# Towards the identification and formalization of LMS instructional design languages

Nour El Mawas, Lahcen Oubahssi, Pierre Laforcade, Aymen Abedmouleh

Laboratoire d'Informatique de l'Université du Maine (LIUM), EA 4023
LUNAM Université, Université du Maine
72085 Le Mans, France
Surname.Name@univ-lemans.fr

**Abstract.** In recent years, researchers have begun to formalize LMSs instructional languages in order to specify models in conformance with their underlying infrastructure. To our knowledge, there is no proposition that focuses on identifying an explicit process to formalize such languages. The research presented in this paper aims to define the necessary analysis and steps for the identification and formalization of an LMS instructional design language. Academic institutes have adopted LMSs; however, many teachers have difficulty using LMSs to create learning designs. Teachers are not familiar with implicit learning design domain of LMSs. They need solutions to bridge the gap between their educational intentions and the pedagogical scenarios proposed by the LMS at their disposal. In our work, we aim at supporting practitioners to overcome these LMSs' obstacles in order to help them in focusing on the design of learning situations. Our contribution consists in extracting, identifying, and formalizing the LMS implicit instructional design language by proposing a meta-model to the design of learning situations.

We propose a process to identify and formalize the instructional language of LMSs. Our approach is based on four viewpoints: the macro-HMI analysis, the factorization of HMI-macro model, the functional analysis and the micro analysis.

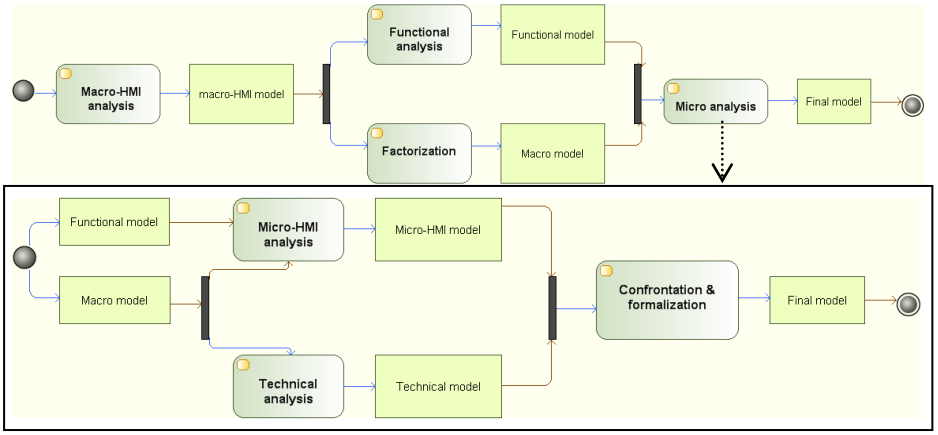


**Fig. 1.** Analysis process of the instructional design language

The first viewpoint consists of HMIs analysis according to two strategies: (1) the analysis of existing situations on the platform and (2) the analysis of interfaces related to the specification of new situations. After the macro-HMI analysis, we factorized the macro-HMI model in order to obtain the simplified macro model. The functional analysis focuses on the identification of LMS existing functions. The last viewpoint concerns the micro analysis of the LMS instructional design language. It is composed of three steps: the micro-HMI analysis, the technical analysis, and the confrontation of micro-HMI and technical models.

We apply our process on the Moodle 2.4 platform. The application of **macro-HMI analysis on Moodle** consists of identifying interfaces related to course design. We analyzed interfaces titles and navigation paths / URLs. We browsed all the links in a specific interface. After the macro-HMI analysis, we applied the **factorization process**. For example, we noticed that all activities/resources had the common attributes: "*commonModuleSettings*", "*restrictAccess*", and "*activityCompletion*". So we moved these attributes to the Activity/Resource class. Based on the macro-HMI model, we proceeded to the **functional analysis on Moodle**. We divided each interface to several areas. Then, for each area, we studied the graphical interface components to identify functionalities related to instructional design. For example, from the main interface of a Moodle course, a teacher can show/hide/move a section. He can modify the course description, and manage different groups. He can also add an activity/resource in a specific section… The application of **IHM-micro analysis** is about characteristics identification of instructional design elements. It is based on the macro and functional models. The **technical analysis** consists in analyzing the Moodle database. Our goal is to identify the Moodle instruction design language from a technical viewpoint to approve the relevant of specific data for this language. This analysis consists in specifying the reduced Conceptual Data Model for Moodle in relation with the instruction design. We have reviewed all Moodle database tables. The micro-HMI analysis and the technical analysis have specified two Moodle instructional design models according to two different viewpoints. In this step, we are interested in the **confrontation of these models** to formalize the Moodle instructional design language. We think that the use of only one analysis method presents many negative points. Indeed, the micro-HMI model depends directly on the Moodle analyst competence. This means the possibility lack of pedagogical attributes. Similarly, the technical analysis is not an easy task. Many data structures are not explicitly reported when creating the database.

In conclusion, we present in this paper a meta-model-based approach for identifying and formalizing LMS languages. We apply our proposed approach on the Moodle platform. The meta-model will be used as a basis for the development of the external editor. It will guide and generate most of the final code for the editor. This will facilitate the use of LMSs and allow teachers becoming more familiar with e-learning.

*This work and submission are funded by the French GraphiT project [ANR-2011-SI02-011] (http://www-lium.univ-lemans.fr/~laforcad/graphit/).*