Development of a Computer Based Question and Tutoring System

Washington State University

Brandon Shibley, Chase Bradford, Brett Feero, J. Thomas Dickinson

Department of Physics, Washington State University, Pullman, WA 99163

Introduction

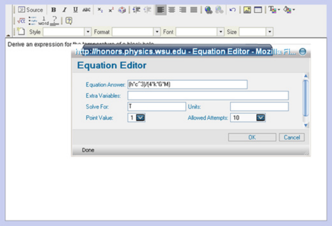
Professors who teach introductory courses at large universities often have classes with large numbers of students. This makes techniques that allow students to confront their understanding or lack of understanding of the material being studied very difficult. In class “student thinking” techniques such as quizzes (hand graded), ConcepTests (Eric Mazur- Harvard), cooperative learning, and Clickers are useful BUT they are limited to simple questions and brief answers – the clock goes by very rapidly when you are trying to cover significant quantities of material (Dr. Dickinson often uses the quote from “some” introductory physics student, “Taking your physics class is like trying to drink from a fire hose!”).

Current online homework systems could be used out of class for quizzing students but in general they do not challenge the student to be creative and they do not provide good feedback. Most services rely on asking questions that demand very specific responses, such as true-false questions or numeric answers. While these types of questions are useful at times, they are only requiring the student to think within very constrained bounds. If we can force students to go back to basics and be creative with their responses, and build a tool capable of interpreting many types of answers, students can learn to be more careful and complete thinkers and problem solvers. Our computer program facilitates making questions and answers, actually administering quizzes (and eventually tutorials) online, grading the quiz, and keeping track of student answers and scores. We briefly present our system as it exists currently.

Purpose

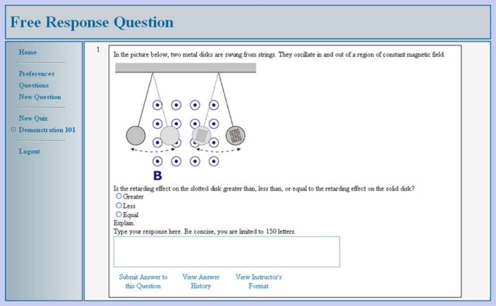
There are problems with typical written assignments because there is an insufficient supply of teaching assistants (TAs) available to grade them, leaving the professors with limited options with homework. A common choice is to assign many homework problems and only grade a smaller selection of them. However, the net result is the same students don’t receive feedback on all of their work.

Online homework systems are already in use, however, the current implementations are still delivering inadequate feedback to students and only ask questions which fail to challenge a student to think critically about the topics covered in lectures and textbooks. Thus, we developed modules for our system capable of handling single sentence responses and equations of varying difficulty. Finally, these systems are often commercially based and are very inflexible. Thus, features are very slow to develop and evolve.



Description of System

We have completed a high level, robust version of the system which allows instructors to create and manage their classes. Instructors have the capability to write problems which demand various forms of thought, from equation derivations to qualitative descriptions of important concepts. One of the key innovations of our grading system is its ability to grade questions which require a short sentence. Rather than feeding them possible answers (Multiple Choice), the students are required to think of possible answers on their own and use one which most accurately describes the phenomena in question.



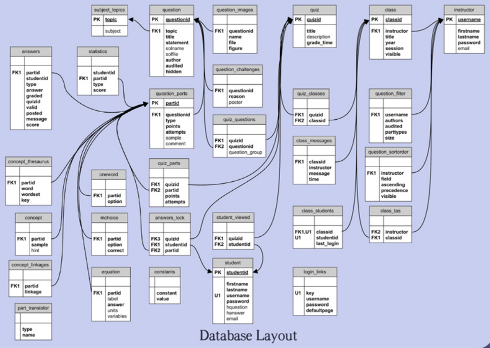
In addition, we’ve developed a system with enough features to make it a practical teaching tool. Throughout development, we designed interfaces that allow the instructor to accomplish what they want without worrying about too many details. As a result, our backend tracks an assortment of options and settings to anticipate the user’s actions. There are also safety checks which help to prevent accidental errors from entering the system, such as questions with flaws.

Structure and Components

The system has three primary components: database, website, and part management. The user interacts with the website, which runs on an Apache server, using PHP to generate the various web pages. The database is controlled by PostgreSQL and manages all user data, question information and course, assignment, and grade information. Part management is performed by C++ modules, for improved power, since some of the grading may require complicated computations (such as natural language responses).

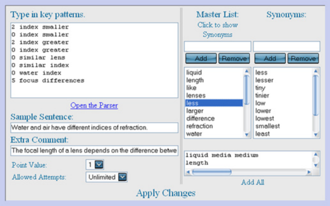
Database

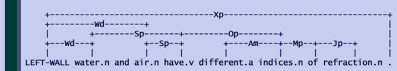
The database is managed by PostgreSQL, which is an easy to obtain open source database system. It stores all user information, as well as any questions in the library or courses and their assignments. As a result, it is necessary to keep regular backups of its contents to ensure that the data remains safe. Most importantly, all question information is stored inside the database, including solution files and images. Separate copies are also maintained on the file system to aide Apache’s retrieval of them, but the important copies are the ones stored as large objects. This way, backing up the question library can be performed when the database is backed up and does not require extra steps to store files on the file system.



Database Layout

Automated Sentence Grading





Example Parse

With our automated grading, we are moving toward a better understanding of sentence structure. By using the Link Grammar Parser developed by Carnegie Mellon University, we are able to analyze the structure of one-sentence responses. Finding relationships between words is done with a considerable amount of accuracy through CMU’s Link Grammar system. However the context of a response may introduce complex problems. Fortunately we may take advantage of question-specific grading criteria which ensures consistent context in a question by question basis.

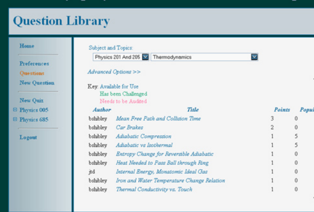
In order to grade a response, grading criteria must be entered, which consists of a list of linkages and a rudimentary, question-specific thesaurus. Our system takes this list of links, and enters it into a C++ container from the C code of the parser. This container allows certain queries entered by the question designer by means of a constraints list to be performed on the set of links.

Progress and Results

The infrastructure for our system is complete and robust. Generating questions and subsequent quizzes, performing the grading, and maintaining grades and student answers work very well. We currently have a database of questions and answers covering both semesters of introductory physics (a total of ~400 questions).

The formats that we are using include:

- Drill – standard one word, multiple choice, enter a number
- Short derivations
- Multi-equation problems
- Concept Questions (single sentence answers)
- Links, embedded movies and Java Applets with questions
- Links to web sites followed by questions
- Fermi Questions (semi-quantitative estimates)



The system has been tested as it has evolved for 4 semesters in Honors Physics 205-206. Each semester we make improvements and additions in all parts of the system and question database. Quizzes were due 2-3 times a week right before class time. The student response is mixed. Asked to rank their opinion at the end of the semester of the online quizzes, student responses on average ranked the experience as follows:

Excellent	Good	Average	Poor
	X		

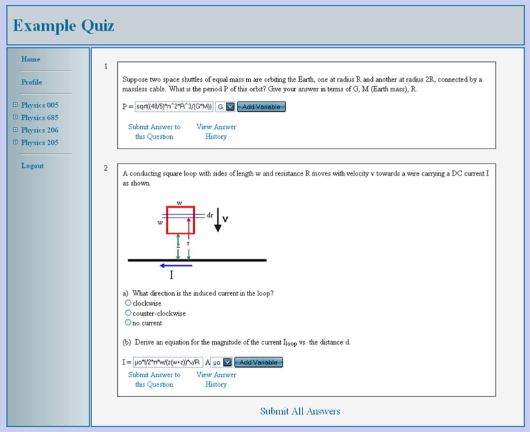
Comments on the on-line Q&T experience were usually focused on the pressures produced by the high frequency of taking these quizzes. Dr. Dickinson felt that overall it performed the function he desired, namely forcing students to grapple with their understanding of concepts and problem solving techniques. Students admitted that they were forced to read the book much more carefully and go over the examples in detail. Dr. Dickinson felt that this was a major goal of his teaching technique (“Read The Book!”). One repeated positive remark was appreciation of being able to take the quizzes over a lengthy period of time (~48 hours) at any time, night or day.

Conclusion

We have generated and tested a computer based Q&T system that will soon be released as open software to appropriate instructors. The latter will be able to generate a class or classes, make up new questions and add them to the database, use the question database to generate quizzes and tutorial sessions, automatically notify students that the quiz is ready with a particular due date/time, computer grade the student’s answers, and keep records on all student answers as well as quiz scores (like a grade book). Although under further development and improvement, the existing system is robust against crashes, easy to use, and functions under test. Students do not love using the system, primarily because of the intellectual challenge and “forced learning” it engenders. At the same time, they do not dislike it, rating it, on the average, “Good”. As a tool to get students to focus on concepts and problem solving, we consider it a success. Although developed for physics classes, with minor modification it could be used in mathematics, chemistry, and a number of other quantitative and equation oriented courses.

Examples

An example quiz appears below along with the solutions as seen by the student after the due date.



Solution:

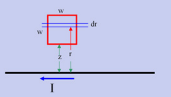
$F_{g_1} + F_{g_2} = F_{c_1} + F_{c_2}$	Begin with force equality. The force of gravity is equal to the centripetal force.
$\frac{GMm}{r_1^2} + \frac{GMm}{r_2^2} = \frac{mv_1^2}{r_1} + \frac{mv_2^2}{r_2}$	Create expression.
$\frac{GM}{R^2} + \frac{GM}{4R^2} = \frac{\left(\frac{2\pi R}{T}\right)^2}{R} + \frac{\left(\frac{2\pi(2R)}{T}\right)^2}{2R}$	Set $v = \frac{2\pi r}{T}$, cancel ‘m’s and set $r_1 = R$ and $r_2 = 2R$
$\frac{5GM}{4R^2} = \frac{4\pi^2 R}{T^2} + \frac{8\pi^2 R}{T^2} = \frac{12\pi^2 R}{T^2}$	Reduce.
$T = \sqrt{\frac{48\pi^2 R^3}{5GM}}$	Solve for T.

Solution:

(a) The current in the loop is counter-clockwise.

Reasoning: The flux due to I through the loop is into the paper. As the loop moves towards the wire the B field is increasing as 1/distance. To oppose this change in flux the B field generated by the induced field should be out of the paper; therefore, the induced current in the loop is counter-clockwise (right hand rule).

(b) We first find the flux Φ through the coil at some d, then find the induced emf $\varepsilon = -d\Phi/dt$ then we find the current $I = \varepsilon/R$. Take the rectangle and introduce a strip of width dz at z as shown:



We know that B from the wire is:

$$B(r) = \frac{\mu_o I}{2\pi r}$$

$$\Phi = \int_S \mathbf{B}\cdot d\mathbf{A} = \int_z^{z+w} B w dr = \int_z^{z+w} \frac{\mu_o I}{2\pi r} w dr = w\frac{\mu_o I}{2\pi}\int_z^{z+w}\frac{dr}{r}$$

$$\Phi = w\frac{\mu_o I}{2\pi}\frac{1}{w}\ln\left[\frac{z+w}{z}\right]$$

Therefore the induced emf will be given by:

$$\varepsilon = -\frac{d\Phi}{dt} = \frac{\mu_o I}{2\pi}\left(\frac{-w}{z(w+z)}\right)v, \quad \text{where we substitute: } v = \frac{dz}{dt}$$

Therefore the induced current, $I = \varepsilon/R$, where we drop the - sign:

$$I = \frac{\mu_o I}{2\pi}\left(\frac{w}{z(w+z)}\right)\frac{v}{R}$$

Note that as w or v → 0, I goes to 0 as expected. If z → 0, I → infinity.

Future Work

We are continuing to add support for various problem types. The architecture allows us to create specialized components that expand the problem types without interfering with the existing framework. Some of the types we are including:

- Chemical equations, and specialized formats such as Lewis Structures (suggested by Professor Hipps in Chemistry)
- Complicated equations with Java
- Other applet driven problems including graphing of results

We are developing a question sharing tool which allows instances of this system controlled by different institutions to share their questions more easily. This will greatly improve our ability to distribute question packages, keeping libraries full of creative questions to use in assignments.

Along the lines of sharing, various improvements to the code base are being made. One of them is changing the underlying layout of our generated web pages. The system being tested in Dr. Dickinson’s Honors Physics classes render all pages inside a single frame, which makes different elements sensitive to change. This is an issue for people hoping to further develop its capabilities; to avoid potential frustration we are moving over to a multiframe window format. We are also establishing a source code location (on our server), for patches and improvements to be sent for incorporation into future versions.

Finally, the install system is being automated. We are replacing a short list of steps which must be taken by a website administrator for this system to be fully installed and organizing them into a series of install scripts which will expedite the install process, after a few configuration rules are set by the administrator.

Acknowledgements

This work was supported by grants from The Camille and Henry Dreyfus Foundation, the Northwest Academic Computing Consortium, and from a WSU Teaching and Learning Improvement Grant. The guidance of our advisor, Tom Dickinson, Professor of Physics and Astronomy is gratefully acknowledged.