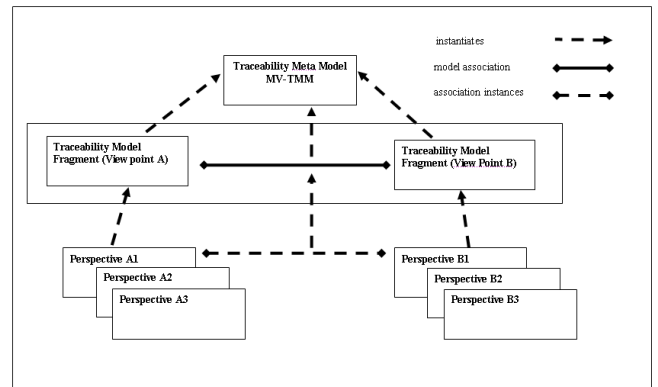


Fig. 2. Abstraction level of the Multi View Traceability Approaches

view to take into account in the design of a traceability model of a project.

Meta model perspectives

-- Actors perspectives

The element Actors Structure of the meta-model describes the traceability information which represents the structure of

those people involved in a project, as well as their commitment to the artifacts produced. In other words the role

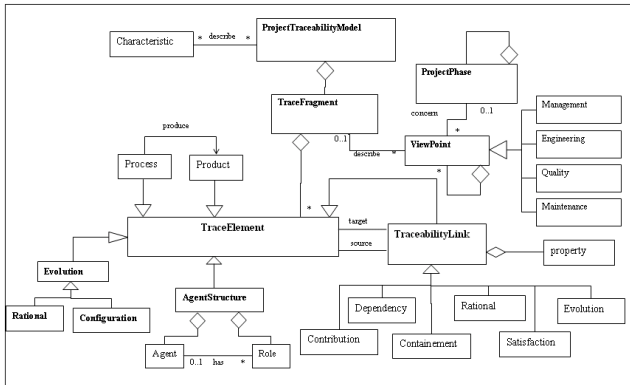


Fig. 3. MV-TMM Meta model

of this element is to represent the social environment and its impact on the traceability model [1].

-- Product Perspectives

This element describes the structure of the different deliverables produced by application of formal and non-formal methods in the specification and design of a system. For example, the product of an Entity / Relationship (E / R) based method, comprises entity and the various relationships between the entities.

We can extend the Product element by the meta-model of product used in the definition of modular methods [28]. It lets capture more detailed traceability information about products. We note that our definition of the product element is not limited to methods, but extends to all types of deliverables in a project.

-- Process Perspectives

The process element records information on the activities giving rise to the creation or the development of a deliverable in a project. Existing methods define such guidance to the actors involved in system development to guide them in defining the process steps and their orders. For example, the Entity-Relation (ER) process provides specific guidelines to create, modify, and delete ER diagrams, entities and relationships between entities.

The process engineering field [29] [9] also provides method for modeling process. We can reuse one of these models to capture traceability information related to process through our meta-model MV-TMM. Our work is not centered on the definition of processes, but the use of existing work in this area for tracing process.

-- Evolution Perspectives

Products and processes will change with the progress of the project. However, the traceability links evolve in parallel and new links appear or disappear because of a change or an evolution. It is therefore necessary to manage traceability link evolution and the reasoning behind their evolution [18] [19] [20].

The evolution element of our meta-model MV-TMM cares about the traceability of artifacts developments and links

relation evolution. We have identified two specialization of this element: (i) the element rational and (ii) configuration element.

(i) Configuration Perspectives: The configuration element represents aspects of configuration management and changes impact analysis in a project, in particular the traceability links configuration.

The role of this element is data status maintains of the identified configuration units (ie, the traceability information), and the data analysis and changes control.

(ii) Rational Perspectives (or justification): This element records information covered by argument and decision following a change or an evolution. Thus, it allows drawing the issues and conflicts generated by the actors in the specification and design work [2].

-- Traceability Link Perspectives

A traceability system can be seen as a semantic network in which nodes are traceability elements of the meta-model MV-TMM and connections between elements are represented by the traceability links. The traceability links describe a temporal relationship in the case of a change between versions of artifacts.

We propose links types in a high level of abstraction adapted to the elements of our meta-model traceability MV-TMM. They represent the different types of possible relations between the elements of the meta-model. The instantiation of these links types can generate other types of links specific to the traceability model of a project. Some of the properties or attributes can be associated with links [21] [22] to characterize them with additional details.

(a) Satisfaction Link Types

The satisfaction links type represents the satisfaction kind of relationships between artifacts of a project, for example, satisfaction links between requirements and the system components that meet requirements. The degree of satisfaction of a requirement can be regarded as a property of satisfaction links type.

(b) Dependency Link Types

Most traceability models explicitly represent dependencies between different artifacts. A dependency link is a relationship between source artifacts that depends on target artifacts. For example, the software requirements of a system depend on hardware requirements that implement them.

Several research publications have focused on the study and characterization of the degree of dependency. We cite as an example the work of Hauser and Clausing [23] proposing a scheme to assign for a dependency a quantitative and qualitative weight. In their approach the degree of dependency is called the "dependency strength" (which is the measure of how much an object affects another). It could be measured qualitatively (for example, high, medium, light, etc.) or quantitative (eg, between 1-10 or in a rating scale of 10). Yu and Mylopolous [24] have adapted another scheme in their framework i* and suggest a type of dependency between actors in an organization.

(c) Evolution Link Types

The Evolution links-type can be used as part of an evolution of artifact over time, in other words, in the order of their creation or modification. The order gives information about the origin of the artifacts as well as trace of evolution history.

(d) Rationales Link Types

The Rational link (or justification link) helps to represent the context in which the artifacts are produced. The context includes all the tools and processes used for the justification of an artifact. Mails and meeting minutes are examples of justifying means of an artifact.

(e) Containment Link Types

Content relations describe a different way of representation of the relationship between artifacts. Such relations deal such as referrals to another artifact, keyword or definition. They provide as a result a new structure and additional forms of traceability.

(g) Contribution Link Types

The term contribution link is used to describe any relationship between an agent (or an actor) and an artifact. It's a two-way relationship because an agent contributes to one or more artifacts and an artifact is produced by one or more actors.

The contribution link may have several levels of granularity depending on the size and nature of the contribution of an actor against an artifact. We can describe it in simple terms as "contribute to". However, it does not give much information about the nature of this contribution that differentiates actor's responsibilities towards artifacts. The work of Gotel & al. [1] is a reusable reference in this kind of relationship.

View Point Analysis

A View Point can be defined as "the description of a part of the information on a particular topic with different perspective" [25].

In the area of requirements traceability we consider that a View Point represents an aspect of traceability that cares about the capture of a particular category of traceability information. It represents a form of traceability among others, which describes only a part of traceability model.

We can classify the views into four basic categories to take into account the various forms of traceability in a project, namely: management, engineering, quality and maintenance views. The classification is deduced from a study of different maturity process of the systems engineering standards [26] [27].

The four categories of the meta-model are basic view points. They are related to each phase of a project and to the various activities carried out. The description of these view points is as follows:

- The management view point: represents traceability information from the management view point. The management aspect in a project includes for example, project management, requirements management, and in general all management activity carried out within a project and

particularly in each phases.

- The engineering view point: The engineering activities include information such as the analysis methods and design, requirement engineering methods, the testing and validation procedures, and in general all formal or informal practices used by engineers in their design and specifications within a project.

- The quality view point: The quality ensures the maturity of project deliverables, and the assuring of the activities quality. Thus, it helps control completeness of the deliverables produced by different actors in a project. A traceability model must also capture the data required by the quality, for example, the information for risk analysis, and compliance with the standards.

- The maintenance view point: The efforts to develop a system have as objective the delivery of a product that meets the user's requirements. However, once the whole or part of product is delivered, it is very likely undergoing changes or developments. Thus, a maintenance phase starts and new requirements emerge. The traceability process must trace the changes and justifications throughout this phase.

III. MV-TMM PROCESS

Our approach requires a process which describes diverse steps to capture the traceability needs of a project and also describing the construction process of a traceability model.

We propose two process models to guide traceability managers in their tasks. The first model guides the identification of organization needs and the traceability model construction. The second model guide users to capture and use traceability information.

A. Strategies for traceability model construction

The construction process of the traceability model is decomposed on two phases (Figure 4):

(i) The capture of the traceability needs

The table 1 shows an example of strategies identifying the context and environment in a project.

Strategies	Resulting information
<I1,S1,I1>	Are users who contribute to the different artifacts in a project and create links between them (analyst, architect)
<I1,S3,I1>	Identifies period of time (phases) in a project to define deliverables expected in each period. Requirements Analysis, Development, Maintenance are examples of phases of the project.

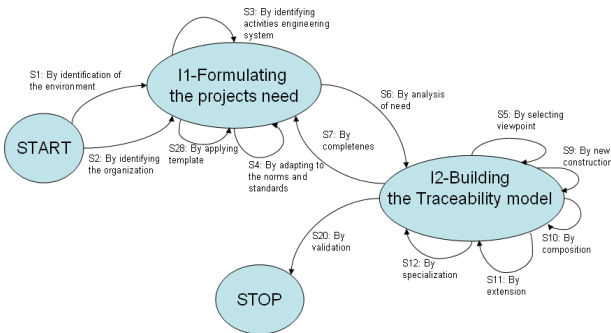


Fig. 4. The MAP of a traceability model construction

<I1,S2,I1>	The departments of an organization that could use traceability in their internal process or as part of a project.
------------	---

(ii) The construction of a traceability model

The model construction process is inspired from the research work of methods engineering domain [28]. Our approach thus will guide the construction with existing approaches by applying strategies like S12, S11, or either by model fragments composition (strategies S10 or S9). Each model fragment must respond to a view point of project traceability needs. The views are described by applying the strategy S5. Thus the model will be built to meet the needs of the project according to several points of view.

Case study:

By implementing strategies S1 and S3 we get for example, the requirements engineering process of the Figure 5. It is composed of three phases: goals capture phase, the requirements capture phase and requirements analysis phase.

We can build a traceability model adapted to this process according to several view points by applying the strategy S5.

The Figure 6 shows an example of view point model corresponding to the requirements capture and analysis phases. We note that the four main views will be decomposed to identify other viewpoints. The traceability model fragments of the two phases must include traceability information

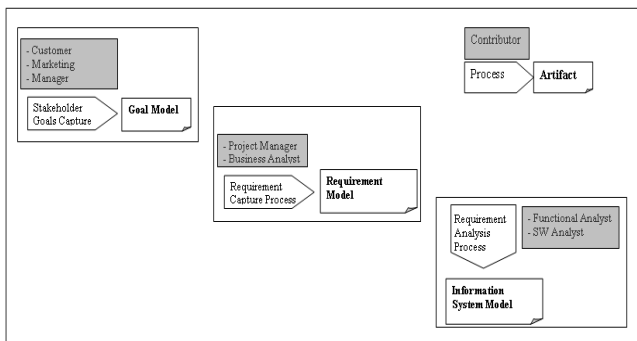


Fig. 5. A Simple RE Process

corresponding to each view.

This example will be analysed in more detail by

implementation of the strategies S9, S10, and S11, to build a model fragments traceability specific to the viewpoint. The traceability model fragments will be integrated to build a generic model for the whole project.

The Figure 7 shows a simple example of traceability model fragment through instantiation of the Product element.

The elements of this model are made up of a goal model of a requirements model, an Entity / Relationship model and traceability links between these three models:

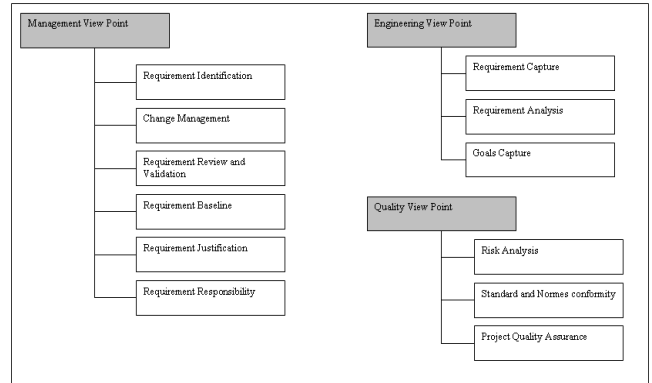


Fig. 6. Example of Traceability View Points

- The goals reflect the need for future, high-level users of the system
- The requirements are captured from high-level goals,
- The Entity / Relationship model represents the software components that make up the information system. These elements must meet the requirements of a customer.
- Relations between the goals, requirements and E / R

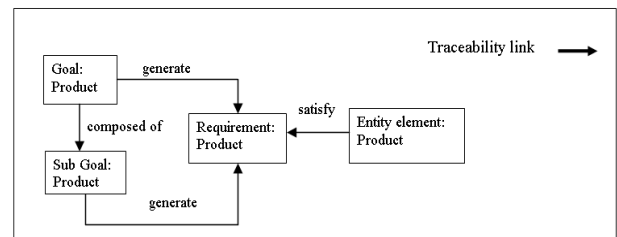


Fig. 7. Traceability Fragment Model (from Engineering View Point)

model are expressed by traceability links. These links are instances of the satisfaction and Dependency relation type of the meta-model MV-TMM.

By analysis of the RE process in terms of change management we get the model shown in Figure 8:

- Requests for changes are stored in the element ChangeRequest.
- The traceability link impact_link contains information on the relationship between request for change and the impacted requirements. It's an instance of the Evolution links type of the meta-model MV-TMM.
- The Stakeholders are at the origin of change requests

and participate in the analysis of such requests.

- The traceability links *approve_link*, *affect_link*, and

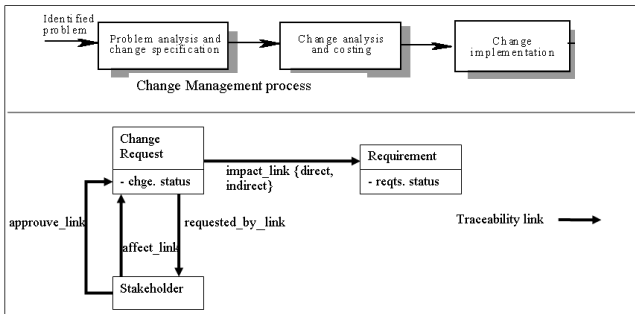


Fig. 8. An Example of the Configuration Management Traceability sub Model

requested_by represent the relationship between the Stakeholder and the changes requests. The links are instances of the Contribution links type of the meta-model MV-TMM.

B. Strategies for information capture and management

(iii) The trace capture guidance

Traceability users must be guided in trace data capture, whenever a stakeholder modifies an approved component; he has to consider the review comments and the approval conditions. Consequently, we should notify the users about the change conditions, retrieve the corresponding traceability data, and display them to the users. The strategies S1 to S5 of the Figure 9 will be used for this purpose.

(iv) The use of trace information

After the capture of the need and the construction of the trace data, the use of trace information constitutes the forth step of our process. This process is generally based on a

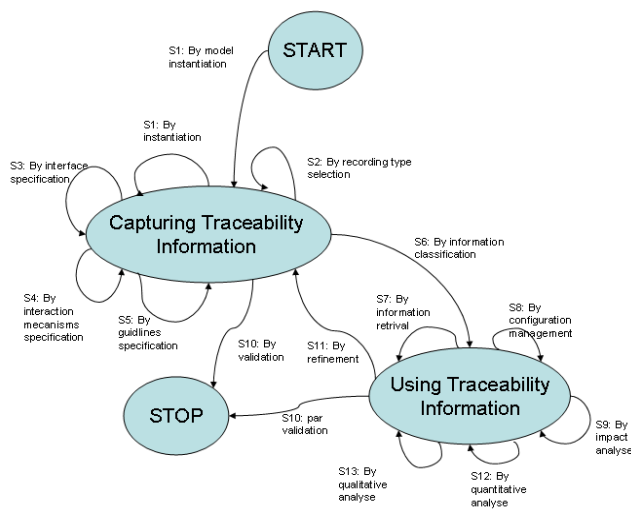


Fig. 9. The MAP of a traceability model usage traceability management tool. We can apply strategies like S6, S7, and S9-S13 for this purpose.

The use of the traceability consists of the update of the traceability data, the automation of certain task, trace information retrieval mechanism and data filter.

(v) Trace data evolution management

Systems always evolve as the environments in which these systems operate change so do stakeholder requirements. Therefore managing change of traced data is a fundamental activity in overall system development. Most Requirement Management (RM) tool manage requirement as configuration units. Then, they play the role of configuration management tools. We must identify traceability evolution policies to control the evolution of trace data and it's rational. This will be done by applying S8 strategy.

IV. CONCLUSION

The multi perspective modelling approach has been adopted to describe a requirement traceability method. We found this approach suitable and often necessary when such a complicated domain must be captured and understood. We propose a method whose core component is a Meta model that provides a taxonomic structure to store all of the sharable, important and fundamental concepts of the requirement traceability domain. The Meta model is based on two formal processes which guide traceability manager in his work. The first model guides the identification of organization needs and the traceability model construction. The second model guide users to capture and use traceability information.

The MV-TMM can be supported by a Requirement Management Tool (RMT) with some extension. We need to add new component to the classical definition of RMT to give support of our approach. This is part of our future work.

REFERENCES

- [1] Gotel O. and Finkelstein A. Contribution structures. Proceedings of RE' 95, 2nd. International Symposium on Requirements Engineering. 1995, March 27-29, York, England. Los Alamitos, California: IEEE Computer Society Press, 1995.
- [2] B. Ramesh and M. Jarke, "Toward Reference Models for Requirements Traceability", IEEE Transactions on Software Engineering, Vol. 27, No. 1, pp. 58-93, January, 2001.
- [3] K. Pohl, K. Weidenhaupt, R. Domges, P. Haumer, M. Jarke, and R. Klamma, "PRIME-Toward Process-Integrated Modeling Environments," ACM Transactions on Software Engineering and Methodology, vol. 8, no. 4, pp. 343-410, Oct. 1999.
- [4] A. Egyed, P. Grunbacher, "Automating Requirements Traceability: Beyond the Record and Replay Paradigm", Proceedings of the 13th IEEE International Conference on Automated Software Engineering, 2002, pp. 163-171.
- [5] J. Cleland-Huang, C. Change, M. Christensen, "Event-Based Traceability for Managing Evolutionary Change", IEEE Transactions on Software Engineering, Vol. 29, No. 9, pp. 796- 810, September, 2003.
- [6] Telelogic, <http://www.telelogic.com/products/doorsers/index.cfm>
- [7] Borland, www.borland.com/us/products/caliber/
- [8] Rational RequisitePro, IBM Corporation, <http://www-306.ibm.com/software/awdtools/reqpro/>.
- [9] Rolland C., Prakash N., Benjamin A. A multi-model view of process modelling. Requirements Engineering Journal, p. 169-187, 1999.
- [10] Benjamin A. Une Approche Multi-démarches pour la modélisation des démarches méthodologiques. Thèse de doctorat en informatique de l'Université Paris 1, 6 octobre 1999.
- [11] Nissen, H.W., Jarke, M. Repository support for multi-perspective requirements engineering. Information Systems 24, 2 (1999), 131-158).
- [12] ISO/IEC International Standard. Information Resource Dictionary System (IRDS) – Framework ISO/IEC 10027 (1990).

- [13] B.Nuseibeh, J. Karmer, and A. Finkelstein A framework for expressing the relationship between multiple view point in requirements specification. *IEEE transaction on software Engineering*, 1994.
- [14] W. Robinson and S. Fickas. Supporting multi perspectives requirements engineering. *In Proc. Of the IEEE Intl. Conf. On Requirements Engineering*, pp. 206-215, Los Alamitos, California. IEEE Computer Society Press (1994).
- [15] Jessica Chen-Burger, *A Knowledge Based Multi-Perspective Framework for Enterprise Modelling*, Informatics Research Report EDI-INF-RR-0036, The University of Edinburgh, February 16, 2001.
- [16] J. Zachman, The Framework for Enterprise Architecture. <http://www.zifa.com/zifajz02.htm>.
- [17] A. Th. Schreiber, J. M. Akkermans, A. A. Anjewierden, R. de Hoog, N. R. Shadbolt, W. Van de Velde and B. J. Wielinga. "Knowledge Engineering and Management: The CommonKADS Methodology", MIT Press, ISBN 0262193000. 2000.
- [18] D. Zowghi and R. Offen, "A Logical Framework for Modeling and Reasoning about the Evolution of Requirements," Proc. Third IEEE Symp. Requirements Eng., Jan. 1997.
- [19] R.C. Sugden and M.R. Strens, "Strategies, Tactics, and Methods for Handling Change," Proc. IEEE Symp. and Workshop Eng. Of Computer Based Systems, pp. 457-462, Mar. 1996.
- [20] M.R. Strens and R.C. Sugden, "Change Analysis: A Step Towards Meeting the Challenge of Changing Requirements," Proc. IEEE Symp. and Workshop Eng. of Computer Based Systems, Mar. 1996.
- [21] H. Kaindle, "The Missing Link in Requirements Engineering," ACM SIGSOFT Software Eng. Notes, vol. 18, no. 2, pp. 30- 39, 1993.
- [22] P. Kruchten, *The Rational Unified Process, An Introduction*, second ed. Addison-Wesley, 2000.
- [23] J.R. Hauser and D. Clausing, "The House of Quality," Harvard Business Review, pp. 63-73, May/June 1988.
- [24] Yu E. Modelling strategic relationships for process reengineering. Ph.D. thesis, Dept. of Computer Science, University of Toronto. 1995.
- [25] Sommerville, Ian; Sawyer, Pete *Requirements Engineering - A Good Practice Guide*. John Wiley & Sons. 1997.
- [26] [SE, 07] CMMI Product Team, *Capability Maturity Model Integration (CMMI)*, SEI, 2002.
- [27] IEEE Computer Society SWEBOK Team, *Guide to the Software Engineering Body of Knowledge (SWEBOK)*, IEEE, 2004.
- [28] Ralyté, J., Rolland, C. An approach for method reengineering. *Proceedings of the 20th ER2001*, Yokohama, Japan, LNCS 2224, Springer, pp.471-484, 2001.
- [29] C. Rolland, M. Jarke, A. Sutcliffe, R. Domges (eds) "The NATURE of Requirements Engineering", Shaker Verlag, Aachen, 1999.