NLII Spring 2003 Focus Session: Next-Generation Course Management Systems

Cosponsored by the University of Arizona

Readings, materials, agenda, and meeting proceedings available at http://www.educause.edu/nlii/meetings/nlii032/.

Applying the conceptual framework developed by NLII 2001 fellow Colleen Carmean, participants at the NLII Spring Focus session titled Next-Generation Course Management Systems were able to identify problems that inhibit the use of course management systems (CMSs) to promote deep learning and the next-generation CMS features, functionality, and policy improvements that would address those problems. Participants met in four groups—teaching and learning, technology and architecture, management and systems, and the NLII CMS Observation Tool and Glossary—to discuss and prioritize the suggestions.

Features Important to Teaching and Learning

- **Cross-course functionality**: Enable students and teachers to share content, tools, and spaces across courses.
- **Course is no longer the container**: Improve connections to materials, and facilitate interaction across courses.
- **Peer interaction**: Create new and better tools to support peer feedback, evaluation, group work, and exchange of resources.
- **Collaboration tools**: Empower students to collaborate with each other and to coteach with the instructor.
- **Learning tools and student wizards**: Use artificial intelligence techniques to customize the learning experience by offering suggestions and content based on a student’s use of the system.
- **Blogging and journaling tools**: Create tools for public and private journaling and reflection and build in mechanisms for peers and teachers to respond.
• **Faculty pedagogy advising:** Use agents and wizards to help faculty create courses that promote deep learning and that provide flexible content to address the diverse needs of learners. Support locally developed wizards.

• **Equation editing:** Give better support for mathematical notation systems.

• **Persistence and portability:** Create student and teacher access to courses and artifacts across systems and beyond the time boundaries of a particular course in order to extend learning over time.

• **Access:** Increase content authoring capabilities for all types of users.

• **Annotation tools:** Make it easier for students to annotate documents, add associated reflections to a document over time, and control access to their comments.

**Features Important to Technology and Architecture**

• **Integration and extensibility:** Improve interoperability via e-mail, student information, and other administrative computing systems; integrate third-party tools, such as electronic portfolio systems and digital library clients; and build custom tools that access the core services of the course management system.

• **Intelligent agents:** Make the system recognize and automate repeated tasks to help learners and teachers become more efficient.

• **Digital rights management:** Enable student work to be student owned. Students should be able to control access to their work, export it to other systems, and download it in a form that captures its context to be reusable elsewhere.

• **Multiple levels of authoring interface complexity:** Provide new users with templates and other scaffolding to make it easy for them to author pedagogically sound course content and structure. Enable advanced authors to access and modify the business logic and user interface in more-sophisticated ways in order to accommodate pedagogies such as problem-based learning.

• **Content Interoperability:** Enable learning objects and other interactive content objects to plug into multiple applications.
Features Important to Management and Systems

- **Organizational structure and institutional structures:** CMS success is dependent on alignment of institutional structures and processes. Such alignment requires a clear institutional vision, which leads to clear statements of policy about how the institution will report, reward, and support technology products and services development. The policies should balance a hierarchy for efficiency and autonomy to encourage innovation. The CMS must accommodate that balance across organizational models and departments.

- **Copyright issues:** Standards for copyright are needed and should address the use of student work for assessment. Move beyond ownership and controlling access to more-open licensing policies.

- **Consortia:** Institutions need to find new ways to work together.

During summer 2003 the NLII staff and Course Management Systems working group will be refining these issues and working with various communities of practice to develop a final draft of a consensual set of functional requirements for next-generation course management systems. The final draft will be provided to instructional management system working groups and course management software developers in fall 2003.

Focus Session: Work Products and Next Steps

- Refine the conceptual framework (see [http://www.educause.edu/nlii/keythemes/lcp/](http://www.educause.edu/nlii/keythemes/lcp/)) that was designed to provide a common framework so diverse participants could have productive dialogues about the use of technology. Final draft due fall quarter 2003.

- Create a glossary of terms (see updated glossary at [http://educ3.utsa.edu/pmcgee/nlii/NLII_glo.rtf](http://educ3.utsa.edu/pmcgee/nlii/NLII_glo.rtf)), purpose of which is, again, to facilitate productive dialogues about the use of technology, in particular course management systems. Final draft due fall quarter 2003.

- Learner-Centered Principles Observation Tool (see updated version at [http://educ3.utsa.edu/pmcgee/nlii/CMS_V2.htm](http://educ3.utsa.edu/pmcgee/nlii/CMS_V2.htm)), designed to help an observer be reflective about deeper learning principles in the context of an actual course supported by a course management system. Final draft due Fall 2003.
- Prioritized set of issues and recommendations associated with features of next-generation CMSs according to learner-centered principles. First draft currently in meeting notes, but list to be extracted and refined into a separate work product to be shared with CMS developers Fall 2003.