Making explicit the Moodle instructional design language

Nour El Mawas, Lahcen Oubahssi and Pierre Laforcade
Université du Maine
LIUM (Laboratoire d'Informatique de l'Université du Maine) Lab
TEL (Technology-Enhanced Learning)Team
Presentation outline

• Context

• The GraphiT project

• Our approach / Moodle case study

• Conclusion & Perspectives
Context

- Learning Management Systems like MOODLE
  - Widespread within academic organizations
  - Not limited to distant courses
  - Provide many tools and services to teachers-designers

But

- Institutions impose a specific LMS to teachers
- Teachers are (sometimes) trained on how to use it
  - Not how to design learning situations on the LMSs
  - Not how to abstract instruction design from technical/administrative details
The GraphiT project

- **General informations**
  - Funded by the french national research agency (ANR)
  - Start/End: February 2012 / September 2015
  - Website: [http://www-lium.univ-lemans.fr/~laforcad/graphit/](http://www-lium.univ-lemans.fr/~laforcad/graphit/)
  - Involved several research members from our LIUM laboratory

- **Objectives**
  - Provide teachers with graphical learning design language
    - Compatible with LMS
  - Help to focus on the pedagogical aspect of the scenario
    - Instead of setting-up complex tools
  - Foster individual reflection about learning design
  - Improve uses of the existent LMSs

<table>
<thead>
<tr>
<th>Context</th>
<th>The GraphiT project</th>
<th>Our approach/Moodle case study</th>
<th>Conclusion &amp; Perspectives</th>
</tr>
</thead>
<tbody>
<tr>
<td>06/07/2015</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
The identification and the formalization process

- We define the necessary analysis and steps for the identification and formalization of an LMS instructional design language.

- It is specified according to three different viewpoints:
  - a viewpoint centred on macro-HMI
  - a functional viewpoint
  - a micro viewpoint.

- Formalism: the meta-model format
The identification and the formalization process: An overview

Context
The GraphiT project
Our approach/Moodle case study
Conclusion & Perspectives

06/07/2015
The macro IHM analysis

Objective: identify platform interfaces related to the Instructional Design (ID).
Moodle 2.4 macro-HMI analysis

Context The GraphiT project
Our approach/Moodle case study
Conclusion & Perspectives
An extract of Moodle 2.4 macro-HMI model
The factorization analysis

**Objective**: find common elements in pedagogical activities/resources and common relations between them.
An extract of Moodle 2.4 Moodle Macro model
The functional analysis

Objective: identify the functionalities dedicated to the course instructional design.
Moodle 2.4 functional analysis

Context The GraphiT project
Our approach/Moodle case study
Conclusion & Perspectives
An extract of Moodle 2.4 functional model

Context

The GraphiT project

Our approach/Moodle case study

Conclusion & Perspectives

06/07/2015
The micro analysis

**Objective**: takes into account two different viewpoints: micro-HMI and technical viewpoints.
The micro analysis
The micro HMI analysis (micro analysis)

**Objective**: identify all elements relevant to the instructional design, including their features (attributes, types, etc.).
Moodle 2.4 micro HMI analysis

Context

The GraphiT project

Our approach/Moodle case study

Conclusion & Perspectives
An extract of Moodle 2.4 micro HMI model
The technical analysis (micro analysis)

Objective: specify a reduced Conceptual Data Model from the one available by LMS providers.
Moodle 2.4 technical analysis

• This technical analysis consists in
  - (1) looking over all database tables in order to sketch a first draft of the model,
  - (2) focusing on tables embedding elements in relation to instructional design concepts.
An extract of Moodle 2.4 technical model

Context
The GraphiT project

Our approach/Moodle case study

Conclusion & Perspectives
The Confrontation & formalization (micro analysis)

**Objective**: allows the confrontation of both micro-HMI and technical models, and the formalization of the final model.
The Confrontation and formalization (micro analysis)

• The micro-HMI and technical models are compared in order to
  -(1) refine the micro-HMI model
  -(2) detect and correct the difference between models
  -(3) ensure that the final model can be easily bind to a
        computer-readable format for the existent LMS.

• Some differences or ambiguities are so identified. They require
  a deeper and finer analysis of both HMI and technical analysis.
  At this step, other technical-centred analysis (source code,
  backup packages, etc.) are used.
# The Confrontation and formalization

**Context**

The GraphiT project

**Our approach/Moodle case study**

**Conclusion & Perspectives**

---

<table>
<thead>
<tr>
<th>Context</th>
<th>The GraphiT project</th>
<th>Our approach/Moodle case study</th>
<th>Conclusion &amp; Perspectives</th>
</tr>
</thead>
<tbody>
<tr>
<td>06/07/2015</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
An extract of Moodle 2.4 final model
An extract of Moodle 2.0 final model
### Difference between Moodle 2.4 and Moodle 2.0 meta-models

<table>
<thead>
<tr>
<th></th>
<th>Moodle 2.4</th>
<th>Moodle 2.0</th>
<th>Comments</th>
</tr>
</thead>
<tbody>
<tr>
<td>External Tool class</td>
<td>Yes</td>
<td>No</td>
<td>The external tool activity module enables students to interact with learning resources and activities on other websites.</td>
</tr>
<tr>
<td>Book class</td>
<td>Yes</td>
<td>No</td>
<td>The book module enables a teacher to create a multi-page resource in a book-like format, with chapters and subchapters.</td>
</tr>
<tr>
<td>Relation between Section &amp; ActivityCompletionCondition classes</td>
<td>Yes</td>
<td>No</td>
<td>This relation determines any activity completion conditions which must be met in order to access the section.</td>
</tr>
<tr>
<td>Relation between Section and GradeCondition classes</td>
<td>Yes</td>
<td>No</td>
<td>This relation determines any grade conditions which must be met in order to access the activity.</td>
</tr>
<tr>
<td>Assignment class</td>
<td>Yes</td>
<td>Yes</td>
<td>In Moodle 2.4 we have 1 class for assignment (Assignment) while in Moodle 2.0 we have 4 classes for assignment (Online text, Advanced uploading files, Offline activity and Upload single file).</td>
</tr>
<tr>
<td>blindMarking attribute for Assignment class</td>
<td>Yes</td>
<td>No</td>
<td>Blind marking hides the identity of students to markers.</td>
</tr>
<tr>
<td>gradingMethodAssignment attribute for Assignment class</td>
<td>Yes</td>
<td>No</td>
<td>This attribute defines the advanced grading method (Simple direct grading, Marking guide, Rubric) used for calculating grades in the assignment.</td>
</tr>
<tr>
<td>Relation between Assignment and Grading classes</td>
<td>Yes</td>
<td>No</td>
<td>Students are able to collaborate on an assignment.</td>
</tr>
</tbody>
</table>
Conclusion & Perspectives

- Propose a meta-model-based approach and method for identifying and formalizing LMS languages.
- We apply our proposed method on the Moodle 2.4 platform.
- We have also applied our method on the Moodle 2.0 and Dokeos platform.
- The meta-model will be used as:
  - a basis for the development of the external editor.
  - A communication format between the editor and the LMS.
- This will facilitate the use of LMS and allow to teachers and pedagogical engineers of becoming more familiar with the specific design upon this LMS.
Making explicit the Moodle instructional design language

Thank you!

Contact emails:
nour.el_mawas@univ-lemans.fr
lahcen.oubahssi@univ-lemans.fr
pierre.laforcade@univ-lemans.fr