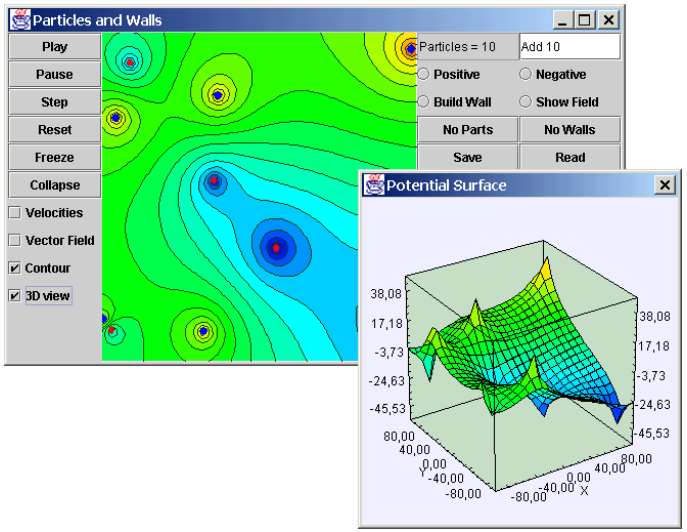
Modeling Science with Open Source Physics and Easy Java Simulations



Francisco Esquembre
Universidad de Murcia, Spain
Wolfgang Christian
Davidson College, USA

Includes: ***Tracker*** *by Doug Brown and* ***BQ*** *Database by William Junkin*

Partial funding for OSP was obtained through NSF grant DUE-0442581.

The talk at a glance

- Courses in simulation and computation have become widespread in physics and are increasing in the other sciences. This is because:
  - Computation offers a third way of doing science.
  - Of the importance of being *computer-literate*. (di Sessa – *Changing minds*)
- An area of particular interest in this respect is that of computer modeling.
  - Learning science is learning to develop, test, and apply models to explain or predict natural phenomena.
  - Computer modeling can improve physics teaching and learning.
- But computer modeling poses a technical barrier
  - Many (teachers and students) shy away from computer modeling with traditional programming tools.
- The **Open Source Physics Project** addresses the needs of programmers, teachers, and students by providing a computational physics library, ready-to-use models, and high level tools such as Easy Java Simulations.

Pedagogic Advantages of Modeling

- The premise of modeling is that when students are not actively involved in the modeling process they lose out on much of what can be learned from computer simulations.
- We make a distinction between modeling and programming. Modeling is closer to laboratory experiments and is more likely to provide the student with a novel perspective on the behavior of a system.
- Modeling has been shown to correct weaknesses of traditional instruction by engaging students in the design of physical models to describe, explain, and predict phenomena.
- Although the modeling method can be used without computers, the use of computers allows students to study problems that are difficult and time consuming, to visualize their results, and to communicate their results with others.
- The combination of computer modeling with theory and experiment can achieve insight and understanding that cannot be achieved with only one approach.

Modeling Cycle

- The goal of modeling is to teach in a student-centered environment where students do not solve problems in a formula-centered way.
- Modeling Instruction attempts to enhance student achievement through a process called the **Modeling Cycle**, (following Robert Karplus' Learning Cycle).
- Throughout the Modeling Cycle we rely on student engagement and explanation as the dynamic of learning.
- There are two major parts to the Modeling Cycle, model development and model deployment.
  - ***Model Development***
    - *A. Qualitative description.*
    - *B. Identification of variables.*
    - *C. Planning for the experiment.*
    - *D. Laboratory experiment.*
    - *E. Analysis of experiment.*
    - *F. Presentation of experimental results.*
    - *G. Generalization*
  - ***Model Deployment***
    - *A. Extrapolation and reinforcement.*
    - *B. Refinement and integration.*

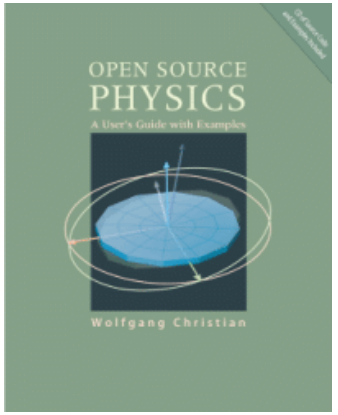
But how to create computer models?

- Creating computer models requires programming.
- The modeling cycle additionally requires advanced visualization and interaction.
- Deployment requires extra techniques (packaging, web access, digital libraries,…).

- These technicalities pose an important technical barrier.
- Tools are required that make the modeling cycle easier.

Part I: Open Source Physics (OSP)

- The OSP project provides an object-oriented, Java-based, open source solution to common teaching and modeling tasks in Physics (but more generally in Science and Engineering).
- It is a combination of
  - ready-to-use curricular material
  - tools that help you adapt and adopt these materials to your actual teaching
  - repositories (libraries) to help you find the right material
- OSP seeks to provide:
  - A set of different software frameworks
  - A Java implementation of them
  - A collection of ready-to-use applications
  - High-level tools for non-programmers
  - International collaboration through national digital libraries



- Contributors
  - **W. Christian, M. Belloni, A. Cox, W. Junkin, H. Gould, D. Brown, J. Tobochnik**
  - **F. Esquembre, Fu-Kwun Hwang, Sebastian Dormido, Jose Sanchez, J. M. Aguirregabiria, S. Tuleja**

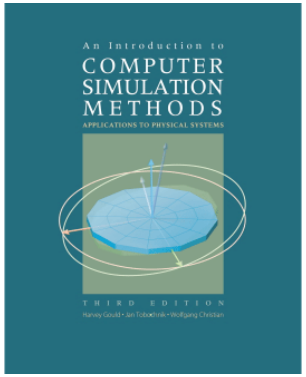
Characteristics of OSP programs

- Can be run (requires Java):
  - As independent applications
  - As applets in HTML pages
  - In a single executable package that bundles many applications together
- Use the Model-View-Controller paradigm. Hence we have frameworks for:
  - Numeric computation (Model)
  - Visualization (View)
  - Graphical user interfaces and thread-handling (Controller)
- Can incorporate standard facilities such as printing, cut & paste, XML input and output, screen shots, data fitting, etc.
- Are written in Java and are distributed under the GNU GPL license. Hence, they are open source.

How to use the OSP material

Consider three different users:

- Regular Teacher
  - Uses ready-to-run curricular material.
  - Creates additional tasks, worksheets, exercises for these materials.
- Modeler (not a programmer)
  - Uses Tracker for video analysis WITH simple modeling.
  - Uses Easy Java Simulations to create new simulations or adapt existing ones.
- Programmer (teacher or student)
  - Has access to the Java code. Uses the object-oriented OSP library and software frameworks.



Ready to use curricular material

Here go some Java demos!

From individual applications:

- Pendulum (Ejs)
- Bouncing Ball (Ejs)
- Radiating Field
- N-body Trajectories (Ejs)
- QM Superposition
- Spinning Top (Ejs)

To curriculum packages:

- Quantum Mechanics
- Statistical and Thermal Physics
- General Relativity

Which can be adapted to produce new packages:

- Launcher and Launch Builder
- Everyone can be a curriculum developer (author) with OSP tools.

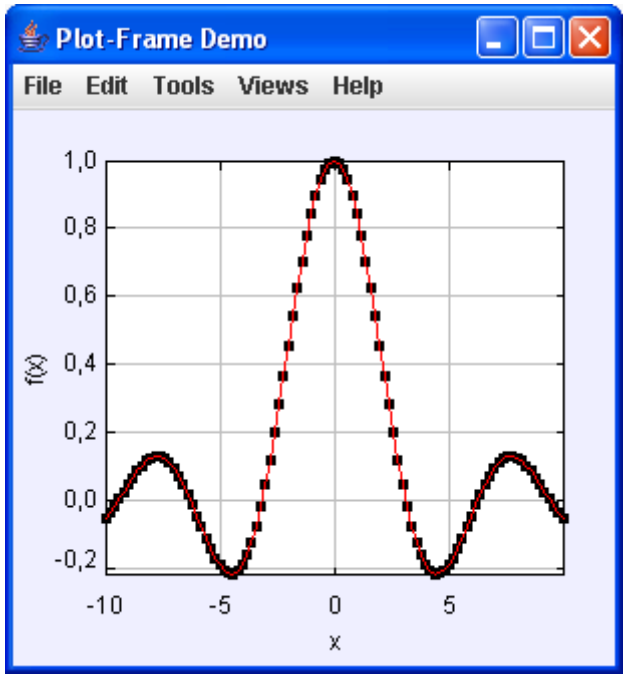
High-level modeling and authoring tools

- Tracker video analysis with simple modeling
  - Benefits of Video Analysis
  - Overlay dynamic models

- Easy Java Simulations (Ejs)
  - A high-level modeling and authoring tool.

OSP libraries

- Frameworks
  - Java tools and libraries for the beginning (and advanced!) programmer
    - OSP Frames – high-level objects for routine visualization tasks




```
import org.opensourcephysics.frames.PlotFrame;

public class PlotFrameApp {
  public static void main(String[] args) {
    PlotFrame frame = new PlotFrame("x", "f(x)", "Plot-Frame
Demo");
    frame.setConnected(true); // connects dataset points
    double dx = 0.2;
    for(double x = -10;x<=10;x += dx) {
      frame.append(0, x, Math.sin(x)/x);
    }
    frame.setVisible(true);
    frame.setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
  }
}
```

Part I Summary:

OSP material can be used by anyone.

- Regular Teacher
  - Use ready-to-use curricular material.
  - Create additional tasks, worksheets, exercises for these materials.
- Modeler (not a full programmer)
  - Use Tracker for video analysis.
  - Use Easy Java Simulations to create new simulations or adapt existing ones.
- Programmer (teacher or advanced student)
  - Has access to the Java code. Uses the object-oriented OSP library and software frameworks.

Part II: Distribution and Collaboration

Dissemination is a serious business

- Computer materials need to be
  - Easy to find
  - Reasonably priced (i.e. free)
  - Easy to install and use

  … but, above all, …

  - **Good educational software needs to be adoptable and adaptable.**

    Edward Redish in the Foreword to “Physlets”:

Distribution and Collaboration with BQ

How to find what we need
(do we find what we need?)

Searching for a Pendulum simulation is easy:

- Google:
  - *Pendulum simulation Java:* 219,000 entries
  - *Pendulum simulation*: 758,000 entries!

- Digital Libraries:
  - Merlot: *Pendulum* in the *Physics* category: 37 results
  - ComPadre: site search for *Pendulum*: 26

Distribution and Collaboration with BQ

How to find what we need? (2)

- In most cases all we get is a nice applet running.
- In very few cases, we get an explanation of the Physics and how the simulation was built.
- In one case (Fu-Kwun Hwang's server), we can register and get a copy of the applet for local use.
- We don't find programs we can (easily):
  - Inspect (to understand or check for correctness) - Adopt
  - Modify (change or improve) - Adapt

Distribution and Collaboration with BQ

How to find what we need? (3)

- The ideal solution would be to get a simulation that:
  - Is ready to run (for a first impression)
  - Requires no non-standard, heavy-weight software
  - Can be inspected in an accessible way
  - Can be downloaded and used locally
  - Can be modified (and re-run)
  - Can be localized (translated to your language)
  - Has been peer-reviewed (Quality Control)
  - Has been used by other teachers
  - Is absolutely free
  - (Add your own requirement…)

Distribution and Collaboration

- The BQ Learning Database
  - The BQ database is used to store OSP curricular materials in a searchable database at www.bqlearning.org
  - The database is designed to give instructors (and students) easy access to curricular resources ready for them to use right away.
  - Registered users can modify these resources to be used locally and/or by the rest of the educational community.
  - The resources are arranged by category (i.e. Mechanics, Modern Physics, etc.) and can also be found by entering search words.
  - It is linked to the USA National Science Digital Library project through the searches provided by ComPADRE.

Distribution and Collaboration with BQ

Time for a Demo! In this demo we shall:

- Work with a simulation
  1. Search for a simulation of a pendulum
  2. Run and inspect it
  3. Modify it
  4. Create a self-executable, distributable version
  5. Pack with other simulations for a complete curricular unit

- Work with a video analysis
  1. Search for a video analysis of a collision
  2. Run Tracker to modify it
  3. Save it as a local resource

Advantages of BQ

- A Java-enabled browser is all it needs. Software downloads and installs automatically as needed.
- No need to register for read-only access.
- Uses signed applets/applications (to write in your hard-disk).
- Allows to browse, search, download, inspect, and run a collection of simulations.
- Registered users can contribute curricular material.
- Registered users can upload new simulations.
- BQ database entries appear within a ComPADRE search.

BQ Authoring

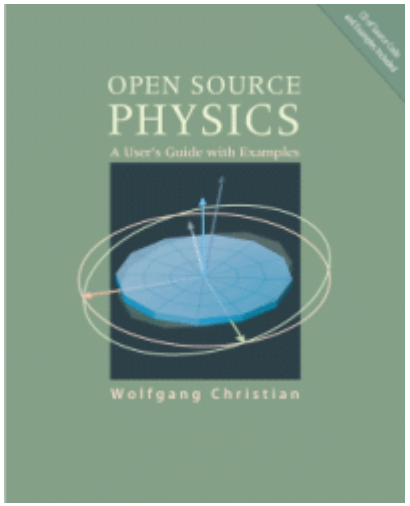
- The BQ database uses on-line (applet or Web Start) versions of high-level OSP tools (Tracker, Easy Java Simulations)
- The browser communicates with a PHP+MySQL database server which serves simulations
- High-level authoring/modeling tools are used and hosted locally independent of the server to:
  - Adapt existing simulations
  - Run simulations
  - Create easy to distribute, complete curricular package.
- Only XML model descriptions are uploaded to the server.

Part II Summary:

Distribution and Collaboration with BQ

- Teachers can browse OSP models and curricular material and immediately distribute this material to students.
- Contributions are welcomed. Users (at different levels) can register and contribute:
  1. Curricular material (exercises, experiments,…) for existing scripts (animations, simulations, or video analysis)
  2. New scripts and new XML-based models.
- www.bqlearning.org

The Open Source Physics project



Visit us at:

www.opensourcephysics.org
www.um.es/fem/Ejs/
www.cabrillo.edu/~dbrown/tracker/
www.bqlearning.org

Partial funding for OSP was obtained through NSF grant DUE-0442581.