A Component-Based Distributed Architecture for Adaptive Web-Based Education

Peter Brusilovsky
School of Information Sciences, University of Pittsburgh
Pittsburgh PA 15260
peterb@mail.sis.pitt.edu

Abstract: This paper presents KnowledgeTree, an architecture for adaptive eLearning based on distributed re-usable intelligent learning activities. The goal of KnowledgeTree is to bridge the gap between the modern approach to Web-based education based on courseware management systems and powerful but underused intelligent tutoring and adaptive hypermedia technologies. This architecture attempts to address both the component-based development of adaptive systems and the teacher-level reusability.

Keywords: Component architecture, student model server, Web-based education, courseware management system, learning portal, adaptive hypermedia.

Adaptive Web-based Educational systems (AWBES) have brought to the Web a number of powerful technologies [1] developed originally in the fields of intelligent tutoring systems (ITS) and adaptive hypermedia (AH). Yet, the absolute majority of Web-enhanced courses rely on so-called course management systems (CMS) such as Blackboard or WebCT. How AWBES can capitalize on the success of CMS in reaching real classrooms? Should our community just copy the CMS approach by developing "more intelligent" CMS? The author is convinced that instead of developing another generation of "monolithic" systems the efforts have to be centered on developing distributed component-based architectures for building AWBES. A distributed architecture should bridge the gap between the modern approach to Web-based education (based on CMS and educational repositories) and powerful but underused AWBES. This paper presents KnowledgeTree, a distributed architecture for adaptive E-learning that we have been developing over the last 2 years and that has already been used in several courses at the University of Pittsburgh. This architecture attempts to address both the component-based development of adaptive systems and the teacher-level reusability. The work on KnowledgeTree continues our past stream of research on component-based and distributed architectures for ITS [2; 3].

The KnowledgeTree is an architecture for adaptive eLearning based on distributed re-usable intelligent learning activities. Capitalizing on the success of integrated CMS, KnowledgeTree aims to provide a one-stop comprehensive support of teachers' and students' needs in eLearning. It doing so it attempts to replace the current monolithic Blackbord -style CMS with a community of distributed communicating servers. The architecture anticipates the presence of at least three kinds of servers: activity servers, learning portals, and student model servers (Figure 1).

A *learning portal* in our vision plays a role similar to modern CMS in two aspects. First, it provides a centralized single-login point for the student to work with all learning tools and content fragments that can be distributed over the Web. Second, it allows a teacher to structure the access to various distributed fragments according to the needs of a specific course. A portal is a component of the architecture that is centered around a *course*. Quite similar to a CMS it provides some course-authoring interface for a teacher and maintains a runtime interface with a student. The difference with CMS is the separation of the unique course structure from the reusable course content. Both the learning content and serviced (altogether called *activities*) used by the students reside not in the portal, but in multiple distributed *activity servers*. A portal has an ability to query activity servers for relevant activities and launch remote activities selected by students.

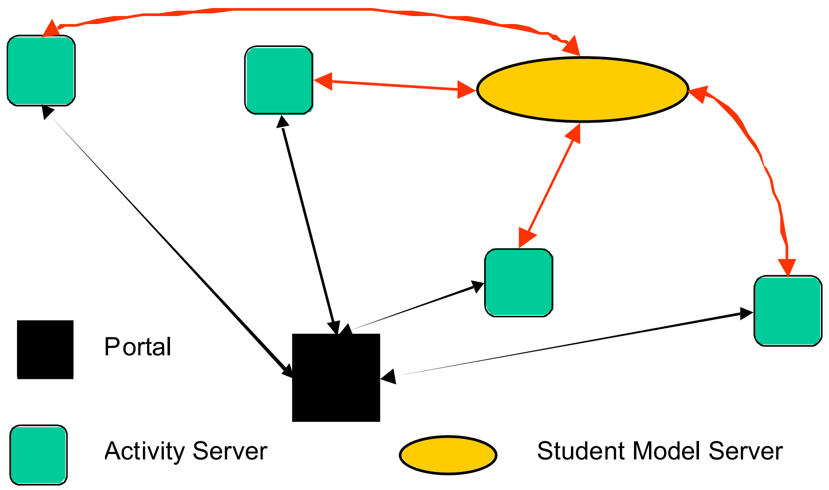

Figure 1: Main components of the KnowledgeTree distributed architecture.

An *activity server* is the component of the architecture that is centered around reusable *content and services*. It plays a role similar to an educational repository in modern courseware re-usability approach in the sense that it hosts reusable learning content. Unlike repositories that are pools for storing simple learning objects that can be copied and inserted into courses, an activity server not only stores, but also delivers the activities. The need for activity servers stems from the nature of adaptive and other advanced learning activities - such as ELM-ART problems. These activities just can't be copied as files, they have to be served by a dedicated Web servers maintained by the content providers. The duty of an activity server is to answer portal's requests for specific activities and to provide a complete support for a student working with each of the activities residing on the server. In particular, by turning an ITS into an activity server it is possible to make intelligent activities, traditionally encapsulated inside an ITS, open to multiple re-use.

Student model server is a component centered on an *individual student*. It collects data about student performance from each portal and each activity server and provides information about the student to adaptive portals and activity servers that are able to personalize their communication with the student. We think that the presence of multiple adaptive activities requires a centralized user modeling architecture that enables each server to get an access to all information about student progress. The problem of centralized student modeling servers has been investigated in a number of earlier projects dealing with multiple intelligent educational agents [2; 4], but Web context poses new requirements to the student model servers. Our user modeling server CUMULATE and Personis server suggested by Kay, Kummerfeld, and Lauder [5] are good examples of student model servers for distributed Web adaptive systems.

With the KnowledgeTree architecture, a teacher develops a course using one portal and many activity servers. A student works through the portal serving this course, but interacts with many learning activities served directly by various activity servers. The student model server provides a basis for performance monitoring and adaptivity in this distributed context. The KnowledgeTree architecture is open and flexible. It allows the presence of multiple portals, activity servers, and student modeling servers. The open nature or it allows even small research groups or companies to be "players" in the new E-learning market. An activity server that provides some specific innovative learning activities can be immediately used in multiple courses served by different portals. An innovative portal with a good interface can successfully compete with other portals since it has an access to the same set of resources as other portals. A more powerful student model server can successfully replace older servers.

The most critical component of a distributed architecture is a set of communication protocols. The KnowledgeTree is based on several standardized http-based communication protocols: a protocol for transparent login and authentication, a protocol for a portal to send a query to the activity servers, a protocol for the activity servers to respond, a protocol for an activity server to send the information about the student progress to the student model server, and a protocol to request information about the student from the student model. The current implementation of KnowledgeTree includes one portal (KnolwedgeTree), one student modeling server (CUMULATE) and four activity servers. All servers interact with each other using our standardized protocols. The whole architecture has been already used for four consecutive semesters in the context of two different programming courses at our department. We have also performed several classroom studies of different activity servers servers. More information about the practical side of KnowledgeTree architecture and links to related publications can be found in [6]. The readers are welcome to try the current version of all system components available on our lab home pages: http://www2.sis.pitt.edu/~taler/.

References

[1] Brusilovsky, P. Adaptive and Intelligent Technologies for Web-based Education. *Künstliche Intelligenz* (4), 1999, 19-25. Available at http://www2.sis.pitt.edu/~peterb/papers/KI-review.html.

[2] Brusilovsky, P. Student model centered architecture for intelligent learning environment. *Proceedings of Fourth International Conference on User Modeling* Hyannis, MA, 1994, 31-36.

[3] Brusilovsky, P., Ritter, S., and Schwarz, E. Distributed intelligent tutoring on the Web. In: du Boulay, B. and Mizoguchi, R., eds. *Artificial Intelligence in Education: Knowledge and Media in Learning Systems*. Amsterdam: IOS, 1997, 482-489.

[4] Machado, I., Martins, A., and Paiva, A. One for all and all for one: a learner modelling server in a multi-agent platform. In: Kay, J., ed. Wien: SpringerWienNewYork, 1999, 211-221.

[5] Kay, J., Kummerfeld, B., and Lauder, P. Personis: A server for user modeling. *Proceedings of Second International Conference on Adaptive Hypermedia and Adaptive Web-Based Systems (AH'2002)* Málaga, Spain, 2002, 201-212.

[6] Brusilovsky, P. and Nijhawan, H. A Framework for Adaptive E-Learning Based on Distributed Re-usable Learning Activities. *Proceedings of World Conference on E-Learning, E-Learn 2002* Montreal, Canada, 2002, 154-161.