Automating Standards-Based Courseware Development Using UML

Andreas Papasalouros[1], Symeon Retalis[2], and Nikolaos Papaspyrou[1]

[1] National Technical University of Athens, 9 Heroon Polytechniou, 157 80 Zografou, Greece
{andpapas,nickie}@softlab.ntua.gr
[2] University of Piraeus, 80 Karaoli & Dimitriou, 185 34 Piraeus, Greece
retal@unipi.gr

Abstract. In this paper we discuss the automatic construction of web-based courseware applications from XML descriptions of appropriate UML models. The created applications conform to a related Learning Technology Standard for learning material interchange, namely the IMS Content Packaging standard.

1 Introduction

In this paper we demonstrate the automatic generation of courseware applications based on proper UML models. These models have been defined as a UML profile. A design model of a courseware application consists of three views or submodels:

- An *Activity Model*, which depicts the abstract solution of the learning problem under consideration in terms of activities that the learner is engaged in and resources, that is, physical, binary entities that are incorporated into the application, such as images or java applets.
- A *Navigation Model*, which describes the hypertext structure of the application by defining nodes and links and their mapping to the elements of the Activity Model.
- A *Presentation Model* which defines the user interface of the application in the form of templates for web pages.

This generation is based on specific technologies and specifications, such as the XML Metadata Interchange (XMI) [3] and XSL Transformations [4]. Furthermore, the produced courseware conforms to the IMS Content Packaging (CP) Specification [1] for defining interchangeable bundles of learning content.

2 The Transformation

The method for the automatic courseware generation is briefly described here. First, a designer creates a model for the application under development using

a general purpose UML tool. This model follows the structure presented in the previous section and provides an abstract representation of the content, structure and presentation of the application. The UML tool is used to provide an XML serialization of the model in XMI format. A number of UML tools are available with this capability. The designer also provides the *resources* which are physical files that will be incorporated into the application.

The courseware generation is carried out by a tool which we have developed. This tool is based on XSL Transformations and consists of a number of XSL files, called by a small Java application. The input to this tool, for a particular application, consists of an XMI description and associated resources. All the Activity, Navigation and Presentation Models are contained into the XMI description. The tool transforms the structural description of the Navigation Model into HTML code, incorporating the resources of the Activity Model and applying the templates of the Presentation Model.

The tool generates the code of the Courseware Application. This consists of a set of HTML pages. The tool also generates an XML description of the structure of the courseware that is called a *manifest file* [1]. The HTML Pages together with the manifest description constitute a bundle of learning material that conforms to the IMS Content Packaging Learning Technology standard. This bundle is reusable and portable. It can be easily deployed into proper web-based systems which conform to the same standard and be presented to learners.

3   Future Work

We are extending this approach for standards-based courseware generation in the direction of conditional sequencing of learning content, as it is specified in the IMS Simple Sequencing Specification [2]. Sequencing refers to the automatic selection of different activities for delivering to individual learners according to their history of interaction with the learning material, i.e. their performance in on-line tests. We are also developing a specialized tool for authoring of courseware UML models, that is expected to be more usable for model authors than general purpose UML tools.

References

1. IMS: IMS Content Packaging Version 1.1.3, http://www.imsproject.org (2003)
2. IMS: IMS Simple Sequencing Version 1.0, http://www.imsproject.org (2003)
3. OMG: The Unified Modeling Language v.1.5, http://www.omg.org/uml (2003)
4. W3C: XSL Transformations Version 1.0, http://www.w3.org/TR/xslt (1999)