Online Course Material Interoperability and Tutorial Module for Moodle

Senele Goba sgoba@cs.uct.ac.za
Nimrod Agasi nagasi@cs.uct.ac.za
Gareth Saul gsaul@cs.uct.ac.za
Donald Cook sgoba@cs.uct.ac.za

Technical Report CS04-24-00
Department of Computer Science
University of Cape Town

ABSTRACT

Due to the increase in online education, organisations are now seeking to reach new levels of quality education by implementing learning management systems. There are various factors that these organisations must take into account. Some of these factors include costs, implementation, development and evaluation of the software.

This report looks at various existing learning management systems and explores their strengths and weaknesses. A thorough evaluation showed that the Moodle Learning Management System offered the most benefits in terms of the services and features that it offers to its users. Moodle is an open source software solution that is entirely free to use. Since its inception thousands of Moodle sites have been registered and the software has been developing as open source. Because of its open source nature, Moodle has been improved by the addition of new modules to provide greater functionality. The addition of two such modules, one for enhanced tutorial management that addresses assignment hand in, mark management (etc.) and a second that provides interoperability with 3rd party courseware. In addition, this report documents modification of the Moodle LMS in terms of the requirements, design and integration of a tutorial and interoperability modules into the Moodle LMS.

1. INTRODUCTION

The Department of Computer Science at the University of Cape Town has a proven track record in employing technology to support innovation in teaching and learning. The current system is such that each course is provided with its own website. PowerPoint presentation, slides, supplementary notes are all stored on these sites and are available for download. Over the last few years, many different web sites and tutorial programs have been created. However, none have been able to meet all of the department's requirements.

We have been tasked with enhancing a learning management system that can satisfy the needs of the Department of Computer Science at the University of Cape Town. However, at the phenomenal rate the e-learning market has been growing, and as numerous new companies and products become available, it also becomes more difficult to identify what could potentially work and fits our requirements, from what could be too expensive, too-complex or simply too difficult to implement.

The objectives of this project are two fold. Firstly it aims to provide a unified system that provides a functional tutorial management system as well as a course material interoperability utility. Moodle was chosen as the LMS that best suited this project's requirements due to is strong modular architecture and strong grounding on social constructionist theories.

2. BACKROUND

2.1 Virtual Learning Environments

A Virtual Learning Environment (VLE) is an IT application designed to facilitate pedagogical communication amongst those participating in an educational process, whether it be distance learning, classroom-based or any combination of the two. The purpose of a VLE is to distribute educational material in digital format and to provide access to online debates and discussions on aspects of the course study program, to introduce relevant content from the Web or to enable external experts or professionals to participate in debates or talks.

2.2 Standards

Standards form the foundation of any serious attempt to provide interoperability amongst E-learning products. Standards for LMSs allow students, faculty members, administrators and other interested parties access to a much wider variety of content than would otherwise be available. Many of the standards in existence currently cover not only the content itself, but also issues related to its use, accessibility and intended function.

Sharing content between different Learning Management Systems allows the time taken to create content to be reduced. It aims to increase the overall quality of E-learning by allowing the best material to be easily made available to a wide audience, thus increasing the effectiveness of instruction. This benefits both the students and the educators. The costs of providing E-learning should also decrease, as a result of the reduced effort required [1].

2.3 Interoperability

Traditionally, an LMS has been designed and developed as a monolithic entity, without much consideration of

interoperability issues. The recent increasing focus on interoperability has lead to the desire to have LMSs which provide architectural interoperability. Architectural interoperability refers to the ability of individual components of LMSs to interact with one another in a loosely-coupled manner. Ideally, the entire LMS should be constructed in this manner. This allows components of an LMS to be modified or replaced without compromising the LMS as a whole. The intended objective of providing architectural interoperability is to increase the functionality and flexibility of a LMS.

2.3.1 Open Knowledge Initiative (OKI)

The Open Knowledge Initiative from the Massachusetts Institute of Technology aims to provide a clear set of interfaces designed to provide a modular framework for architectural interoperability [2]. The OKI project aims to provide interoperability both between components of the LMS and with enterprise systems which may already be in place. [3]

2.3.2 Shareable Content Object Reference Model

SCORM is a wide framework covering both how to describe course material, but also how to use it to providing learning assistance, and how to track the progress of learners. SCORM is the result of the Advanced Distributed Learning project initiated by the United States Department of Defense and White House Office of Science and Technology Policy. The project was launched in 1997, and the first version of SCORM was released in 2000, and has since been followed by a more current 2004 version. SCORM is not a single specification, but rather a library of specifications from different producers which together outline a more general framework [3].

2.4 LMS, CMS and LCMS

2.4.1 LMS

These systems are used to simplify the administration of learning programs within an organization or institution. LMSs manage the log-in of registered users, manage course catalogues, record data from learners, and provide reports to the appropriate people. They typically do not have the ability to create content.

2.4.2 CMS

These are systems that are used to simplify the creation and administration of learning content. A CMS allows content to be stored, retrieved, edited, updated and then outputted in a variety of ways.

2.4.3 LCMS

LCMS is a mostly web-based system that is used to author, approve, publish, and manage learning. A LCMS combines the administrative and management aspects of a regular LMS with the content creation and personalised assembly dimensions of a CMS.

2.5 Existing LMSs

2.5.1 ATutor

ATutor is a system that provides good documentation, easy installation and strong potential for development. The user interface can be non-intuitive at times, but the overall functionality is good, the design is modular and ATutor is dedicated to standards. It is strong on standards and can import external content in IMS/SCORM format.

2.5.2 Moodle

Moodle is one of the most user -friendly and adaptable open source courseware products currently available. It has excellent documentation, strong support for security and administration, and is now evolving towards IMS/SCORM standards. The essence of Moodle's success is that it is developed with both pedagogy and technology in mind. Moodle has a strong development and user community.

3. APPROACH

3.1 Tutorial Module

3.1.1 Interviews

User interviews form the basis of the project design. These interviews attempt to extract the requirements that the end user is looking for from the system. These include aspects such as user expectation of the system, as well as how they would interact with it. These interviews were performed on the various end users of this system. These included administrative staff, teaching assistants, markers and students.

3.1.2 Questionnaires

A survey named COLLES (Constructivist On-Line Learning Environment Survey) was then modified, and given to the participants. The COLLES survey was designed to help assess key questions about the quality of an online learning environment with a social constructivist viewpoint [4]

3.1.3 Heuristic Evaluation

A Heuristic evaluation will be performed as a structured inspection of the interface design for usability. The goal of this evaluation was to find the usability problems in the design so that they could be examined and attended to. Five evaluators were used on a subset of the heuristics.

3.1.4 Integration

This illustration clarifies the general system operations. PHP scripts are used to display the elements of the HTML forms. This uses the Moodle API. PHP functions are then used to execute MySQL queries as well as PHP functions that check and index Moodle data files.

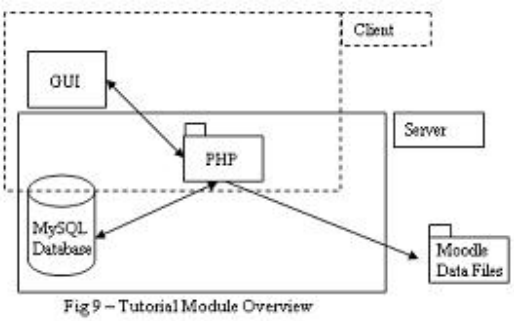

Fig 9 – Tutorial Module Overview

Of the interactions illus trated in the previous diagram, database integration is the most crucial. The Moodle database is required to control a large amount of information and ensuring consistency of the information is essential. The

following entity relationship diagram shows relationships that affect online assignment submission. Each assignment comes under a topic, which may be an online test, an exam, a lab assignment, etc. Each section should have assignment categories associated with it. Student groups can be made for assignment submissions that are under the same assignment category. Student groups may be divided into sections and an assignment may require individual and/or group submission. All the information about each course is distributed between two databases.

3.2 Interoperability Module

3.2.1 Requirement

Most third part courseware that is available is provided in specific formats for each different LMS. Of these, the most widely useful is the IMS content standard, which is supported by a number of commercial LMSs, including WebCT. Thus, to provide effective interoperability with this material in Moodle, we must provide support for this standard.

3.2.2 Moodle Restore Process

The Moodle software already has a system for performing backups and restores. Careful examination of the existing system and supporting documentation allows us to understand the similarities that exist between the normal Moodle backup and an IMS content package. Both make use of similar XML manifests to store information, and a number of careful transformations allow the use of IMS content within Moodle.

The IMS content specification supports exports that contain substantially less information than Moodle normally requires. These results in sections of the information provided to Moodle being supplied with defaults or with user-supplied preferences. Rather than being a disadvantage, it can be used to provide an additional level of customizability and flexibility to the use of the imported content.

3.2.3 Integration

Rather than developing a standalone tool that converts IMS content into a Moodle backup archive, the resulting product consisted of a series of modifications to the Moodle restore process. Early in the restore process Moodle detects IMS content, and executes an alternate set of restore functions. Existing Moodle content is handled as per normal, and IMS content is handled with dedicated procedures. This allows the IMS import process to make use of advanced features of the Moodle engine, as well as not requiring any additional features on the server beyond those already in use.

4. RESULTS

4.1 Tutorial Module

The prototypes that were done during the requirements and design phase form part of the cognitive analysis of the project. This involved designing the system, testing it by communicating with the users and redesigning it to suit better the needs of the users. The prototypes were improved upon and used at the implementation phase. After implementation, user testing was conducted as part of validation testing, to ensure that what was implemented reflected what was expected.

The users, which included student, tutors and teaching assistant tested the system by performing some tasks and then filled in the form to evaluate the system.

The results showed that 100% of the tasks were accomplished. The system did not show 100% usability. The lowest score for usability was 76.9%. 78% of the users preferred the new system to the original disjoint systems they were using.

The users highlighted that they like the feature of being able to query the assignment marks online. This is a good feature of the system as it encourages the transparency on the marker and encourages the student to have a clear understanding of their feedback.

Stress testing was then performed to analyse the scalability of the system. The following results were obtained.

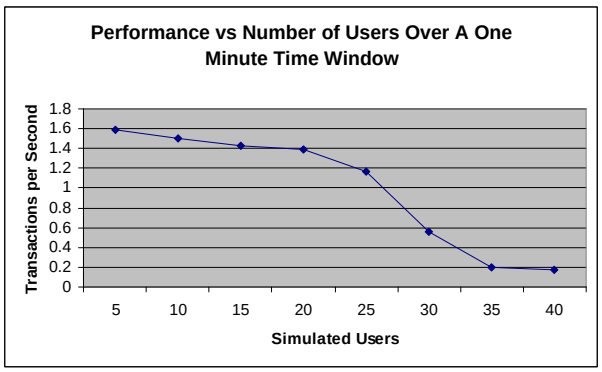

Fig 1 – System Slowdown over One Minute Intervals

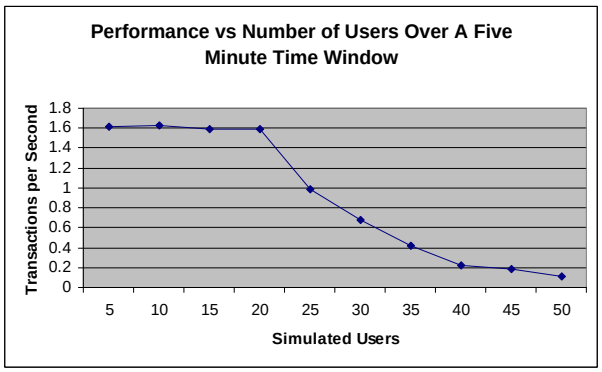

Fig 2 – System Slowdown over Five Minute Intervals

4.2 Interoperability Module

Testing was performed on the IMS content import system to ensure that the imported content was useable within the Moodle environment. The tests indicated that it was quite feasible to extend Moodle with support for the IMS content specification.

Whilst the content that was imported was fully available from the Moodle environment, a few issues were raised du ring the user tests.

Of most notable concern was the fact that the modifications to the content were not preserved. Essentially, since the import process transformed information to make it useable by Moodle rather than working with the original content pac kage, any modifications made from Moodle were localised in scope. Without an export function these modifications could not be shared with other members of the academic community.

5. CONCLUSIONS

5.1 Tutorial Module

Most of the users preferred the new unified system to the current systems they are using. The interesting finding was that the users who wholly preferred the new system had been using the most disjointed system, i.e. the first year system.

These scalability graphs suggest that Moodle was unable to handle large amounts of simultaneous users and therefore scaled badly. However, Moodle has been shown to work in large universities with over 1 7,000 users [5]. These universities use sophisticated servers with distributed databases as well as complex load balan cing techniques. After discussing this issue with Matthew West, a system administrator in the Department, it was agreed that this lack of scalability is in fact a flaw in the PHP language as well as the basic nature of the Moodle server used.

5.2 Interoperability Module

In completing this project it was examined whether it was possible to extend an open -source Learning Management System to support interoperability and standards -compliance. The project focused attention on the IMS specification, particularly as it is used by the WebCT package for exporting content.

We were able to conclude that it was quite possible to extend Moodle to support and handle IMS compliant content. Issues did arise as a result of Moodle not being developed with IMS / SCORM content i n mind. Content occasionally had to be reorganised to fit into the Moodle course structure. In general, however, these were not large problems.

This project presented some interesting results. It demonstrated that using an open -source Learning Management S ystem held a number of advantages where content interoperability is concerned. Moodle, in particular, benefits from having a highly modular structure that allows the addition of new functionality to be accomplished with minimal difficulty. It seems apparen t that adding support for standards other than IMS should be feasible.

6. FUTURE WORK

The tutorial module could be extended or appended to add automated marking for the assignments. This would make the marking more moderated and consistent. It would also make the assignment marking process to be more efficient.

Testing of the tutorial module was done on a small sample of students therefore it cannot give clear and definite conclusions about the system. A larger sample would be best to test such a system and would give a more definitive conclusion.

As was shown earlier, problems were discovered with PHP functionality as well as the basic nature of the Moodle server. Some solutions could include exploiting multiple databases using load balancing techniques. This should rectify the scalability problems. Furthermore, moving towards a java based system may produce a more stable system. Java beans provide a completely integrated system for management and administration of the system. This also allows for transaction management which allows for failed requests to be rolled back.

The interoperability module has a number of potential options for future work. Providing a corresponding export module to complement the import function would allow Moodle developed content as well as modified IMS content to be made available to users of other LMS systems. As a more ambitious project, it would be possible to extend the Moodle architecture to directly work with the IMS content packages. Doing this would make any changes to the content to be immediately available.

IMS has a number of specifications available. In this project we focused on the content specification, but in the academic community there have been a number of requests for the Question and Test Interoperability specific ation to be provided for the Moodle environment. Implementing this specification would allow the majority of important content currently stored in LMSs to be freely transferable. [6]

7. REFERENCES

[1] Brown, S., and Fallon, C. *E-Learning Standards: A Guide to Purchasing, Developing and Deploying Standards-Conformant E-Learning*. St. Lucie Press, Boca Raton FLA, 2003.

[2] Open Knowledge Initiative. *Open Knowledge Initiative Final Report*, Massachusetts Institute of Technology, 2004. Available at http://www.okiproject.org/documents/%20%20OKI%20Report%20to%20Mellon.pdf

[3] Advanced Distributed Learning. *Sharable Content Object Reference Model (SCORM®) 2004 2nd Edition Overview*. Available at http://www.adlnet.org/.

[4] Taylor, P. and Maor, D. Assessing the efficacy of online teaching with the Constructivist On-Line Learning Environment Survey. *Proceedings of the 9th annual teaching learning forum*, (Winter 2000).

[5] Koontz, B. Breaking the Chains, Moodle – An Open Source Distance Learning Environment. *Texas Distance Learning Association* (Irving TX, March 2004)

[6] IMS Global Consortium. *IMS Specifications*. Available at http://www.imsglobal.org/.