Dissemination of Animation-Based Curricular Materials: BQ-OSP Database

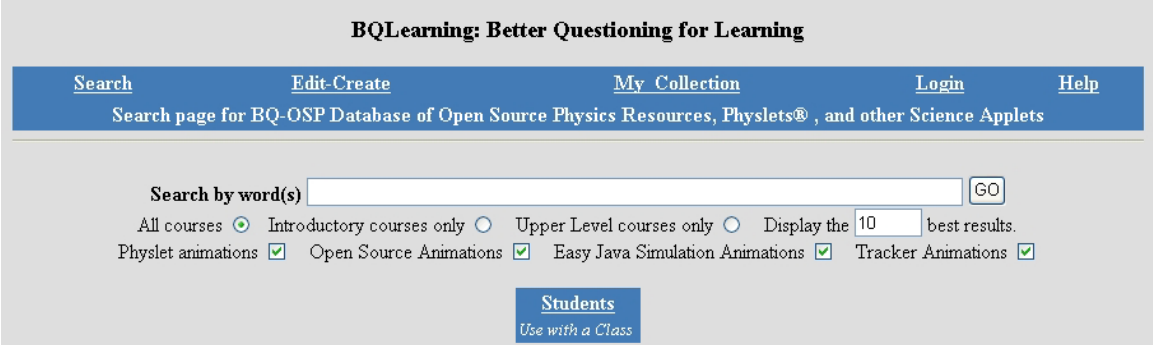
Anne J Cox and William F. Junkin III, Eckerd College, St Petersburg, Florida, USA

Abstract:

This poster describes the design and development of a database and an associated website for disseminating computational-based animations and the accompanying curricular materials (http://www.BQLearning.org). The audience should range from users to contributors, i.e., from students or teachers using materials 'as is' to people who delight in computer programming. The primary design considerations are ease of use, of searching, of organization, of editing or customizing prior submissions, and of submitting new resources. We also worked closely with the ComPADRE digital library so that a search of ComPADRE could include a seamless search of the BQ Database. This poster will describe the design of the website and its database used to reach these goals.

Open Source Physics is supported in part by the National Science Foundation (DUE-0442581)

Background and Objectives: Focus on the Users



There are many different groups with many different objectives and levels of technical ability -- students, instructors, curriculum authors, and programmers (and these are not mutually exclusive).

The basic goal of the BQ-OSP Database is to provide a user-friendly place where people can exchange ideas within their group and between groups. Our aim, therefore, was to create a site where all users could find and submit resources which are

1. Easy to find
2. Easy to use
3. Easy to contribute

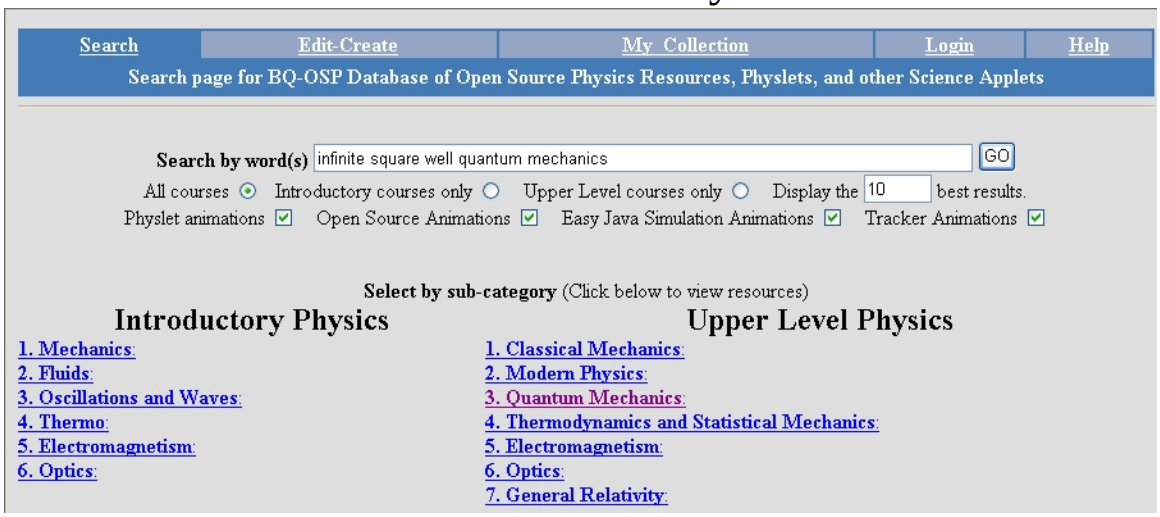
Additionally,

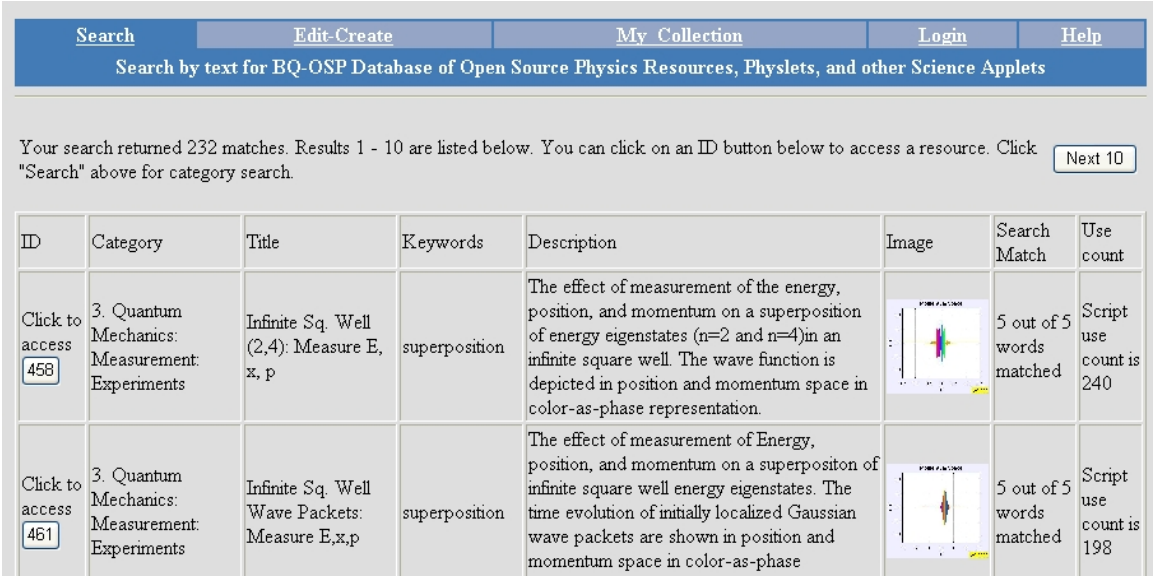
- These requirements must be true for physlet, OSP, and EJS animations as well as Tracker resources.
- Each resource is a simulation **with** curricular material.

Easy to Find:

www.BQLearning.org

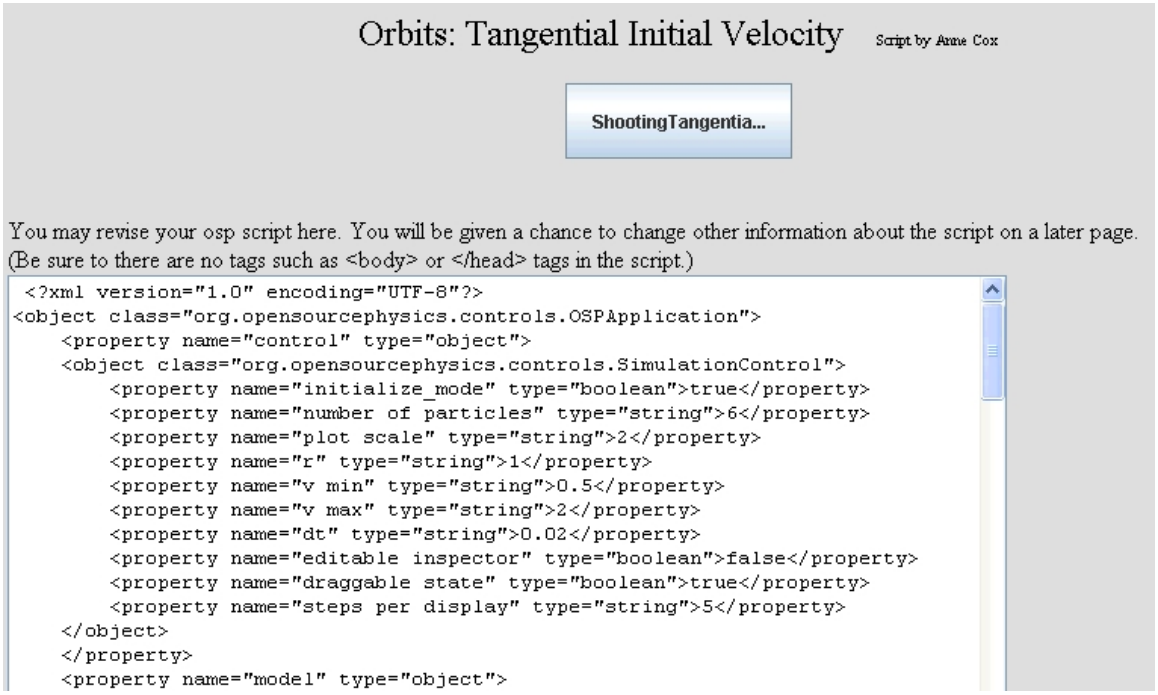
- Searchable
- Integrated with ComPADRE (keywords and necessary metadata)
- Organization consistent with traditional physics texts
- Thumbnails of simulations
- Search returns rank ordered by use statistics





Easy to Use:

- Teachers: Each resource has a URL which they can put on their own page or site hosted within database
- Students: Go to Teacher's page or Database and click on link to view curricular materials and run simulation. (Note: Database page does not support on-line answer submission or automatic grading)
- Programmers, developers and curriculum authors: Can see what others are doing and build on the work of others (see scripts of Physlets, xml files for EJS, Tracker and OSP resources)



Easy to Contribute

Level of “geekiness” (login required by self-registration)

Level 000: Submit new curricular material or modify existing curricular material (to go with existing simulation) via typing text in a webform.

Level 001: Modify existing simulation (javascript for Physlet, xml for OSP)

Level 010: Download simulation (EJS, Tracker), modify files locally and upload modified resources.

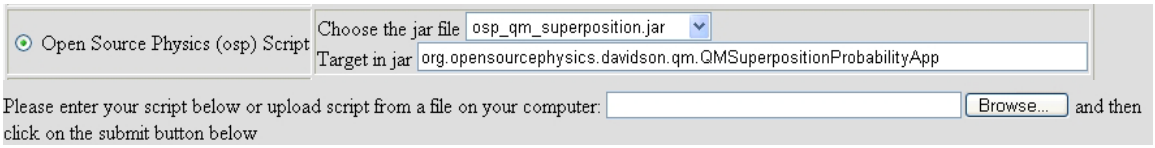
Level 011: Submit new simulation (javascript or xml that uses jars stored in Database)

Level 100: Install Database and programs on local Windows machine (automatically installs Apache, MySQL and php on any Windows PC)

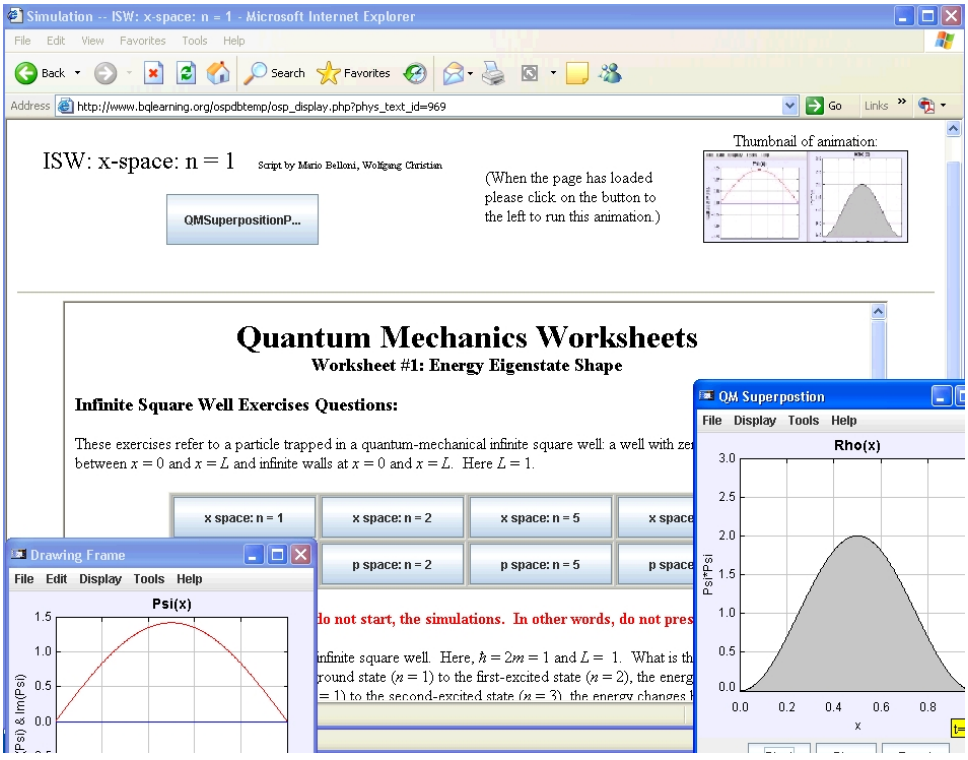
Level 101: Develop jars and send for inclusion in Database (currently via email only to coxaj@eckerd.edu or junkinwf@eckerd.edu)

OSP Resources

- Upload xml files: Asks user to choose jar file name and supply target class as well as xml file



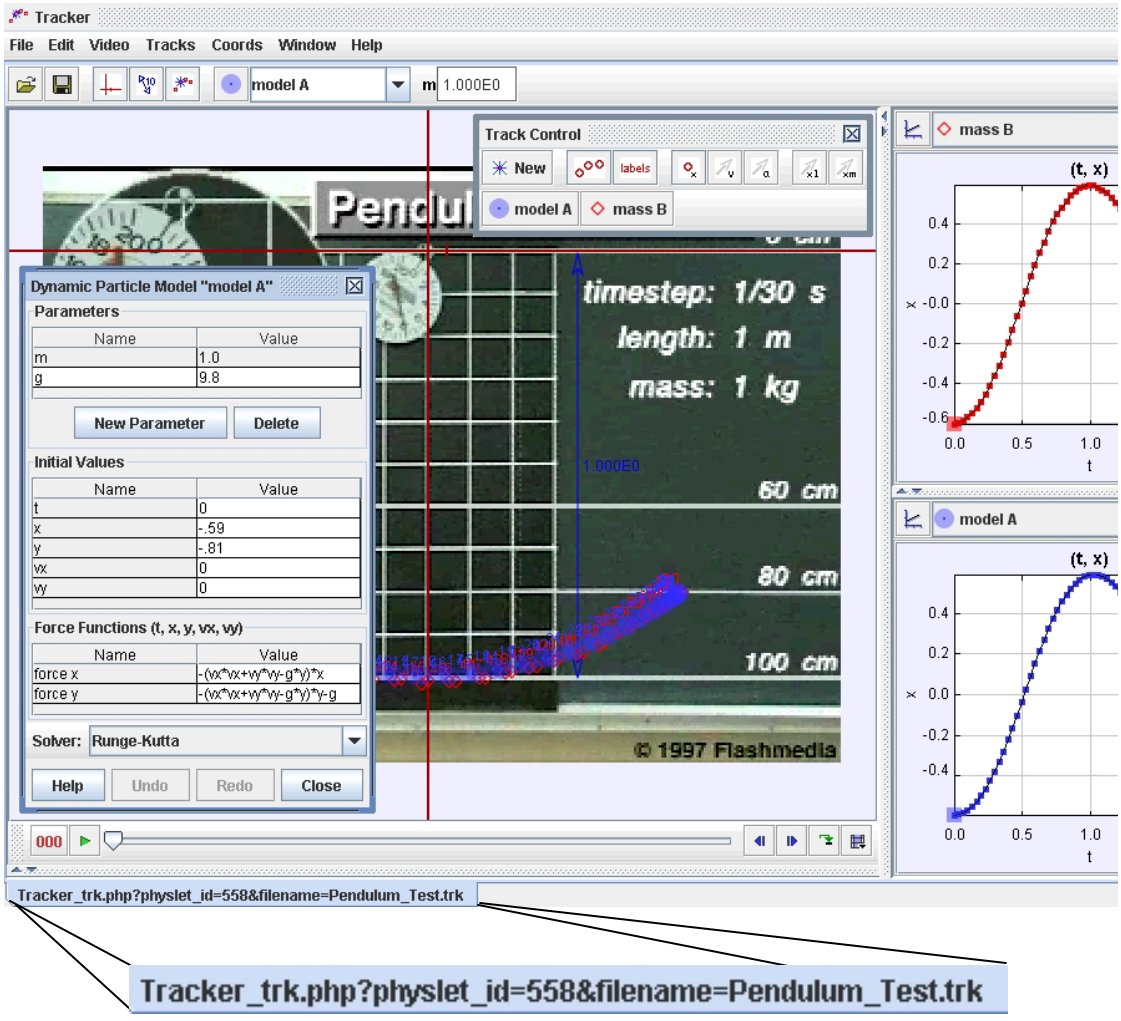
- Delivery via Application Applet and iFrame containing html page (this allows for a single curricular resource using multiple simulations)



Tracker (Proof of Concept):

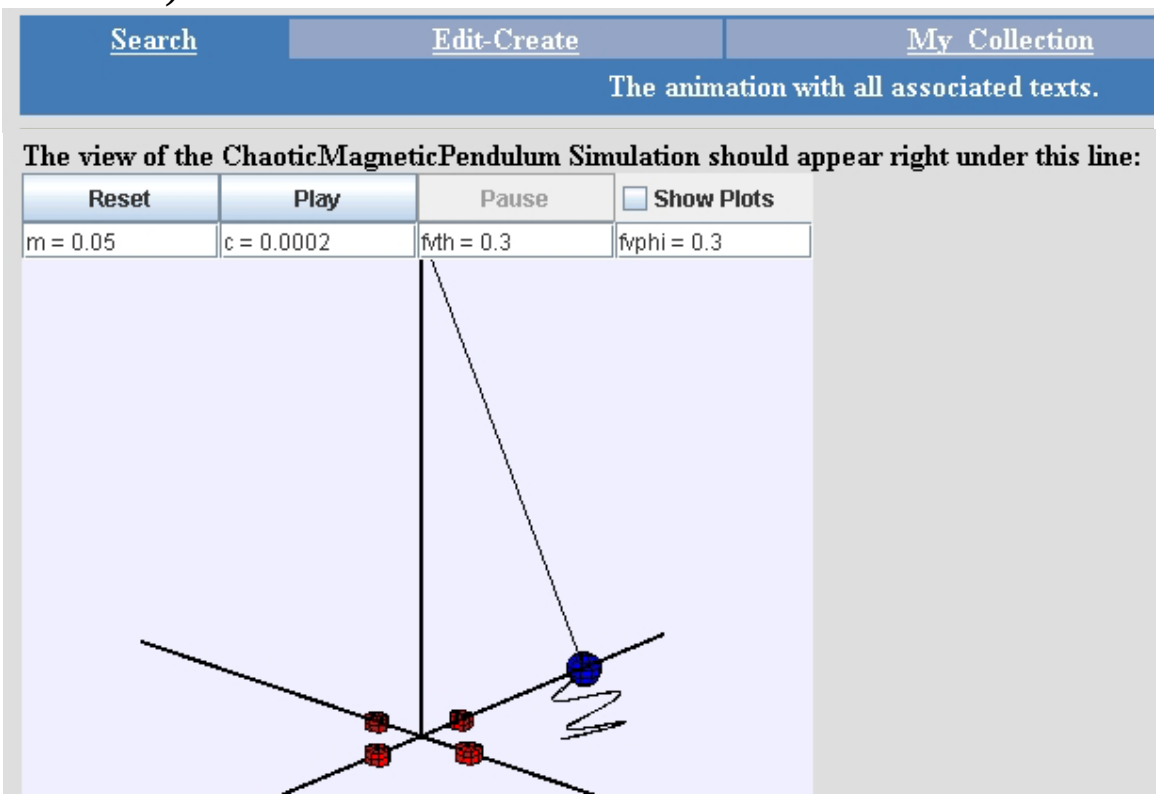
Tracker: Video analysis of movies

- Uses a php file to launch Java WebStart for Tracker and the requested xml file.
- Curricular materials delivered in iFrame (to allow for pdfs, docs, etc).
- Movies must be stored on other servers, not within the Database (generally need to be less than ~ 4 MB or connection will time-out) so user must specify the movie address with an absolute URL reference.



EJS (Proof of Concept):

- User submits xml file from local computer
  Note: File must be in a subdirectory of
  /_users/<organization>/<username>
  (for example: /_users/murcia/fem/Demo/EMWave.xml)
- Program parses xml for auxiliary files and requests the submission of these files
- EJS creates a small jar for that application which can be run locally or from the Database
- The Database program generates a zip file with this jar, the xml, and all auxiliary and html files. A link is provided to this zipped file, allowing for a download of this resource to local computer and use the full power of EJS (this assumes the user has downloaded EJS)



Database Server Security

- Users self-register (people set own password and are certified users as soon as they complete registration)
- Any certified user can submit files
- To protect Database server from virus submission, all submitted resources are raw URL encoded (php function) and placed in fields in database tables
- Any xml files, images, html files and other resources that are used in a simulation are obtained from the Database by a php file, raw URL-decoded and sent with the proper header

Exception:
Zipped EJS resources: The necessary resource files are put on server in proper directories, a php program is called which zips them up and creates the desired zip file that is stored as a flat file on Database server. Resource files are then deleted and the relative address of zipped file is stored in Database. The associated jar (created by EJS) is also a flat file on the server using the directory structure of _user/<organization>/username/ to avoid one author overwriting a resource of the same name created by another author.

Technical Summary:

- Database is MySQL database and program is written in php by Bill Junkin
- Some BQDatabase resources will soon require Java 1.6
- Tested with IE, Firefox and Safari (requires that Javascript be enabled but uses no pop-ups)

Future:

- Move EJS and Tracker from proof-of-concept
- Address potential overwrite of EJS files (user directory structure)

Acknowledgements:

- Support from NSF Grant DUE-0442581
- Francisco Esquembre, Doug Brown, Wolfgang Christian and Mario Belloni