# Instructional Design Rationale with QOC: A Model Driven Engineering and Domain Specific Modeling Approach

El Amine Ouraiba, Christophe Choquet, Philippe Cottier

***Abstract*—This article deals with the design rationale of open pedagogical scenarios based on the QOC (Questions, Options, Criteria) model. In order to concretely formalize the different aspects of instructional design rationale we have adopted a constructive approach based on Model-Driven Engineering and Domain-Specific Modeling. This pragmatic paradigm facilitates to provide the dedicated supports allowing designers to work at a high level of abstraction. We have experimented our approach on Hop3x TEL environment. The Hop3x's DSEML is described by a metamodel of open learning sessions which is defined basing both on QOC and Hop3x-specific educational domain semantic, and accordingly a first version of a specific editor has been generated thanks to EMF tools.**

***Index Terms*—Design rationale, QOC, open pedagogical scenario, model-driven engineering, domain-specific modeling**

## I. Introduction

TEL (Technology Enhanced Learning) systems are complex environments that mobilize human agents (learner, teacher) and artificial ones in interactions conceived in order to improve the quality of the human learning. The design of these systems is a significant effort for learning institutions [29]. A pedagogical scenario could be considered as a model, "a simplification of a system built with an intended goal in mind" [2], where the system is the TEL situation itself and the modeling goal is the organization of the learning activity. According to [1], "such models must necessarily be open, deliberately and strategically imprecise, objects which raise the reflection [31], allowing a collaboration between researchers and practitioners". Moreover, pedagogical scenarios could be described with the help of an Educational Modeling Language (EML) [21], defined by a specific metamodel which is itself linked by conformity relations with the scenarios. Within this framework, we distinguish two approaches of instructional design [9] [13]:

- A "classical" design process here considered as "*interpretative approach*", where an existing EML (such as IMS Learning Design) is chosen for specifying a scenario. Here, the designers (mainly the teachers) have to appropriate the semantics of the EML in order to transform their specific domain model into the metamodel of the chosen EML. However, practitioners and the literature [12] [16] notice the lack of appropriation of the EMLs semantic and the difficulty of use of existing editors in practice [28]. The risk when choosing a pre-defined metamodel is that certain particularities may emerge in situ (in the real learning situation). These particularities cannot be always anticipated at design-time. Their description cannot be supported by the predictive pedagogical scenario because the metamodel chosen to express it does not allow this. The metamodel should empower designers to adapt and evolve their scenarios [13]. In addition, using a generic EML imposes a modeling structure based on some metaphor. To use it, it is necessary to model the scenario in accordance with this structure [13].
- To address the inadequacies of the interpretative approach, a more iterative design process is considered as "*constructive approach*", where the designers, generally helped by modeling specialists, build the metamodel of their specific domain (and thus, their "domain specific" EML) and use it for specifying their scenarios. This approach is much closer to the Domain Specific Modeling one and engages the designers in an iterative design process, eventually supported by reengineering phases. Moreover, according to [17]: the greatest weaknesses in instructional design are to stop at the "*'delivery"'* stage to learner the design's product. Actually, the designing activity of a pedagogical scenario must continue at runtime. According to [8]: it must not be simplified just as a preliminary modeling act of an artifact that is exogenous to the real context of its usage process; it must be continued in the activity of users themselves. This requires the development and the use of models that are endogenous to usage contexts and that may be evolving in parallel (simultaneously) with theirs metamodel.

Our work falls under this last approach. By the use of Model-driven Engineering (MDE) and Domain-Specific Modeling (DSM) paradigm, we want to overcome the difficulties a practitioner teacher can encounter when using generic EMLs and existing editors for designing Open

Manuscript received March 06, 2011; revised March 28, 2011. This work was supported by the Computer Science Laboratory of Le Mans University, France. Instructional Design Rationale with QOC: a Model Driven Engineering and Domain Specific Modeling Approach.

E. A. O., C. C., P. C. Authors are with the SRC Department, University's Institute of Technology of Laval, Le Mans University, France ({el_amine.ouraiba, christophe.choquet, philippe.cottier}@univ-lemans.fr).

Pedagogical Scenarios (OPS) [28] that can be adapted according to execution context. In this article, we use the QOC (Questions, Options, Criteria) model to investigate the instructional design rationale of OPS. The section 2 gives an overview about the design rationale and QOC and explains thereafter our investigation of the OPS design rationale using QOC. In section 3 we present the MDE/DSM paradigm and how we instantiate it for supporting teacher both at design and run time of OPS. Section 4 presents an implementation of our proposal about learning sessions of Hop3x TEL system. We conclude our article by current and future works.

## II. Design rationale of open pedagogical scenarios using QOC

### A. *Design rationale with QOC formalism*

According to [25], Design Rationale (DR) emphasises working with explicit representations not only of possible design solutions, but also of the reasons and processes behind them. In other words, DR is the explicit listing of decisions made during a design process, and the reasons why those decisions were made [18]. DR can be used, according to Burge and Brown [6], for many aims: design verification, evaluation, maintenance, reuse, teaching, communication, assistance, and design documentation.

DR is used by research communities in several science areas such as software engineering, mechanical design, artificial intelligence, civil engineering, knowledge management, cognitive science, and human-computer interaction research. But there is still very little experience of applying the DR in the TEL engineering area, particularly in the instructional design. Indeed, DR can be adopted as a framework for justifying the reasons behind pedagogical decisions taken at design process of units of learning, in order to allow the understanding, recreation, and/or adaptation of design production.

To supporting design rationale, several frameworks are proposed, such as: IBIS [7], DRL [22], DIPA [23], and QOC [25]. According to [19], the studies have concluded that the designers wanted a method requiring less effort to keep this logic, while maximizing the possibility of its reuse. For its simplicity and relevance of its elements we choose to rely on the potential of the QOC model (Questions, Options, Criteria), proposed by MacLean [25], for investigating the instructional design rationale. The QOC is a semi-formal notation which allow producing a graphical representation of DR. [25] noted that QOC can be used for representing the design space around the artifact being produced, thus situating this artifact in a broader context than would otherwise be the case [26]. According to [26] the diagrams can help designer to explain, elaborate, compare, and review design ideas and issues. As it is showed in the left side of figure 1, QOC represents design reasoning [26] as a network of "*Questions*" which highlight key design issues, "*Options*" which represent alternative solutions to these issues and "*Criteria*" to explicitly describe the methods to evaluate the options, such as the requirements to be satisfied or the properties desired. A solid line between a criterion and an option means that the criterion is favorable for option, otherwise it is unfavorable. The preferred option is framed. This allows the designer to read enough to understand the reasons for or against the various options [26].

### B. *Design rationale of open pedagogical scenarios*

[34] notices that the pedagogical choices of a teacher are rarely made explicit. Then, it is difficult to really understand the criteria that led him/her to take such decision at the expense of another deal with a particular teaching-learning context. We choose QOC for capturing instructional design rationale in order to improve learning design quality by arguing decisions thanks to explication of design criteria and to capture evaluated variants to avoid duplication of effort in the future lifecycles. We have defined a model of OPS inspired from the QOC one (see the right side of figure1). In order to achieve his/her *pedagogical objective*, the teacher/designer has to define the necessary elements at the design-time. However, the multitude of the possible execution contexts requires him/her to describe different ways for conducting learning session. We call these different ways "*variants*", where each *variant* is intended to be executed in a particular teaching-learning *context* characterized by a set of *indicators* describing teaching-learning circumstances, where each indicator has a name and a value. Thanks to the *semantic open points*, designer specifies the diverse variants for combining them into a single integrated model, called "*open pedagogical scenario-OPS*". This model is defined in a manner to be personalized before its operationalization in order to meet specific circumstances of a given teaching/learning context by taking into account only the most relevant variant. However, in order to avoid the selection of inconsistent variants that lead to undesirable situations, the validation of OPS personalization must be made at a high level of abstraction in order to ensure the persistence and consistency.

During execution some indicators which characterize the actual context of the learner's activity are calculated at run-time (we use here the Usage Tracking Language for modeling and calculating these indicators, see [30] for details). Based on these indicators, the teacher, in his/her tutor role, could adapt the session by modifying the scenario structure. That causes the emergence of new variants. Then, teacher decides if he/she capitalizes or not these emerged variants with their own relevant execution contexts in OPS structure as predictive variants for reusing, sharing and re-engineering.

ISBN: 978-988-18210-6-5
ISSN: 2078-0958 (Print); ISSN: 2078-0966 (Online)

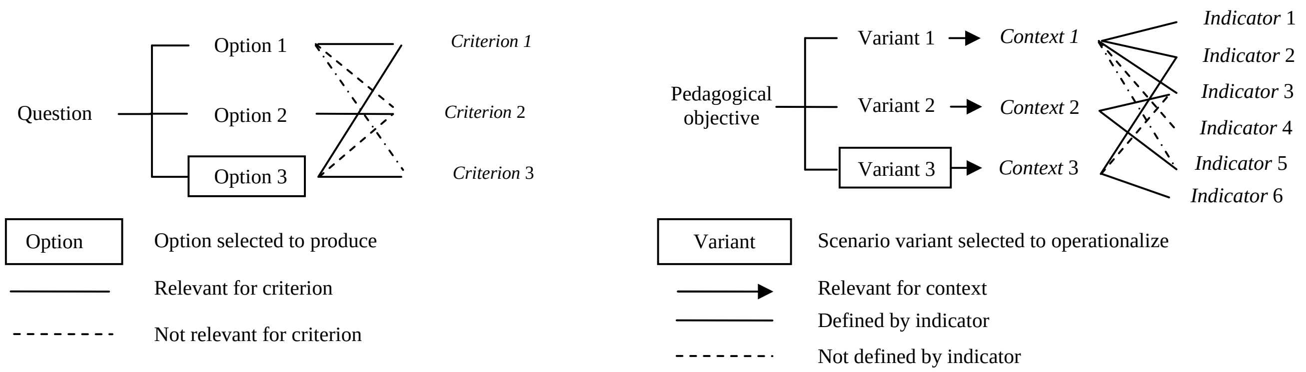


Fig 1. Model of QOC (Questions, Options, Criteria) (on left), model of Open Pedagogical Scenario (on right).

Indeed, this approach based on variants avoids to design every time the same pedagogical scenario. It can promote the systematic reuse of common or proven practices in a specific educational domain, and reduces the modeling effort for teacher. However, this should enable him/her to work at an abstract level.

## III. The MDE/DSM approch

The MDE/DSM represents a pragmatic and robust approach which has best practices and dedicated tools. It allows a high productivity and quality by facilitating the generation of code from models designed at abstract level.

### *A. MDE/DSM principles*

Model-Driven Engineering (MDE) is basically a software development approach. It is an enhancement of the Model-Driven Architecture (MDA) approach, initially proposed by Object Management Group (OMG) in 2001 [27] to provide a solution to the problem of software technologies continual emergence that forces companies to adapt their software systems every time a new “hot” technology appears.

MDE focuses on creating productive models that describe the elements of a system [32] and guide the implementation. The MDE’s goal is to define models that can be operationalized and manipulated by computer. This implies that the produced models (1) conform to the explicit and formal metamodel; and (2) they represent without ambiguity an aspect (a point of view) of artifact to produce. These relations of conformity (conforms-to) and representation (represents) are the basis of MDE [3].

Indeed, this approach is attractive for our problematic because (1) it advocates the development of productive models, which helps the designer to control the choice of development and implementation; and (2) it allows to working directly into the business world of the target application by defining its domain-specific languages.

Moreover, among the advantages of MDE we retain the possibility of reuse and capitalization of both models and practices (transformation and transcription rules between models), the ability to "project" the business knowledge expressed within abstract models (Computation Independent Model – CIM) towards concrete and platform dependent ones (Platform Specific Model – PSM). However, MDE is also regarded as too simplistic and normative [15]. For this, [5] [33], again in an MDE approach, prefer to define their models using domain specific languages (DSL) based on a metamodel simpler but more focused on the business domain of designers. A domain is defined by [10] as “a*n area of knowledge: scoped to maximize the satisfaction of the requirements of its stakeholders, including a set of concepts and terminology understood by practitioners in that area”*.

The Domain Specific Modeling approach (DSM) was defined (1) to reduce the complexity of the transformations and the *semantic losses* they generate, and (2) raise the *level of abstraction* beyond programming by specifying the solution in a language that directly uses concepts and rules from a specific problem domain [20]. The principle here is to develop a DSL, tailored for specifying artifact which instruments a specific activity in a specific context. This DSL has to be formal but its metamodel reflects the domain of the users: the modeling vocabulary used is the domain one. Then, code generators could be developed for directly transform models expressed with a DSL into a specific technological platform framework.

### *B. MDE/DSM for instructional design rationale*

We adopt MDE/DSM approach for providing the necessary supports to practitioner teachers in order to allow them design and adapt the OPS at a high-level of abstraction. We consider in this framework that a scenario, for being really designed and manipulated by a teacher, has to be considered as a domain specific model, expressed with a DSEML (Domain-Specific Educational Modeling Language) situated in his/her teaching context and rooted in his/her practices. In such a paradigm, MDE techniques have to support the transformation of the scenario from domain specific representation to operationalized one, both at the design phase to support the operationalization and at runtime to support the dynamic adaptation [28].

The figure 2 illustrates how we instantiates MDE/DSM paradigm to support teacher both at design and run time. According to the OMG layers view, the DSEML’s metamodel is specified at the M2 level conforming to a meta-metamodel (MOF-MetaObject Facility) defined at the M3 level. This metamodel, inspired from the QOC, is the generic model of OPS (see figure 1). It should formalize the semantic of the teacher own educational domain by describing him/her business language (vocabulary, rules, constraints, etc). The abstract OPS must be specified at the M1 level as models in conformity to DSEML’s metamodel. At the design-time of OPS, designer instantiates the generic elements of metamodel for specifying the mandatory elements, the open points, the foreseen variants with their relevant contexts, etc. At run-time, mandatory elements and selected variant are operationalized on TEL system where they are executed in the learning real context. In unforeseen situations, new variants can be emerged thanks to dynamic adaptations which can be performed, at real time, by defining the elements (*open points*) that are not yet done for responding to new emerged needs.

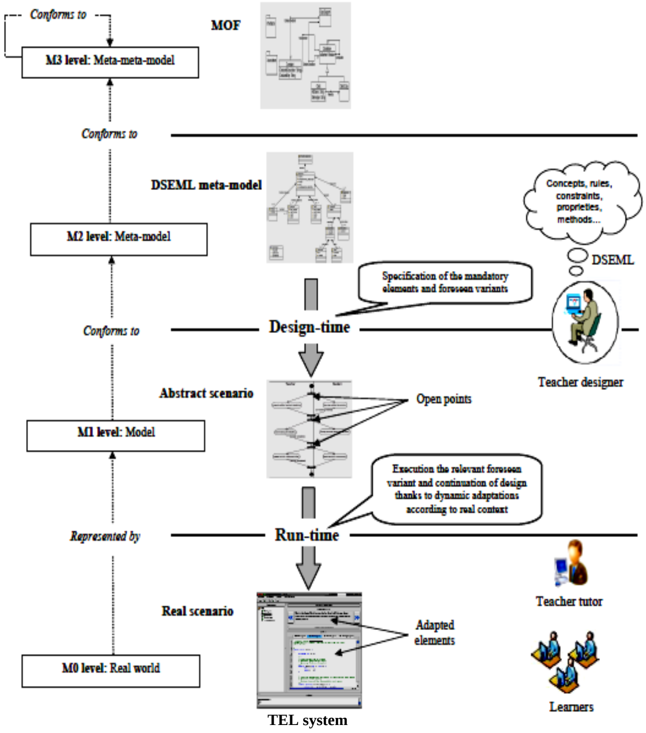


Fig 2. OMG layers view of the open pedagogical scenarios engineering.

## IV. IMPLEMENTATION

To verify our proposal we took Hop3x domain as experimentation field. Hop3x is a practical works TEL environment [11] developed for learning and teaching object-oriented programming languages like Java, C, Ruby, etc. Our objective is to provide the dedicated means to teachers for helping them to design, at an abstract level, the open practical works sessions [29]. For this, the MDE/DSM approach is adopted for concretely formalizing the instructional design rationale of open learning sessions. For implementation, we use Eclipse Modeling Projects [14] because it is a unified set of modeling frameworks, tooling, and standards implementations, such as EMF (Eclipse Modeling Framework), GMF (Graphical Modeling Framework) and ATL (ATLAS Transformation Language).

Thanks to a preliminary study on the Hop3x's usual practices and based on the OPS model proposed above (see figure 1) we have defined a metamodel which describes the Hop3x's DSEML. Technically, this metamodel is an ECORE model where ECORE is the MOF-like meta-metamodel in EMF. Figure 3 illustrates this metamodel in the class-diagram-oriented view proposed by the ECORE graphical internal editor of EMF.

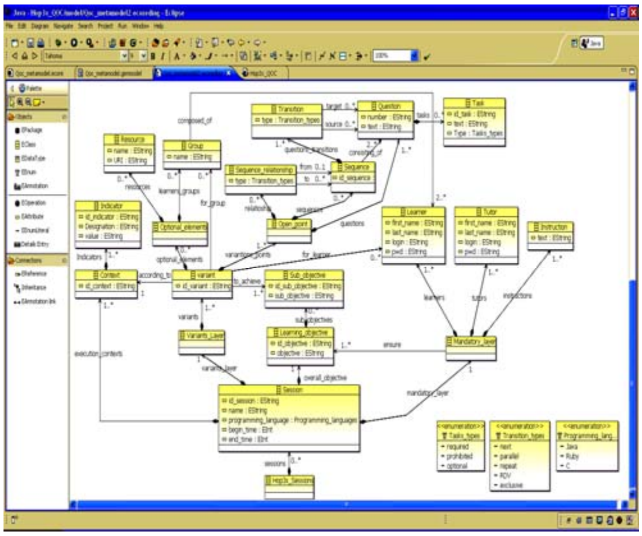

Fig 3. Metamodel of Hop3x's open learning sessions.

Then, the code of a first version of a dedicated editor has been generated automatically from Hop3x's DSEML metamodel thanks to EMF tools. This editor provides a tree-view of the models which are namely the open Hop3x learning sessions (see figure 4). By using this editor the teachers who want to use Hop3x can design sessions at an abstract level and generate them directly in the XML format readable by Hop3x system (in practice, we are currently developing a more easy-to-use editor, using functionalities of GMF).

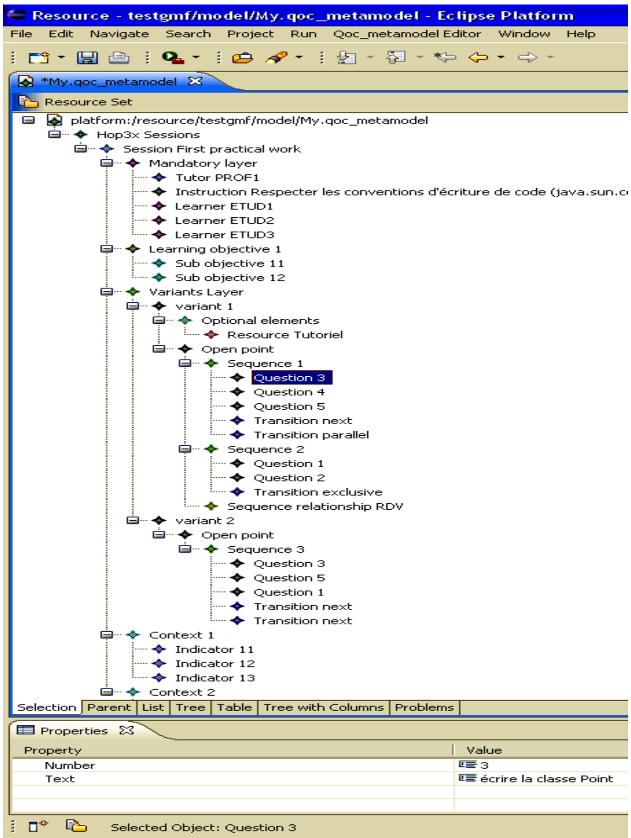


Fig 4. Example of an open Hop3x learning session designed by the dedicated editor.

ISBN: 978-988-18210-6-5
ISSN: 2078-0958 (Print); ISSN: 2078-0966 (Online)

## V. Conclusion and Future Works

In this article we have used the QOC model for investigating the design rationale of open pedagogical scenarios. In order to concretely formalize this, we have adopted a constructive approach of instructional design based on the use of the MDE/DSM paradigm. This pragmatic paradigm facilitates indeed to provide the necessary supports for designers for allowing them to perform the instructional design rationale at a high level of abstraction. To verify our proposal we took Hop3x as experimentation field. We aim to help practitioner teachers to have a reflection about their design rationale. Our objective is to provide them the dedicated supports for designing open learning sessions. For doing this, we have defined a metamodel of OPS based both on QOC and on Hop3x-specific domain semantic, and then a first version of a dedicated editor has been generated automatically thanks to EMF tools. We will use the GMF in a second time to add a graphical layer on top of EMF in order to provide a graphical editor allowing designing open learning sessions at a more abstract level.

Using the developed Hop3x-specific editor we are conducting iteratively interviews with Hop3x's users in order to promote the expression of the dynamic adaptation requirements. The information gathered from these interviews will also help us for adapting Hop3x's functionalities for transforming it into an open TEL system that can be configured by its users according to usage contexts. We consider that the reconfiguration of this type of TEL systems is possible through the adaptation of open pedagogical scenarios [29].

## References

[1] M. Baker, "Rôle et usage des modèles en Environnements Informatiques pour l'Apprentissage Humain". 5ème école thématique du CNRS sur les EIAH Personnalisation des EIAH 7-12 juillet 2007 Saint Quentin sur Isère, France
[2] J. Bézivin and O. Gerbe, "Towards a precise definition of the OMG/MDA framework".Automated Software Engineering,USA,2001.
[3] J. Bézivin, M., Blay-Fornarino, M. Bouzeghoub, J. Estublier, and J. M. Favre. "Rapport de Synthèse IDM". Synthèse de l'AS CNRS sur l'Ingénierie Dirigée par les Modèles, 2005.
[4] L. Becky and L. Hooey. "Requirements for a Design Rationale Capture Tool to Support". NASA paper, 2006.
[5] V. Bhanot, D. Paniscotti, A. Roman, and B. Trask. "Using Domain-Specific Modeling to Develop Software Defined Radio Components and Applications". In: the 5th OOPSLA Workshop on Domain-Specific Modeling (DSM'05), San Diego, California, USA, 2005.
[6] J. Burge and D.C. Brown, "Design Rationale: Types and Tools,Technical Report", Worchester Polytechnic Institute, Computer Science Dept,1998.
[7] J.E., Conklin and M.L. Begeman. "gIBIS: A Hypertext Tool for exploratory Policy Discussion", ACM Transactions on Office Informations Systems, 1998, pp 303-331.
[8] P. Cottier and H. El-Kechaï, "L'usager concepteur en situation: Conception collective d'un livret électronique d'apprentissage (LÉA)".In: Ingénierie des systèmes d'information, ISSN1633-1311. vol.14, n°3, 2009, p162.
[9] C. Choquet, "Ingénierie et réingénierie des EIAH – L'approche REDiM". Habilitation à diriger les recherches en informatique, Université du Maine, 2007, France.
[10] K. Czarnecki and U.W. Eisenecker. "Generative Programming:Methods, Tools, and Applications". Addison Wesley, 2000.
[11] C. Després, P. Jacoboni, P. "Hop3x: Un Environnement de Suivi de TP de Programmation". Software, Université du Maine, 2010.
[12] J.M. Dodero, C. Tattersall, D. Burgos, and R. Koper, "Transformational Techniques for Model-Driven Authoring of Learning Designs". H. Leung et al. (Eds.): Advances in Web Based Learning – ICWL 2007, LNCS 4823, Springer-Verlag Berlin Heidelberg, Vol.4823, 2008, pp. 230 – 241.
[13] H. El-Kechaï. "Conception collective de scénarios pédagogiques dans un contexte de réingénierie Une approche par la métamodélisation située". Phd thesis of the Université du Maine, Université du Maine, 2008, France.
[14] Eclipse EMP, "Eclipse Modeling Projects" , http://www.eclipse.org/modeling/. Last view 12/2010
[15] J.M. Favre, J. Estublier, and M. Blay-Fornarino. "L'ingénierie dirigée par les modèles – Au delà du MDA", Traité IC2, série Informatique et Systèmes d'Information, Hermès, Lavoisier, Paris, 2006.
[16] D. Griffiths, J. Blat, R. Garcia, H. Vogten, and K. Kwon, "Learning Design Tools". In Learning Design – A Handbook on Modeling and Delivering Networked Education and Training, R. Koper and C. Tattersall (Eds). ISBN 3-540-22814-4, Springer-Verlag, 2005, pp. 114-116.
[17] F. Henri. "Modèles du sujet pour la conception". Revue de l'éducation à distance, vol.22, n°.1, 2007, pp101-106.
[18] Alex P. J. Jarczyk, Peter Löffler , and Frank M. Shipman Iii, "Design Rationale for Software Engineering: A Survey", 25th Hawaii International Conference on System Sciences, 2, 1992, pp. 577-586
[19] L. Karsenty, "Capitaliser le contexte des décisions en conception", livre Management des connaissances : modèles d'entreprise et applications, , 2001, p.49-69
[20] S. Kelly, and J. Tolvanen, "Domain-specific modeling: enabling full code generation". Wiley-IEEE Computer Society Press. ISBN: 978-0-470-03666-2, 2008, 448 pages.
[21] R. Koper, "Modeling units of study from a pedagogical perspective The pedagogical metamodel behind EML"; OUNL Technical report,Heerlen, The Netherlands, 2001.
[22] M. Lewkowicz, and M. Zacklad, "La capitalisation des connaissances tacites de conception à partir des traces des processus de prise de décision collective", 1998
[23] M. Lewkowicz, and M. Zacklad, "A guide through the construction of a groupware of efficient knowledge management", 2000
[24] J. Lee, "Design Rationale Systems: Understanding the Issues". *IEEE Expert* 12 (3), 1997, pp 78–85
[25] A. Maclean, R.M. Young, V.M.E. Bellotti, and T.P. Moran, "Questions, Options, and Criteria: Elements of Design Space Analysis", Human-Computer Interaction, Vol.6, 1991.
[26] D. McKerlie, and A. MacLean, "Experience with QOC Design Rationale". INTERCHI'93, Amsterdam : ACM, 1993
[27] OMG, "MDA specification guide. Version 1.0.1", 2006, accessible at: http://www.omg.org/docs/omg/ 03-06-01.pdf, consulted 01/05/10.
[28] E.A. Ouraiba, C. Choquet, P. Cottier, C. Després, and P. Jacoboni, "Engineering of open learning scenarios: the case of Hop3x learning scenarios".Proceedings of IEEE ICALT'10,Tunisia, 2010, pp. 264-268
[29] E.A. Ouraiba, C. Choquet, P. Cottier, "Opening TEL systems for teachers: A Domain-Specific Modeling & Model-Driven Engineering Approach". CSEDU'11, Netherlands, in press
[30] D. Pham Thi Ngoc, S. Iksal, C. Choquet, and E. Klinger, "UTL-CL: A Declarative Calculation Language Proposal for a Learning Tracks Analysis Process". Proceedings of IEEE ICALT''09, Riga, Latvia, July 2009, pp.681-685.
[31] M. Puustinen, M. Baker, and K. Lund GESTALT, "A framework for redesign of educational software". Journal of Computer Assisted Learning 22, , 2006, pages 113.
[32] D.C. Schmidt, 2006. "Model-Driven Engineering". *IEEE Computer*. 39 (2), 2006.
[33] J.P. Tolvanen. "Domain-Specific Modeling: How to Start Defining Your Own Language", DevX.com, 2006, accessible at: http://www.devx.com/enterprise/Article/30550
[34] E. Villiot-leclercq, "La méthode des Pléiades : un formalisme pour favoriser la transférabilité et l'instrumentation des scénarios pédagogiques", research paper, Vol.14, 2007