

Work in Progress - Using Information Technology to Author, Administer, and Evaluate Performance-Based Assessments

Mark Urban-Lurain¹, Guy Albertelli², and Gerd Kortemeyer³

Abstract - Performance-based assessment (PBA) require students to construct responses in an “authentic” problem-solving context. However, PBAs are traditionally labor intensive to create, administer, and score. We are extending previous work using information technology (IT) to author, create, and deliver PBAs by incorporating this functionality into an existing open-source learning management system (LON-CAPA.) The LON-CAPA system permits the delivery of individualized PBAs to the students’ desktops, where they use other computer software to create files (e.g., spreadsheets, word-processing files, etc.) as solutions to the PBA and electronically submit those files to the system. The system then provides Teaching Assistants with the students’ files and the individualized scoring rubrics used to assess each student’s performance. This hybrid use of IT permits the use of complex, authentic tasks that cannot be evaluated by a computer, by providing a cost-effective means to leverage TA resources.

Index Terms – Performance-Based Assessment (PBA), Instructional Technology, Hybrid technology, Open Source Software, Learning Management Software.

INTRODUCTION

Performance-based assessments (PBA) require students to construct responses in “authentic” problem-solving contexts [1]. PBAs can support student learning by re-focusing the instruction and learning activities on the fundamentals of a discipline, a process that requires deeper understanding and can expose student thinking, providing opportunities for formative evaluation of learning [7] and help students progress from acclimation towards competence. However, scoring PBAs entails human judgment. They are therefore more expensive and time-consuming to score than “objective” tests [2]. Preliminary work by members of the project team has demonstrated that IT can scaffold the creation, delivery, and evaluation of PBAs in an introductory computer science course.

In 1996, a team from the Department of Computer Science and Engineering (CSE) at Michigan State University (MSU) designed a new introductory computing course, to

serve 2000 non-computer science students per semester. The course aligned assessment with instruction using a modified-mastery PBA system called Bridge Tasks [3]. Bridge Tasks (BTs) are criterion-referenced PBAs in which students solve a series of authentic computing problems.

BTs are dynamically generated from a database and delivered as a Web page to the student’s computer screen in a supervised laboratory. Security is maintained by software that requires a proctor to authenticate each student after verifying the student’s identity against a picture ID.

Evaluating 14,000+ BTs per semester requires a stringent set of scoring rubrics used by human graders. The software retrieves each student’s files and presents the grader the appropriate scoring rubrics for the particular items on that BT [3,4]. The grader evaluates each of the rubric criteria on a pass/fail basis and adds open-ended formative feedback for the student when appropriate. After all rubric criteria are evaluated, the database computes the overall BT outcome based on the individual pass/fail scores on each of the rubric criteria as defined by the instructor for each BT. Students can then view their scored BT with the detailed rubrics and any grader comments on the Web.

IMPLEMENTING PBAS IN OTHER COURSES

Evaluation of the outcomes in our CSE course over several years demonstrated that a course designed around PBAs could be implemented at a large scale in a cost-effective manner that results in improved student outcomes [5,6]. However, it was difficult to implement this system in other courses using the software infrastructure we created. The database backend and ODBC drivers needed for the grading client software are commercial products that are licensed on a per machine basis. To support the administrative requirements of the course, we added many of the functions of standard Course or Learning Management software packages, such as communications, student record keeping, etc. These were impediments to faculty implementing PBAs in other courses.

An Open Source Solution

During the time that the CSE team was developing the BT software, a team from the College of Natural Science was creating the Learning Online Network with Computer-

¹ Mark Urban-Lurain, Division of Science and Mathematics Education, Michigan State University, urban@msu.edu

² Guy Albertelli, Division of Science and Mathematics Education, Michigan State University, albertel@msu.edu

³ Gerd Kortemeyer, Lyman Briggs School of Science & Division of Science and Mathematics Education, Michigan State University, kortemey@msu.edu

Assisted Personalized Approach (LON-CAPA), an open-source, Web-based Learning Management System (LMS) to support large-enrollment science courses. (Detailed information is available at www.loncapa.org.) LON-CAPA is a full-featured, Web-based course management system similar to commercial systems; in addition, it has the following features:

- Content sharing and content reusability, including a network of shared resources from 27 participating institutions, a shared repository of 61,000 resources including assessment questions and multimedia content, and the ability to add new resources.
- Creation and grading of randomized homework, quizzes or exams, including the ability to create sophisticated question types such as: click-on image, random label, random plot, or formula response, reduce copying of answers by giving a different version of the problems to each student, and contextualized threaded discussions.

The Development Effort

A team of developers from LON-CAPA and the Department of Computer Science analyzed the requirements for porting the BTs into LON-CAPA and determined that LON-CAPA had the basic infrastructure needed to support most of the LMS requirements for supporting BTs. Although the BT software was implemented using a relational database, the authoring structure was built on XML. Since LON-CAPA resources are based on XML and already supported randomization of assessments, the basic data structures needed were already in place.

The primary development effort has focused on:

- **Support for the mastery model nature of the BT assessments.** The various parts of each BT are evaluated on a pass/fail basis. These are combined in an instructor-specified manner to determine if the student passes or fails the BT. Students must repeat a given level of BT until they pass before moving to the next BT.
- **Support for BT grading rubrics.** There are detailed scoring rubrics created by the instructor for each part of each BT. The graders need to evaluate each student's files based on the particular rubrics for that randomly generated BT. The grader evaluates each part of the BT on a pass/fail basis and must be able to provide additional comments to the student.
- **Support for proctored authentication.** LON-CAPA already supports fine-grained scheduling of resource availability. However, online resources are typically completed by students as homework without proctors. BTs are administered in scheduled labs with a proctor who validates the students' identities and enters a password to allow the student to receive a BT. This system also requires the ability to schedule BTs in particular labs at particular dates and times.

Rollout

At the time of writing (end of May, 2005) we have completed the initial development effort on these areas. The CSE faculty

are beginning to test the system using existing XML-based BT resources. The preliminary version of the BT capabilities will be in version 2.0 of LON-CAPA, scheduled for release in August, 2005.

The CSE faculty will do a parallel test of this release during fall, 2005, with a small number (about 100) of students. We will report on the current state of testing in the presentation of this paper at FIE. If all goes as planned, the CSE department will convert the CSE course to this system in spring semester, 2006.

FUTURE WORK

Including support for PBAs in an open-source, full-featured LMS will allow faculty from a range of disciplines to adopt PBAs in their courses. After the conversion of the current CSE course, faculty in other CSE courses are planning to adopt LON-CAPA and implement PBAs. We are working with faculty in Teacher Education to incorporate PBAs as part of the technology requirement for pre-service teachers. We look forward to working with other faculty to implement and evaluate the impact of PBAs on student problem-solving across a number of disciplines.

ACKNOWLEDGMENT

This material is based upon work supported by the U.S. Department of Education, under special project number P342A030070, and the National Science Foundation through the Information Technology Research (ITR 0085921) and the Assessment of Student Achievement (ASA 0243126) programs. We are grateful to Michigan State University for over a decade of encouragement and support. Any opinion, finding, conclusions or recommendations expressed in this publication are those of author(s) and do not necessarily reflect the views of any of the supporting institutions.

REFERENCES

- [1] Baxter, G.P. and Glaser, R., "An approach to analyzing the cognitive complexity of science performance assessments", National Center for Research on Evaluation, Standards, and Student Testing (CRESST), Center for the Study of Evaluation (CSE) Graduate School of Education & Information Studies, University of California, Los Angeles, CA, 1997.
- [2] Khattri, N., Reeve, A.L. and Kane, M., B., Principles and practices of performance assessment, Lawrence Erlbaum Associates, Inc., Mahwah, NJ, 1998.
- [3] Urban-Lurain, M. and Weinshank, D.J., "'I Do and I Understand:' Mastery model learning for a large non-major course", Special Interest Group on Computer Science Education, 30, No. (1999), 150-154.
- [4] Urban-Lurain, M. and Weinshank, D.J., Mastering computing technology: A new approach for non-computer science majors. AERA Annual Meeting, Vol. Division C, American Educational Research Association, Montreal, CA, 1999.
- [5] Urban-Lurain, M. and Weinshank, D.J., Attendance and outcomes in a large, collaborative learning, performance assessment course. <http://www.cse.msu.edu/rgroups/cse101/AERA2000/attendance.htm>, 2000.
- [6] Urban-Lurain, M. and Weinshank, D.J., "Do non-computer science students need to program?" Journal of Engineering Education, 90, No 4, (2001), 535-544.
- [7] Wiggins, G. and McTighe, J., Understanding by design, Association for Supervision and Curriculum Development, Alexandria, VA, 1998.