Web services plug-in to implement complex pedagogical set on LMS and Web application

Model Driven Engineering approach for eLearning

TENCompetence workshop
Trigone Laboratory, team NOCE
Université des Sciences et Technologies de Lille 1
Pierre-André Caron, Thomas Vantroys
Introduction

• New applications (blog, wiki, social network, collaborative tag, map, search engine...) : examples of new usages set up collaboratively or not by active users.

• E-learning 2.0 : pedagogical usage of theses applications

• To take into account these new usages in pedagogical way :
  – To define suited tools for teachers.
  – To propose objects for better rationality of teacher's preparatory work
Plan

• Our approach
  – Proposition
  – Approach

• Conceptual tools
  – What is a “dispositive”
  – To build "dispositives" on Web applications
  – Tools to model and contextualize "dispositives"

• Technical realization
  – To write server
  – To define platform model
  – To implement Constructor

• Evaluation and experimental results

• Conclusion and future works
Our approach

Generic web applications, not completely adapted to the teaching project the teacher wants to lead.

Our proposition:

→ to plug on these applications a pre-structuring model that facilitates their use in an educational context.
Our approach

- A software framework that allows building pedagogical "dispositives" on a web 2.0 application via the Web Service call
- Web 2.0 applications = emerging structuration mechanisms
- Our proposition identify three phases: designing model, defining context, and building it.
Conceptual tools: What is a “dispositive”

An intermediate mixed notion between use and concept

- The organization of technical means in order to realize a goal
- The expression of the designers and users intentionality

- Ambivalence between the resources put in work and “the floating and transverse intentionality”
- A cognitive landscape, tolerant to error
Conceptual tools: To build "dispositives" on Web applications

• Identifying it’s constructing functionalities.
• We propose to wrap these functionalities by services
  – Permits not to modify the code of the application
  – Provides a standard solution to address an application
  – Facilitate the phase of context definition
Conceptual tools: Tools to model and contextualize "dispositives"

From the expression of services it is possible to define the functionalities model of the Web application.

We propose to use this model for two different purposes:

- using it like a Meta model to get a specific modelling tool for this web application
- implements with this model the service generator compatible with the models produced
Technical realization: to write server

- We implement services to wrap web application API as Web Services
  - Useful technology for connecting a learning environment to organisational data and processes [Wilson 2005]
  - Easy, even trivial, to implement with PHP and Java
  - ELF initiatives present the different service-orientated approaches in the learning's sphere
  - Our proposition use and expand IMS Enterprise specification

- In our experimentation we address a WikiWeb application

- Also conducted on three of the most popular Open Source E-learning platforms (Ganesha, Claroline, Moodle) [Dooley 2006].
Technical realization: to define platform model

- Difficulty to define the technological meta-model of a domain [Marvie & al 2006], [Favre & Musset 2006]
- In the E-learning context this problem is reinforced by the diversity of platforms

Our approach:
- Limit functionalities.
- Identify element's factories.
- Define the mechanism of factorization.
Technical realization: to implement Constructor

Modeling tool

MOF

Meta model

Specific modeler model

Generic constructor

Specific constructor Collection of Object

Web based TOOL

Dispositiv WS Collection of Calls

Class implementation

Service stub implementation

Generic interface to construct
Evaluation and experimental results

• The PCDAI project aims to enable more active forms of learning over the Internet
  – The first phase of the project has been described previously in [Caron & al 2006], [Leclercq 2005]
  – Second phase features
    • New meta-model from fusion of meta-models previously made
    • Transformations are now automatic
    • Possibility to define the context of a "dispositive" with an emerging use and to compose the "dispositives" to construct
Evaluation and experimental results

Computer scientist and pedagogical community

Define for each platform a global meta Model, define modeler for Wikini

Implement specific constructor for Wikini

Web Services to wrap Wikini API

Teacher

Uses of GenDep to build and contextualize model

Dispositives on Wikini

separate different aspects of a global "dispositive".
Conclusion and future works

• To build pedagogical “dispositives” on web applications: **modelling, defining context and building**

• We propose to develop for each application:
  – a web service plug-in
  – a specific modeller
  – a specific constructor

• With an MDE approach, for a classic web application, the simple interpretation of the web services and the model generates nearly 6000 code lines automatically; only 300 lines are still to implement (Statistics from WikiniMST project)
Conclusion and future works

- Web services plug in will be easier to implements when Web application will be based on SOA architecture.
- The experimentations → to evolve our approach from explicit transformation mechanism used in the first phase to fusion of models.
- We are entering into a new phase of experimentation on the Clarolines and Moodles platforms, and ePortfolio Web applications
  - use of our engineering process,
  - explore new ways with the possibility to express methodologies
  - and to bind these methodologies to the generated models.
The experimentation can be reproduced see Gendep space at:
http://noce.univ-lille1.fr/projets/ModX

Question?