Chapter IX
coUML:
A Visual Language for Modeling Cooperative Environments

Michael Derntl
University of Vienna, Austria

Renate Motschnig-Pitrik
University of Vienna, Austria

ABSTRACT

In this chapter we present coUML, a visual modeling language for cooperative environments. As modern instructional environments have a highly cooperative nature, coUML is proposed as a powerful and effective language for modeling instructional designs in such environments. Being based on UML, it was conceived and refined through application and experience over multiple years, primarily in a cooperative blended learning environment. We present relevant requirements and applications that contributed to the development of coUML, as well as a detailed specification of model elements, characteristics and features that describe its current state.

INTRODUCTION

This chapter presents the coUML approach to modeling of cooperative learning designs and environments. coUML stands for "cooperative UML." Its notation is based on UML, and it extends UML with a modeling profile specifically designed to enable the modeling of complex, cooperative learning environments. While coUML clearly focuses on process modeling in cooperative environments, it also allows modeling and integrating relevant structural information such as goals, documents, and involved roles.

While the name "coUML" was coined during the preparation of this chapter, initial ideas and uses of the coUML language date back to 2002, when we were starting an initiative to discover and document the e-learning practices at our department. During this project the coUML modeling approach proved to be a valuable aid in creating visual models of our teaching and learning activities for documentation, communication, research,

and dissemination purposes. The complete and user-friendly specification of coUML in this chapter along with illustrations and examples, is provided to make this approach available to interested readers and practitioners.

The chapter is structured as follows: In the next section we provide some background information on the roots and requirements of the coUML approach. In the third section a detailed specification of the coUML language is provided, illustrated with examples. In the fourth section we present three application scenarios of coUML. This is followed by a discussion on the coUML features and the presentation of a survey on visual instructional design modeling languages among blended learning experts. In the final section we present a conclusion and an outlook on further coUML-related activities.

THE coUML APPROACH

Background

The coUML approach emerged from practice (cf. Derntl & Motschnig-Pitrik, 2005). About 4 years ago, we were searching for a way to capture our teaching and learning designs. Our primary approach to designing the instructional processes for our courses was based on the principles of blended learning (Garrison & Kanuka, 2004). As a traditional university we build on face-to-face meetings in the courses, and we have gradually started introducing online and distant means of collaboration, evaluation, and delivery into our teaching and learning activities. The goal then was to build a comprehensive library of blended course designs or patterns including verbal descriptions and semi-formal models of scenarios that were already in use at our department. As no visual modeling language was particularly suited for such a task, we started to employ the following simple procedure: First, we write down a verbal description of a course and its activities, including an outline of relevant teaching and learning goals, and the primary teaching approach employed (e.g., project-based learning). The second step is to visualize the course scenario as one or more threads of activities according to the course description. Initially, we used simple symbols for drawing activities and arrows as connectors between activities. Gradually the notation system evolved from requirements drawn from practice and experience, and was finally based on a more formal, standardized notation system. Additionally it was apparent that the current wave of Web-based tools and enhancements not only penetrated educational environments, but any environment where people cooperate to achieve personal and organizational goals, e.g., in projects or communities. Therefore we present coUML as a language that is rooted in, yet not constrained to educational environments.

Requirements and Need

It was clear that a course was, conceptually, not just a sequential thread of activities; we would need additional control-flow structures such as decisions, concurrent flows, and composite activities. We were also interested in a number of additional, instructionally relevant information to be included in the visual models of teaching and learning activities, which are outlined in the following list of requirements:

R1: *Support for logical/temporal arrangement of activities, as well as decisions and concurrent activities.*

R2: *It should be possible to model activities at different levels of detail, which requires a means of refining composite activities. This should allow multiple views on complex activities and help to keep the models clear and understandable, even though this introduces layers of abstraction which might be difficult to grasp for many people.*