Methods of Interoperability: Moodle and WeBWorK

Michael E. Gage
University of Rochester
Rochester, NY, USA
gage@math.rochester.edu

ABSTRACT. The first requirement for an online mathematics homework engine is to encourage students to practice and reinforce their mathematics skills in ways that are as good or better than traditional paper homework. The use of the computer and the internet should not limit the kind or quality of the mathematics that we teach and, if possible, it should expand it. Now that much of the homework practice takes place online we have the potential of a new and much better window into how students learn mathematics, but only if we continue to ensure that students are studying the mathematics we want them to learn and not just mathematics that is easily gradable. Learning Management Systems handle mathematics questions poorly in general but when properly combined with specialized mathematics question engines, they can do much better and still retain their own look and feel for managing a course and collecting data. This paper presents an overview of two interoperation mechanisms developed to connect Moodle (an LMS) and WeBWorK (a mathematics question engine) for use in the WEPS (World Educational Portals) Open Online Courses. It provides a preview of the type of data that can be collected using these tools.

Keywords: Moodle, WEPS, WeBWorK, LMS, mathematics online homework

1 WHY INTEROPERABILITY IS NEEDED

Now that much of mathematics homework is mediated by online applications, we have the potential of a new and much better window into how students actually learn mathematics, but only if we continue to ensure that students are studying the mathematics we want them to learn and not just the mathematics that is easily gradable. Most universities mediate online activity through Learning Management Systems (LMS), also called Virtual Learning Environments (VLE), which are not good at providing feedback for mathematics intensive content. Specialized mathematics homework engines are better able to accomplish this task, but for student and instructor convenience, it helps if they can be melded seamlessly with the institutions' LMS to avoid the proliferation of different sites and interfaces.

1.1 Online Mathematics Homework Should Be Good

The first requirement for an online mathematics homework engine must be to encourage students to practice and reinforce their mathematics skills in ways that are as good or better than traditional paper homework. Online homework has some unique advantages; in particular, the immediate feedback provided by such homework keeps students on task and appears to have a positive effect on student

learning (Ellis, Hanson, Nuñez, & Rasmussen, 2015). The use of the computer and the internet, however, should not limit the kind or quality of the mathematics that we teach and if possible it should expand it. Mathematics education is compromised if certain portions of subjects are left out because they cannot be reinforced by presenting relevant, challenging questions using an existing online homework system. It is better in that case to abandon the online homework system despite any other advantages it might have.

Typical paper homework for mathematics has answers that might contain algebraic expressions, functions, complex numbers, matrices, or indefinite integrals (which are defined only up to an additive constant), and so forth. In most cases, there are many equivalent ways of expressing each of these answers and the computer should be able to match a human's ability to recognize all equivalent answers. For example, for the prompt "Give an example of a polynomial whose roots are 1 and −2," x^2+x-2, $(x-1)(x+2)$, and $5(x-1)(x+2)$ are all correct responses. Another example is that in a calculus class, $\cos^2(x) + \sin^2(x)$ is a legitimate representation of the constant function 1. Ideally a computer will correctly analyze all forms of these correct answers so that the students' focus is entirely on understanding and responding to the mathematics and they are not distracted by second guessing the limitations of the online system in recognizing correct answers (Bradford, Davenport, & Sangwin, 2009).

While mathematics instructors might grudgingly concede that a collection of well-designed multiple choice questions might be able to determine a student's mathematics knowledge and calculation skills with some accuracy (the SAT and GRE, for example, are used in practice) I don't think any of them would agree that they could learn these skills via multiple choice exercises. Our focus here is on learning mathematics (formative assessment — in other words: homework) and obtaining learning data on how this is done. Online mathematics engines can also be used for assessing knowledge (normative assessment) via quizzes or exams but that is a different topic.

1.2 Learning Management Systems (LMS) Need Help to Present Good Mathematics Homework

Because online homework use is now widespread, the data collected has the potential to provide improved insight into how students actually learn mathematics, but we must continue to ensure that students are studying the mathematics we want them to learn and not just mathematics that is easily gradable. Standard Learning Management Systems (LMS) do not by themselves do a good job of handling mathematics homework. One solution is to augment the LMS with plugins that are mathematically savvy. There are several popular and capable open source mathematics question engines, among them STACK, WeBWorK, MyOpenMath, ASSISTments (see Ostrow, Wang, & Heffernan, 2017), and lon-CAPA, which operate independently but have the potential to augment standard LMSs as well.

Providing ways to integrate these mathematics engines seamlessly into standard LMSs makes the LMS itself more effective as a learning tool and makes the question engine available to more students. If the

homework engine handles mathematics answers well, then the data collected has a much better chance of providing insight into how students are learning mathematics. Using only multiple choice questions for assessing student progress will probably not accurately provide insight into the learning process. To the greatest extent possible, learning analytic data should be collected using tools that best support mathematics learning and that minimally interfere with the learning process.

2   THE WEPS WEBWORK/MOODLE INTEROPERABILITY PROJECT

In 2014, Mika Seppälä and I undertook to revise and modernize the interoperability of WeBWorK and Moodle. This would allow the WEPS (World Educational Portals) project, founded by Seppälä, to better use the large collection of WeBWorK questions for its online education courses, in conjunction with existing STACK (System for Teaching and Assessment using a Computer algebra Kernel; Sangwin, 2004; 2013) questions. The mechanisms described here connect WeBWorK2 with Moodle2.x or Moodle3.x.

This article describes two ways that question engines and LMSs can interoperate, focusing specifically on the capabilities afforded by WeBWorK. The WEPS WeBWorK/Moodle project is part of a larger WeBWorK effort to make the WeBWorK question engine interoperate smoothly with Moodle, Canvas, Blackboard, and other LMSs using the Learning Tools Interoperability (LTI) interface as well as questions embedded in HTML pages using iframes.

In a rapidly changing educational ecosystem, it is difficult to keep up with the possible tool combinations available to educators and to those attempting to collect data on student learning. This article presents a few of the possibilities available through WeBWorK in hopes that it will provoke a wider discussion and spread the awareness of the type of learning analytic data that can be gathered using these systems.

In Gage (2017), we present detailed instructions explaining the installation of the WeBWorK Opaque server so that those currently using WeBWorK can recreate their homework sets as Moodle quizzes. This gives them access to the additional flexibility of the Moodle quiz structure. For those using Moodle quizzes, these instructions explain how to add mathematics questions from the WeBWorK OpenProblemLibrary to their current mathematics courses.

3   THE WEBWORK ONLINE HOMEWORK SYSTEM

WeBWorK began twenty years ago as a stand-alone application consisting of a minimal LMS and a powerful mathematical question engine. Since then, homework questions contributed and curated by many mathematicians to the OpenProblemLibrary (OPL)[1] have created a collection of over 30,000 Creative Commons licensed problems primarily directed toward calculus but ranging from basic algebra through matrix linear algebra. More than 1000 institutions are part of the co-operative open source

---

[1] https://github.com/openwebwork/webwork-open-problem-library

community that supports and uses WeBWorK. Students at more than 700 institutions actively used WeBWorK homework during the spring semester of 2016 (Figure 1). The large number of users[2] and the open source software repository[3] creates a mechanism for rapidly disseminating new questions, innovations, and data collection techniques to mathematics classes across the US and increasingly in other countries as well.

• WeBWorK is a server–client application. Students and instructors need only their browsers and an internet connection to use WeBWorK. The server side software runs on any Unix server.
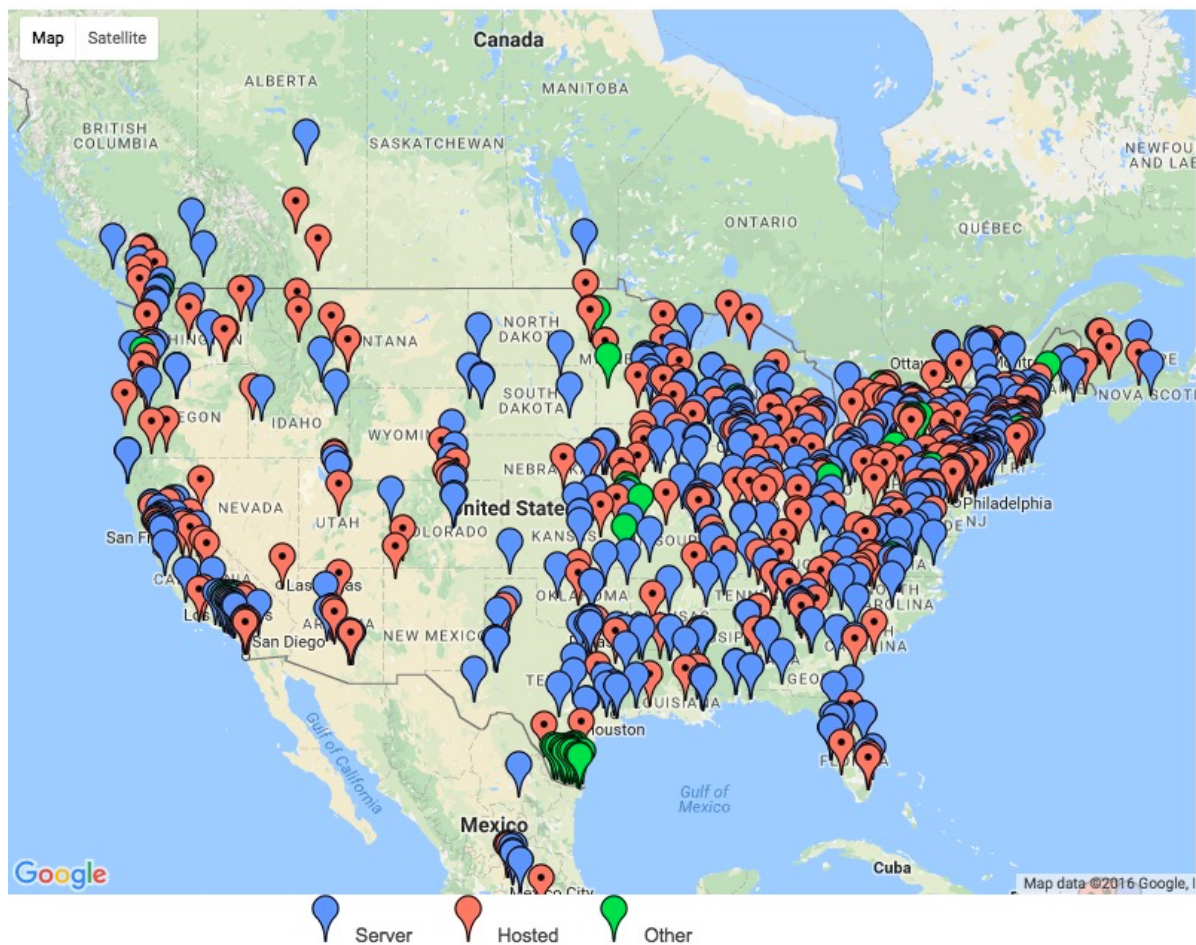

Figure 1: Map of institutions using WeBWorK[4].

• WeBWorK is partially distributed because most large institutions have installed their own servers. Some institutions, lacking the means or interest in having their own server, pay to use a central server hosted by the Mathematics Association of America (MAA).

• WeBWorK questions are algorithmic — each student receives a slightly different version of the problem. This allows students to practice together on the same homework questions without simply copying answers.

• The instant feedback given by WeBWorK as to whether the answer is right or wrong is a strong motivator to continue to work on the homework until every answer is correct.

• A small number of question authors have also written complete solution steps and hints. These are obviously more work and are not included in all of the questions in the library. It is possible to automate hint suggestions for specific wrong answers.

• Most instructors encourage students to use WeBWorK to check the correctness of their answer and to try to find and correct the mistake on their own if they can. If they are still having difficulty the student is encouraged to seek help from a human: a roommate, the science kid down the hall, a TA, or the instructor. The "Email the instructor/TA" feature is well regarded by students.

• Online homework engines are not yet at the point where they can grade abstract mathematical proofs of the type required in third- and fourth-year courses (real analysis, abstract algebra). This type of reasoning is seldom asked for in current first- and second-year mathematics courses. One can, however, write problems that ask for intermediate results as well as the final answer, thereby obtaining insight into the student's thought process. One can also write scaffolded problems — essentially worksheets — in which students cannot proceed to the next step until they have successfully completed the current one.

• Fortunately, it is not necessary to use WeBWorK for every homework question in a course. WeBWorK fits easily into hybridized courses where many different tools are used for homework practice and assessment. The time and energy saved by the automatic grading of computational problems can be used to pay closer attention to problems involving formal proofs.

• The extensibility and open source nature of WeBWorK and the high quality OpenProblemLibrary collection makes it a good research tool as well as an excellent learning platform.

• WeBWorK questions have been developed by college and university mathematicians for their own courses following the WeBWorK motto: "Ask the questions you should not just the ones you can." To meet this goal they have also expanded WeBWorK's answer checking capabilities. College and university mathematicians are also the curators of the OpenProblemLibrary (Holt & Jones, 2013). The result is a high quality and continuously growing collection of mathematics questions covering material from middle school to undergraduate.

4 WHY UNBUNDLING WEBWORK IS USEFUL

The WeBWorK software's greatest strength is its ability to ask and check answers for a wide variety of mathematics questions from pre-calculus through multivariable calculus, probability, and complex variables. We have not put the same effort into convenience tools for the instructor (which are not usually unique to mathematics). Our maximum effort has been placed on allowing the student to grapple with the mathematics directly without fussing with additional syntactic constraints imposed by the computer. If the answer is mathematically correct then the computer should accept it. Whether the derivative of $f(x)=x^5-2x^2+x+5$ is written as $5x^4-2x+1$ or $1-2x+5x^4$ or for that matter as $x(5x^3-2)+\cos^2(x)+\sin^2(x)$ should not affect the accuracy of the grading.

Rather than duplicate the effort of communities such as Moodle, which provide extensive instructor tools for managing quizzes, handouts, calendars, and communicating class announcements, the WeBWorK community has put most of its specialized expertise into mathematics question rendering and is putting this capability at the service of the general purpose LMSs to augment the quality of their mathematics presentation. This led to the early development of a WeBWorK web service, which over the last 10 years has become increasingly useful and supports two interoperability mechanisms, "assignment level" and "quiz level," for Moodle/WeBWorK interoperability.

5 THE BENEFITS OF INTEROPERABILITY AND PLUGINS

For mathematics instruction in particular, there is a big advantage to using plugins of either assignment level or question level type inside an existing LMS. The bulk of the services provided by an LMS involve class roster management, question management (due dates, etc.), organizing quizzes and adaptive learning assignments, clear communication with students (syllabi, calendars, bulletin boards, and so on), and grade books. None of this is specific to mathematics and it makes sense to have this developed by and for the widest group of educators.

The accurate checking of algebra, calculus, and higher mathematics answers on the other hand is needed for a much smaller number of courses, and development of these questions and answer checkers requires deeper knowledge of the material being presented. Having mathematicians and mathematics educators concentrate specifically on the mathematics engines makes sense in terms of efficiency as Hunt (2015) points out in the design documents for the Opaque protocol.

6 ASSIGNMENT LEVEL INTEROPERABILITY

Since 1999, a specialized Moodle plugin module "wwassignment" (originally "wwmoodle"[5]), has enabled assignment level interaction between Moodle and WeBWorK. The recent advent of the LTI (Learning

---

[5] https://github.com/openwebwork/wwassignment

Tools Interoperability) interface makes a standardized version of this mechanism available between any LMS and question engine that have implemented LTI. WeBWorK and MyOpenMath have LTI implementations as do most major LMSs such as Moodle, Canvas, Blackboard, and Desire2Learn.

An assignment level bridge presents a link in the LMS that takes students to a stand-alone math homework system such as WeBWorK, logs them in and generates a homework assignment for them automatically. (This is known as Single Sign On or SSO.) Once they are finished, the grade from the homework is returned to the LMS for inclusion into the grade book. (This is the grade reporting feature.) In this scenario, the homework engine maintains a record of each student's homework assignment and their progress. Instructors still use the homework engine infrastructure to create questions and assemble them into homework assignments and students still see two different interfaces, but SSO means that they need only a single login name and password to smoothly transfer from one application to the other.

The standardized LTI interface also provides single sign on (SSO) capabilities between the LMS and a specialized educational tool and the most recent LTI (1.2) provides a limited ability to pass the resulting grades back from the specialized tool to the LMS. This means that the assignment level interoperability originally provided between WeBWorK and Moodle by wwassignment is now available between WeBWorK (or MyOpenMath) and the newest versions of the major LMSs (Canvas, Blackboard, Moodle, D2L) that implement LTI1.2. While wwassignment allows for more customization between WeBWorK and Moodle the LTI interface connects WeBWorK and Moodle without requiring additional third party Moodle modules. For the moment, each has advantages but it's likely, as the LTI interface develops, that wwassignment will cease to be needed.

Thanks to the LTI interface (Goehle, 2016a; 2016b) the assignment level bridge between LMSs and mathematical question engines is poised to become mainstream and will significantly expand the choices and capabilities for delivering high-level mathematics intensive homework online.

7   QUESTION LEVEL INTEROPERABILITY

For quiz level interoperability, all student data and templates that determine the mathematics questions are maintained by the LMS quiz module. The quiz module handles all interactions with the student and passes the mathematics question template and the student's answer to the mathematics engine via a web service call for the rendering process. The rendering process uses the instructor designed question template to produce an individualized problem for each student. For example, one student might be asked to differentiate $3x^6\cos(5x)$ while another would be asked to differentiate $-7x^5\sin(2x)$. These mathematics formulas are written in the TeX language[6] and transformed into typeset mathematics

---

[6] See https://en.wikipedia.org/wiki/TeX

ready for display in a browser. When the student answers the question, the engine is consulted and returns an analysis of the student's answer including its correctness and possible syntax errors. The question and its answer evaluator are essentially small computer programs that when run, i.e., rendered, produce an individual question for each student and subsequently analyze the student's answer to determine whether it is correct.

In the pure case, the mathematics engine web service is completely stateless and all data is stored and manipulated in the LMS. This is true of STACK, where the Moodle quiz module stores the question template.

STACK and Moodle provide the most fully developed example of question level interoperability. STACK[7] was designed by Chris Sangwin and from the beginning it was designed to function solely as a "backend" mathematics engine while the "frontend" interface was to be provided by the Moodle quiz module. The Moodle quiz module has a long history but the current version Quiz2 of the quiz module was authored by Tim Hunt (2010). Hunt, Sangwin (2013, p. 103), and Pauna (2016) worked to ensure close interoperability between STACK and the Moodle quiz.

Hybrid question level interoperability is also possible. In this case, the question template library is maintained and updated at the question engine site but all student data and interaction is handled by the LMS. While reworking the Moodle quiz module, Hunt (2015) created the Opaque (Open Protocol for Accessing QUestion Engines) client to implement Moodle's end of this mechanism. The data required for the connection are minimal: only a URL for the engine, an ID for the question, and a URL for the base question bank are needed. Hunt has linked the Opaque client to the OpenMark question engine developed at Open University. The WeBWorK Opaque server created for this project uses the Opaque client as does the OUnit engine (Hoogma, 2016) that analyzes Java code submitted by students.

Using the Opaque question type, Moodle and the question engine communicate over the internet via a web service protocol. The interoperating applications do not need to be on the same computer or even in the same country. The question template data is maintained by the mathematics engine while the student data is maintained by the LMS. In principle, Blackboard or Canvas could implement an Opaque client module that would allow these hybrid mathematics engines to render questions in their quizzes as well. The current version of STACK is more tightly bound to Moodle so connecting it to Canvas or Blackboard would take more work.

8   COMPARING ASSIGNMENT AND QUESTION LEVEL INTEROPERABILITY

Each interoperability mechanism has advantages. The question level interoperability allows one to take advantage simultaneously of the highly customizable quizzes available in Moodle and the highly

---

[7] https://stack.maths.ed.ac.uk/demo/

developed mathematics capabilities of WeBWorK. (The standalone version of WeBWorK has a "Gateway Quiz" module but it is not as sophisticated as the one in Moodle.)

The advantages of the question level plugin include that all of Moodle's quiz data gathering capabilities can be used and that students see only one interface — the Moodle quiz — regardless of the question engine being used to evaluate their answers.

The assignment level interoperability (LTI) has the advantages that 1) it is more mature and more stable, 2) it is available between most LMSs and most question engines, unlike the question level interoperability, which requires more coordination, and 3) it allows additional collection of data by the question engine (see below).

9 THE MOODLE QUIZ

With the current implementation of the WeBWorK opaque server, it is now possible to include both WeBWorK questions from the OpenProblemLibrary and questions authored by STACK in the same online homework session, even within the same quiz. Students interact only with the Moodle interface, which is the same no matter which mathematics engine is being used. It is not yet perfect, but students have a uniform experience no matter which mathematics engine is checking their question. The only part of WeBWorK they see is the HTML presentation of the question and the analysis of the student answers. The question in Figure 2 is from the "Demonstrating STACK" course at the STACK Demonstration site[8].

Randomly generated versions

STACK generates random question versions for each attempt at the quiz. This is done in a sophisticated mathematical way, not by just putting random numbers into boxes which might lead to impossible questions.

Find the equation of the line tangent to $p(x) := 2 \cdot x^3 + 3 \cdot x^2 + 2 \cdot x$ at the point $x = 2$.

1. Differentiate $2 \cdot x^3 + 3 \cdot x^2 + 2 \cdot x$ with respect to x.

6x^2 - 6x+2

Your last answer was interpreted as follows:

$$6 \cdot x^2 - 6 \cdot x + 2$$

The variables found in your answer were: $[x]$

Incorrect answer.

Marks for this submission: 0.00/0.33. Accounting for previous tries, this gives **0.30/0.33**. This submission attracted a penalty of 0.00. Total penalties so far: 0.03.

Figure 2: A question analyzed by STACK.

---

[8] https://stack.maths.ed.ac.uk/demo/mod/quiz/attempt.php?attempt=1094&page=3

The question in Figure 3 is the quiz version of problem 3 in set "Demo" at the course 2014_07_ur_demo[9].

Figure 3: A question analyzed by WeBWorK.

Having both STACK and WeBWorK questions available in the same course, and even within a single quiz, gives instructors and mathematics education researchers a larger pool of questions to draw on. Having different engines (with different strengths) increases the ways one can analyze the student answers. The ability to place all of these questions in a common framework facilitates both teaching mathematics and the ability to do comparative research on the effectiveness of different questions and different question engines. The simple questions above were chosen for demonstration purposes. Both STACK and WeBWorK can present much more involved mathematical questions.

There remains ample opportunity for experimentation and tweaking of STACK, WeBWorK, and the Moodle Quiz behaviours to analyze the effectiveness of different configurations. The research on the most effective way to use online homework in mathematics courses is just beginning.

10   ADDITIONAL INTEROPERABILITY MECHANISMS

WeBWorK affords other mechanisms for interoperability.

Interoperability at the assignment level can be provided by LTI with many mainstream LMSs, as mentioned above. Detailed instructions for setting up the LTI communication between Canvas and

---

[9] https://hosted2.webwork.rochester.edu/webwork2/2014_07_UR_demo/Demo/3/

WeBWork have been written by Geoff Goehle (2016a; 2016b) and can be accessed through the WeBWorK blog aggregator[10] as well.

Detailed instructions for setting up the WeBWorK opaque server as described have been written by Gage 2017).

WeBWorK can be called upon to render an OpenProblemLibrary question within a web page in a manner very like the opaque question protocol. The question appears within an iframe on the page and is interactive. The demonstration web page (Gage, 2016a) contains several embedded calculus problems while Gage (2015) provides an explanation of the technique. The questions are all interactive and can be used for practice. Since no login is required, it is not possible to save the results to a particular student's record but aggregate data about how students perform on the question can be collected.

Using this same technique, interactive WeBWorK questions can be embedded in a MathBook XML document (Beezer, 2015). In 2016, Beezer and Jordan created a prototype example of a MathBook XML textbook with embedded WeBWorK problems.

11   WHAT LEARNING DATA CAN BE COLLECTED?

The learning data collected by Moodle quizzes is described in more detail in the companion piece in this issue by Matti Pauna (2017). Adding the WeBWorK connection allows the quizzes to access the contents of WeBWorKs 30,000 question OpenProblemLibrary but because it uses the Moodle quiz module, the data collected on each student's performance is unchanged.

The assignment interoperability behaves more like the independent version of WeBWorK — the WeBWorK server keeps a separate copy of a student's homework record and shares the summary results with the Moodle grade book. In this case, there are really two separate courses, a Moodle course and a WeBWorK course, linked by a shared single sign on and a shared grade book. The WeBWorK half of the course collects additional data: student progress toward completion, real time statistics on how many students have succeeded on their homework, and most interestingly a list of "past answers," the complete record of a student's submissions while attempting to answer a question. See Figure 4 for an example of a past answer data stream as a student tries to solve an algebraic equation.

Analyzing data of this sort by hand for around 100 students, Roth, Ivanchenko, and Record (2008) were able to reliably distinguish between productive progress and random guessing. They also found that guessing was extinguished for longer answers (or expressions) — students quickly realized that guessing was hopeless. Their analysis also revealed an unexpected pattern: strong students resubmitted wrong answers far more than other students.

---

[10] http://webwork.maa.org/planet/

09/19/2014 at 06:52pm EDT	$\left(\dfrac{3x+11}{5}\right)+\dfrac{3}{4}$
09/19/2014 at 06:53pm EDT	$\left(\dfrac{3+11x}{5}\right)+\dfrac{3}{4}$
09/19/2014 at 06:58pm EDT	$\left(\dfrac{3-11x}{5}\right)+\dfrac{3}{4}$
09/19/2014 at 07:01pm EDT	$\left(\dfrac{3-11x}{5}\right)$
09/19/2014 at 07:04pm EDT	$\left(\dfrac{3+11x}{5}\right)$
09/19/2014 at 07:13pm EDT	$\left(\dfrac{\sqrt{9+(x^2)}+11x}{5}\right)+\left(\dfrac{3}{4}\right)$

Figure 4: Example of a past answer data stream.

This study, done early in WeBWorK's existence, was used to improve the user interface. In the intervening years, computer power and data analyzing techniques have increased in power and WeBWorK's use is more widespread. At this time, we think that their analysis could be automated and applied to a much larger population of students using their Student Response Model categories (Roth et al., 2008) and the methods they used to train undergraduates to analyze the response data.

Currently past answers, student/class progress, and student/class statistics are available to a WeBWorK course instructor in a form convenient for real time decision making. Those hosting a WeBWorK site can mine the database of their site to aggregate this information, anonymize it, and analyze it. One of our goals is to create WeBWorK scripts that will perform this process at the click of a button on the site administration page, producing a spreadsheet sufficiently anonymized to be shared with an educational data site such as DataShop at Carnegie Mellon University[11].

12   CONCLUSION

This article presents researchers and educators with an overview of choices available for effectively presenting mathematics questions online with emphasis on opportunities provided by WeBWorK.

Gage (2017) describes in detail how to set up an opaque client within Moodle and how to connect to an existing WeBWorK Opaque server or to set one up alongside an existing WeBWorK installation. We hope this will make it easy for current WeBWorK users to move existing homework sets into Moodle quizzes and for those using Moodle quizzes to add WeBWorK questions from the OpenProblemLibrary. Goehle (2016a; 2016b) gives details on implementing LTI interoperability between WeBWorK and mainstream Learning Management Systems.

---

[11] https://pslcdatashop.web.cmu.edu/

Samples of the data that can be collected have been presented in the hope of encouraging additional researchers to use this rich collection of data to analyze student learning of mathematics. Analyzing the data collected from the extensive network of institutions using WeBWorK and Moodle has the potential to answer important questions about student learning. Such data should arise from the current best practice for using online homework in mathematical science courses and we believe that the combination of Moodle and WeBWorK makes this possible. "Ask the questions you should, not just the ones you can."

REFERENCES

Beezer, R. (2015, May 8). Open source mathematics with MathBook XML. [Presentation]. http://buzzard.ups.edu/talks/beezer-2015-manitoba/beezer-2015-manitoba-mbx.html

Beezer, R., & Jordan, A. (2016, June 13). Integrating WeBWorK into textbooks. [Demonstration]. http://spot.pcc.edu/~ajordan/ww-mbx/html/index.html

Bradford, R., Davenport, J., & Sangwin, C. (2009). A comparison of equality in computer algebra and correctness in mathematical pedagogy. *Lecture Notes in Computer Science*, *5625,* 75–89. http://dx.doi.org/10.1007/978-3-642-02614-0_11

Ellis, J., Hanson, K., Nuñez, G., & Rasmussen, C. (2015). Beyond plug and chug: An analysis of Calculus I homework. *International Journal of Research in Undergraduate Mathematics Education*, *1*(2), 268–287. http://dx.doi.org/10.1007/s40753-015-0012-z

Gage, M. (2015, June). Embedding single WeBWorK problems in HTML pages. [Blog post]. http://michaelgage.blogspot.com/2015/06/whether-writing-full-text-book-or-just.html

Gage, M. (2016a, January). Math 162 Calculus examples. [Demonstration]. https://hosted2.webwork.rochester.edu/gage/2016JMM/mth162_overview.html

Gage, M. (2017, June). Installing the WeBWorK opaque server. [Blog post]. Retrieved from http://michaelgage.blogspot.com/2017/05/p.html

Goehle, G. (2016a, March). WeBWorK LTI: Authentication. [Blog post]. http://webworkgoehle.blogspot.com/2016/03/webwork-lti-authentication.htmlGoehle, G. (2016b, March). WeBWorK LTI: Grading. [Blog post]. http://webworkgoehle.blogspot.com/2016/03/webwork-lti-grading.html

Holt, J., & Jones, J. (2013). WeBWorK OPL Workshop. [Blog post]. http://webwork-jj.blogspot.com/2013/07/webwork-opl-workshop-charlottesville-va.html

Hoogma, U. (2016). External question engine OUnit. [Blog comment]. https://moodle.org/mod/forum/discuss.php?d=331220

Hunt, T. (2010). Question engine 2. https://docs.moodle.org/dev/Question_Engine_2

Hunt, T. (2015). Open protocol for accessing question engines. [Moodle documentation]. https://docs.moodle.org/dev/Open_protocol_for_accessing_question_engines

Ostrow, K. S., Wang, Y., & Heffernan, N. T. (2017). How flexible is your data? A comparative analysis of scoring methodologies across learning platforms in the context of group differentiation. *Journal of Learning Analytics, 4*(2), 91–112. http://dx.doi.org/10.18608/jla.2017.42.9

Pauna, M. (2016, Spring). Calculus II [Online course]. https://geom.mathstat.helsinki.fi/moodle/enrol/index.php?id=312

Pauna, M. (2017). Calculus course assessment data. *Journal of Learning Analytics, 4*(2), 12–21. http://dx.doi.org/10.18608/jla.2017.42.3

Roth, V., Ivanchenko, V., & Record, N. (2008). Evaluating student response to WeBWorK, a web-based homework delivery and grading system. Computers & Education, 50(4), 1462–1482. http://dx.doi.org/10.1016/j.compedu.2007.01.005

Sangwin, C. (2004). Assessing mathematics automatically using computer algebra and the internet. Teaching Mathematics and its Applications, 23(1), 1–14. http://dx.doi.org/10.1093/teamat/23.1.1

Sangwin, C. (2013). Computer Aided Assessment of Mathematics. Oxford, UK: Oxford University Press.

WeBWorK URLs:

Wiki: http://webwork.maa.org/wiki

Sites: http://webwork.maa.org/wiki/WeBWorK_Sites

Blogs: http://webwork.maa.org/planet

Software repository: https://github.com/openwebwork

OpenProblemLibrary: https://github.com/openwebwork/webwork-open-problem-library