

## Identification et formalisation du langage de conception pédagogique des plateformes de formation

Nour El Mawas<sup>1</sup>, Lahcen Oubahssi<sup>1</sup>, Pierre Laforcade<sup>1</sup>

<sup>1</sup> LUNAM Université, Université du Maine, EA 4023, LIUM, 72085 Le Mans, France  
prenom.nom@univ-lemans.fr

**Résumé.** L'utilisation des plateformes de formation en ligne présente plusieurs difficultés liées à l'opérationnalisation des scénarios pédagogiques et à l'appropriation des outils de conception d'un cours au niveau des enseignants. Le travail de recherche présenté dans cet article a pour objectif de proposer un processus d'identification et de formalisation du langage de conception pédagogique implicite des plateformes de formations en ligne. Ce processus est composé de trois parties principales : une analyse IHM-macro, une analyse fonctionnelle et une analyse micro. Nous expérimentons notre processus sur la plateforme de formation Moodle.

**Mots-clés.** Plateforme de formation, apprentissage en ligne, conception pédagogique, Opérationnalisation, Processus, Moodle.

**Abstract.** The use of LMS systems presents many difficulties related to the operationalization of learning scenarios by teachers and their need to appropriate tools about how they must think and design their courses. The objective of this paper is to define the necessary analysis and steps for the identification and formalization of an LMS instructional design language. The process takes into account three different viewpoints: a viewpoint centered on macro-HMI (Human-Machine Interfaces), a functional viewpoint and a micro-HMI viewpoint. We concretely illustrate the proposed process on the Moodle platform.

**Keywords.** Learning Management System, E-learning, Instructional Design, Operationalization, Process, Moodle.

### 1 Introduction

Notre travail de recherche s'inscrit dans le domaine de l'ingénierie et de la réingénierie des EIAHs. Selon [1], un EIAH représente un environnement informatique proposant une intention didactique et/ou pédagogique et dont la finalité est de susciter ou d'accompagner un apprentissage. En tant qu'environnement informatique, un EIAH offre des services spécifiques dédiés au domaine de l'éducation pour ses multiples clients et usagers. Les services offerts incluent généralement un contrôle d'accès, des

outils de communication, l'administration des groupes d'utilisateurs, etc. En tant qu'environnement d'apprentissage, un EIAH est proposé pour toute personne impliquée dans un processus d'apprentissage ou dans un parcours pédagogique. Il embarque une intention didactique/pédagogique et des éléments métiers relevant de la conception pédagogique. Parmi les systèmes informatiques spécifiques aux EIAH, nous nous intéressons aux plateformes de formation en ligne.

Le travail de recherche présenté dans cet article est réalisé dans le cadre du projet GraphiT (*Graphical Visual Instructional Design Languages for Teachers*) qui a pour objectif d'élaborer et d'instrumenter des langages de modélisation pédagogiques graphiques centrés sur le métier des plateformes de formation et dirigés vers les besoins des praticiens. Nous avons étudié dans le cadre de ce projet les possibilités et limites en termes d'expressivité pédagogique de langages de scénarisation opérationnalisables, c.-à-d. dont les scénarios produits peuvent être informatiquement exploités pour la mise en œuvre sur des plateformes de formation à distance. Ces langages de conception pédagogique visent à promouvoir et à améliorer les utilisations des plateformes par la spécification de langages pédagogiques graphiques, et leurs outils-auteurs, centrés plateforme. Malgré les standards [2] [3], approches [4], langages [4], architectures [5] et outils existants [6] [7] pour faciliter la conception pédagogique, ils sont souvent incompatibles avec les plateformes. De plus, ils ne simplifient pas l'opérationnalisation des modèles produits. Ils entraînent plusieurs modifications du scénario initial et des pertes d'informations et de sémantique lors de l'opérationnalisation des scénarios spécifiés à l'extérieur des plateformes. Pour cela, nous proposons dans cet article un processus d'identification et de formalisation des métiers des plateformes. La suite de cet article est découpée comme suit. La section 2 définit notre approche tout en le mettant en application sur la plateforme Moodle. La section 3 conclut l'article et présente ses perspectives.

## 2 Notre approche

L'adoption d'un langage conforme à celui d'une plateforme a pour avantage de préserver la sémantique des scénarios pédagogiques. Ces scénarios peuvent être ainsi implémentés sans perte d'informations ou de données. Ce processus est à destination de la communauté des plateformes de formation en ligne et plus spécifiquement aux ingénieurs pédagogiques qui travaillent dans des services TICE et aux concepteurs ayant des compétences en informatique [8].

Notre processus est composé de quatre principales parties : l'analyse IHM-macro, la factorisation du modèle IHM-macro, l'analyse fonctionnelle, ainsi que l'analyse micro. La répartition du processus en quatre étapes principales est le résultat de l'étude que nous avons réalisée sur plusieurs plateformes ainsi que le résultat de plusieurs réflexions, de recherches et de réunions de travail.

Nous présentons le processus d'identification et de formalisation du langage de conception pédagogique sur la plateforme Moodle (version 2.4).

## 2.1 Analyse IHM-macro

L'analyse IHM-macro sur Moodle consiste à identifier les interfaces liées à la conception des cours (analyser les titres et les chemins de navigation). Tout d'abord, nous avons analysé le contenu des interfaces cours sur Moodle. Chaque élément du cours amène à une nouvelle interface, soit pour sa présentation ou pour sa modification. Moodle propose une approche descendante (*top down*) pour la conception du cours. Les autres interfaces telles que l'interface d'ajout d'un forum sont accessibles à partir de l'interface principale. La figure 1 illustre le résultat de l'application de l'analyse IHM-macro sur la plateforme Moodle<sup>1</sup>. Un cours est composé de sections, de catégories, de groupes, de groupements, d'objectifs et de barèmes. Chaque section est composée d'activité(s) et de ressource(s). Dans la figure 1, nous présentons seulement une ressource (*Label*) et cinq activités (*Survey*, *Chat*, *Workshop*, *Quiz* et *Forum*) pour des raisons de clarté.

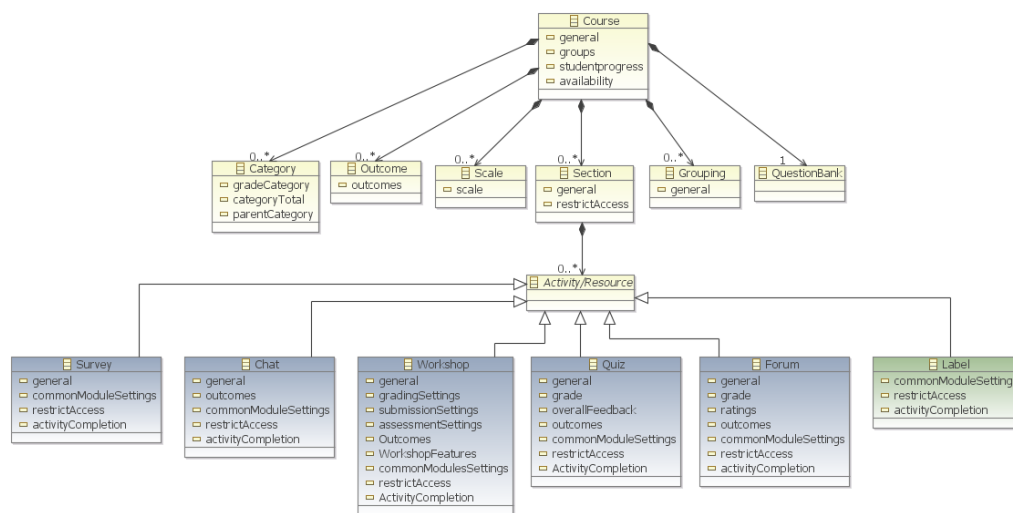


Fig. 1. Extrait du modèle IHM-macro de Moodle

Dans la page de spécification de chaque concept, les attributs se répartissent en différentes parties. Par exemple, les champs de l'activité *Chat* sont divisés en quatre parties nommées : *general*, *common module settings*, *restrict access*, and *activity completion*. Les noms de ces parties sont présentés dans le modèle IHM-macro. A noter qu'il existe deux types de relation dans ce modèle : la relation de composition et la relation d'héritage.

## 2.2 Factorisation

Après l'analyse IHM-macro, nous proposons un processus de factorisation. Pour le cas de Moodle, nous avons remarqué que toutes les activités et toutes les ressources ont en commun les sections suivantes : « *common module settings* », « *restrict ac-*

<sup>1</sup> Le méta-modèle complet de moodle est consultable sur le lien suivant : [http://www-lium.univ-lemans.fr/~laforcad/graphit/wp-content/uploads/2015/01/Moodle\\_2.4.png](http://www-lium.univ-lemans.fr/~laforcad/graphit/wp-content/uploads/2015/01/Moodle_2.4.png)

« *activity completion* ». Nous avons alors déplacé ces attributs dans la classe *Activity/Resource*. De même, toutes les activités ont en commun l'attribut « *general* », pour cela nous avons créé la classe « *Activity* » et nous avons déplacé l'attribut « *general* » dans cette classe. Certaines autres activités peuvent avoir des objectifs par exemple le chat, le workshop et le quiz. Ce qui explique l'ajout de la classe « *ActivityWithOutcomes* » au modèle macro. Cette classe a « *outcomes* » comme attribut. Nous avons remarqué que certaines activités avec *outcomes* peuvent être notées. Nous avons donc ajouté la classe « *GradedActivityWithOutcomes* ». Certaines activités ont en commun l'attribut « *grade* ». La classe « *ActivityWithGradedSection* » est créée et elle contient l'attribut « *grade* ». De même, certaines activités de « *ActivityWithGradeSection* » ont en commun l'attribut « *ratings* ». La classe « *ActivityWithRatingsSection* » est ajoutée au modèle macro avec l'attribut « *ratings* ». Toutes les étapes suivantes sont effectuées sur la base de cette analyse.

### 2.3 Analyse fonctionnelle

En se basant sur le modèle IHM-macro, nous avons procédé à une analyse fonctionnelle sur Moodle. Au cours de cette analyse, nous avons tout d'abord décomposé chaque IHM en plusieurs zones. Ensuite, pour chaque zone, nous avons étudié les composants de l'interface graphique (les *widgets* tels que les liens, les boutons, les icônes, etc.) pour identifier les fonctionnalités liées à la conception pédagogique. Par exemple, à partir de l'interface principale d'un cours sur Moodle, un enseignant peut afficher/cacher/déplacer une section. Il peut modifier la description du cours et gérer les différents groupes. Il peut aussi ajouter une activité ou une ressource dans une section spécifique. Quand un enseignant ajoute un forum, il est alors redirigé vers une nouvelle page concernant les spécifications du forum. Il peut ajouter des fichiers, il peut aussi ajouter/modifier/supprimer/séparer une discussion et répondre à une discussion. Cette analyse est très importante dans notre processus, car elle permet de vérifier l'existence des éléments du modèle IHM-macro ainsi que leurs relations.

### 2.4 Analyse micro sur Moodle

**Analyse IHM-micro.** Elle consiste à identifier les caractéristiques des éléments relevant de la conception pédagogique. Elle est basée sur les modèles macro et fonctionnel. Par exemple, dans Moodle, la classe « *course* » a comme attribut « *general* ». Dans cette phase, nous étudions en détail les champs avec l'usage pédagogique lié à cet attribut. « *Fullname* » et « *shortname* » sont ces champs, donc nous remplaçons l'attribut « *general* » dans le modèle IHM-macro par les attributs « *fullname* » et « *shortname* » dans le modèle IHM-micro.

La figure 2 montre un extrait du modèle IHM-micro (sans prendre en considération les corrections en rouge). Des relations de référence apparaissent dans ce modèle. Par exemple la classe abstraite « *ActivityWithOutcomes* » fait référence à la classe « *Outcome* » : un enseignant peut définir les objectifs d'un cours puis il associe un objectif spécifique à des activités de Moodle à l'exception des activités *Choice*, *Survey*, *Wiki* et *Feedback*.

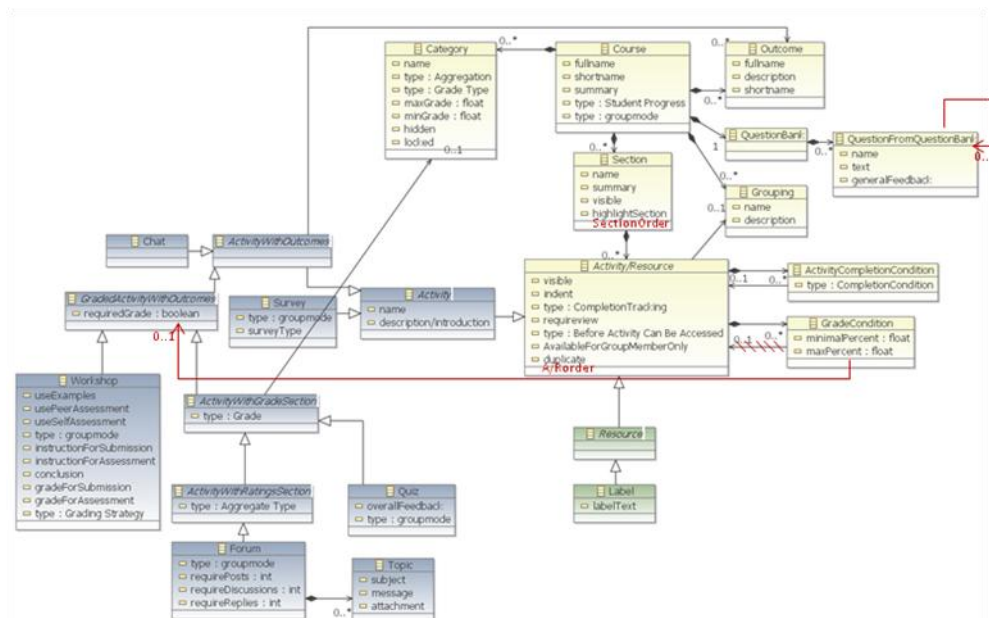


Fig.2. Extrait du modèle IHM-micro (sans les corrections en rouge), extrait du modèle final de Moodle (avec les corrections en rouge)

**Analyse technique.** L'analyse technique consiste à étudier la base de données de Moodle. Notre but est d'identifier le langage de conception pédagogique d'un point de vue technique afin de valider la persistance des données spécifiques à ce langage. L'analyse de la base de données consiste à spécifier le modèle conceptuel de données réduit. Nous avons parcouru toutes les tables de la base de données de Moodle. L'analyse sémantique des titres (des tables et des champs) a permis (1) d'identifier un premier dénombrement des tables liées à la conception pédagogique d'un cours en termes d'activités et de ressources pouvant être intégrées et (2) d'écarter toutes les tables qui portent sur des spécifications techniques ou qui ne possèdent pas d'enregistrements telles que les tables de gestion des utilisateurs, etc. Nous avons étudié les dépendances et les relations entre les tables de la base de données. La génération du modèle conceptuel de données réduit est basée sur les règles de *rétro-conception*. Les clés étrangères ont permis la spécification des multiplicités requises.

**Confrontation et formalisation.** A partir de la comparaison des deux modèles, nous remarquons que chaque élément/attribut présent dans le modèle IHM-micro est généralement présent dans le modèle technique. Par contre, certains éléments sont présents dans le modèle technique sans être présents dans le modèle IHM-micro. Pour cela, nous nous référons au code source PHP pour vérifier la présence de ces éléments. La figure 2 (y compris les corrections en rouge) montre un extrait du modèle final de Moodle. Les corrections en rouge présentent le résultat de confrontation des deux modèles. Par exemple, grâce à l'analyse technique, nous avons trouvé que chaque section a un ordre. Cet attribut ne peut pas être détecté dans le modèle IHM-micro. L'analyse du code source confirme la présence de cet attribut. L'attribut *SectionOrder* est représenté dans le modèle final de Moodle.

### 3 Conclusion

Dans cet article, nous avons abordé le problème de l'identification et de l'explicitation du métier de conception pédagogique *implicite* dans les plateformes de formation existantes. Nous avons proposé une approche et un processus basés sur la méta-modélisation pour la formalisation du modèle résultant. Nous avons illustré le cas particulier de la mise à l'essai, à fin de première validation, de la plateforme Moodle (en version 2.4), mais nous avons également expérimenté notre approche sur la version 2.0 de Moodle ainsi que sur la plateforme Ganesha. Nous sommes en train d'appliquer cette approche sur deux autres plateformes (Dokéos, Sakai). Ces différentes expérimentations devraient permettre d'affiner encore davantage les différentes étapes du processus, comme la formalisation des différents modèles intermédiaires et finaux que nous proposons. Ce travail d'explicitation et de formalisation contribue à aider la compréhension, la simulation, l'appropriation d'un système d'apprentissage que cela soit pour guider son utilisation, étendre ses fonctionnalités, ses usages, etc.

**Remerciements.** Ce travail a été réalisé dans le cadre du projet ANR GraphiT [ANR-2011-SI02-011] (<http://www-lium.univ-lemans.fr/~laforcad/graphit/>).

### Références

1. Tchounikine P., Baker M., Balacheff N., Baron M., Derycke A., Guin D., Nicaud J.F. et Rabardel P. (2004). Platon-1: quelques dimensions pour l'analyse des travaux de recherche en conception d'EIAH. Rapport de l'Action Spécifique du département STIC du CNRS : Fondements théoriques et méthodologiques de la conception des EIAH.
2. Martinez-Ortiz, I., Sierra, J.L., and Fernández-Manjón, B.: Enhancing IMS LD Units of Learning Comprehension. In the 4th International Conference on Internet and Web Applications and Services, Venice, Italy (May 2009) 561-566.
3. Mekpiroona O., Tammarattananonta P., Buasrounga N., Apitiwongmanita N., Pravalpruka B. et Supnithia T.: SCORM in Open Source LMS : A case study of LEARNSQUARE. In ICCE2008, Taipei, Taiwan, (2008) 166-170.
4. De Vries, F., Tattersall, C., and Koper, R., "Future developments of IMS Learning Design tooling", Educational Technology & Society, 9 (1), (2006) pp. 9-12.
5. Alario-Hoyos, C., Bote-Lorenzo, M.L., Gómez-Sánchez, E., Asensio-Pérez, J.I., Vega-Gorgojo, G., Ruiz-Calleja, A., GLUE!: An Architecture for the Integration of External Tools in Virtual Learning Environments, Computers & Education. 60(1), (2013) 122-137.
6. Baggetun, R., Rusman, E., and Poggi, C., "Design Patterns for collaborative learning: from practice to theory and back", In EdMedia, pp. 2493-2498, Chesapeake, 2004.
7. Al-Ajlan, A., and Zedan, H., "E-learning (Moodle) based on service oriented architecture", In Proc. of the EADTU, Lisbon, Portugal, 8-9 November, Lisbon, vol. 1, (2007).
8. Loiseau E, Laforcade P.: Specification of learning management system-centered graphical instructional design languages - A DSM experimentation about the Moodle platform. In ICISOFT'13, Reykjavik, Iceland (2013) 29-31.