USING PHYSLETS AND EASY JAVA SIMULATIONS TO TEACH PHYSICS AND ASTRONOMY

Mario Belloni
Davidson College

Wolfgang Christian, Doug Brown, Anne Cox, Francisco Esquembre, Harvey Gould, Todd Timberlake, and Jan Tobochnick

This work is supported by the National Science Foundation

(DUE-0442581, DUE-0126439, and DUE-9752365)

Overview: Open Source Physics Resources

Open Source Physics (OSP) **Project** provides curriculum resources and tools that engage students in astronomy & physics, computer modeling, and computation with the goal of providing students with new ways to understand, describe, explain, and predict physical phenomena.

- The **OSP Collection** is a ComPADRE repository where >500 EJS (Java & JavaScript) models and curricular materials are organized & shared.
- **Physlets** are small interactive Java applets that are designed for the teaching physics in a web environment. **Physlet Physics 2E** is an integrated curriculum of over 800 items spanning the introductory physics sequence. **Physlet Quantum Physics 2E** contains over 200 items covering modern physics through advanced quantum theory.
- **Easy Java Simulations** **(EJS)** encourages modeling and authoring with basic programming in Java*. EJS removes many of the complicated tasks involved in integrating computation into the classroom allowing students and teachers to focus on the science.
- **Tracker** video analysis and modeling tool analyzes video clips. Students can both analyze the motion of objects and overlay simple models on the video & see how the model matches the real-world.

Overview: Open Source Physics Resources

- Open Source Physics: www.compadre.org/OSP/
- Physlet Physics 2E: www.compadre.org/Physlets
- Physlet Quantum Physics 2E: www.compadre.org/PQP
- Tracker Video Analysis Tool: www.cabrillo.edu/~dbrown/tracker/
- Easy Java Simulations: fem.um.es/Ejs/

Overview: Open Source Physics Resources

- Open Source Physics: www.compadre.org/OSP/
- Physlet Physics 2E: www.compadre.org/Physlets
- Physlet Quantum Physics 2E: www.compadre.org/PQP
- Tracker Video Analysis Tool: www.cabrillo.edu/~dbrown/tracker/
- Easy Java Simulations: fem.um.es/Ejs/

Why might we want to use simulations in teaching physics?

Why might we want to use simulations in teaching physics?

Colliding Galaxies: The Mice

How can we use simulations in teaching physics?

- **User:** Students access pre-made simulations that (hopefully) they must interact with.
- **Modeler:** Students are given access to a software package with a simple user interface. Students must then simulate the physics of a problem by modeling at a high level of abstraction. For example, adding the physics in the form of differential equations (rates of change) and initial conditions.
- **Programmer:** Students are given tools to program a physics example using traditional computational physics techniques.

What Level of Course?

What Level of Course?

- **Non-Science Major/Astronomy**
- **Introductory Physics and Astronomy**
- **Major-Level Courses**
- **Introducing Current Research into Courses**

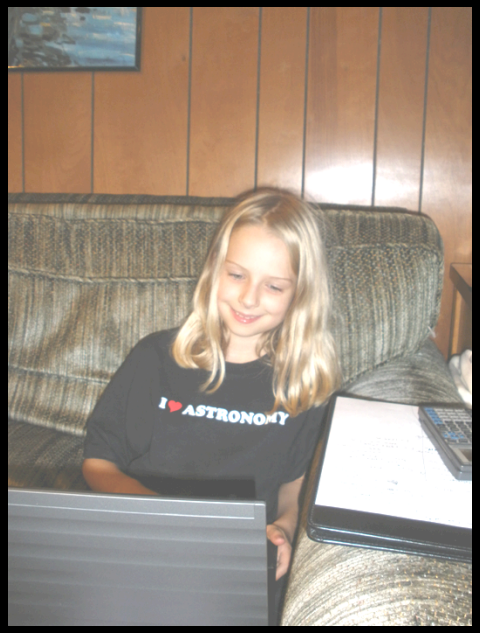
But expectations, outcomes, and scaffolding are different

What version you use will be related to the course you are teaching, your student body, and your expertise.

In general…

…the less sophisticated the student,
the more sophisticated the user interface
...and the more interactive, the better….

...keeping in mind that
technology without pedagogy...
...is just technology.



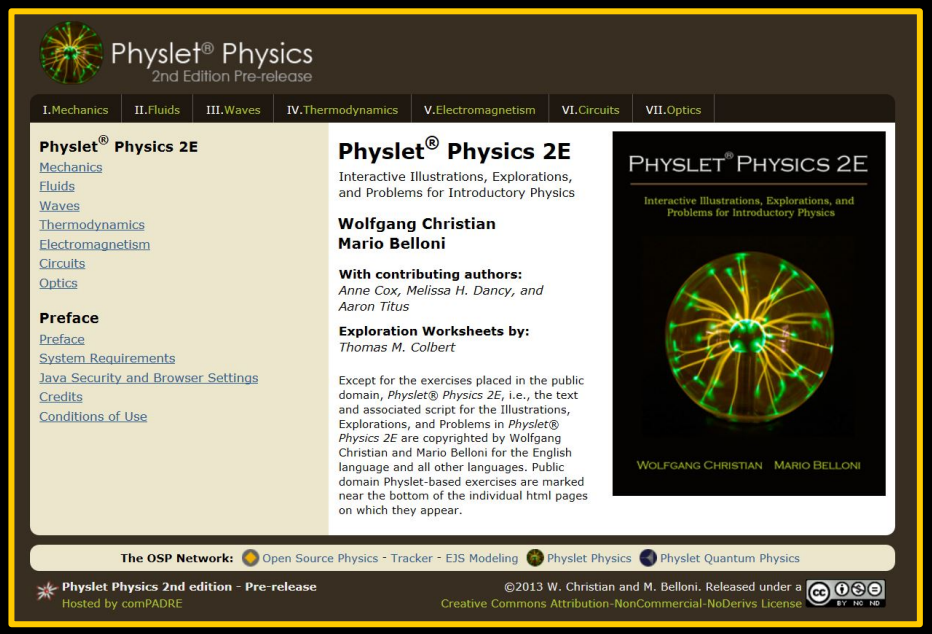
Coupling Simulation with Pedagogy

- **User:** Students access pre-made simulations that they must interact with.
- Pedagogies:
  - Peer Instruction / Think-Pair-Share / Clickers
  - Just-in-Time Teaching
  - Guided Inquiry/Tutorial
  - Group Problem Solving
  - Lecture Demonstration
  - TIPERs (Ranking Tasks, etc)
  - In-class Exercise
  - Homework
  - Laboratory Exercises (pre-lab, in-lab, post-lab)
  - Etc.

PHYSLETS ON COMPADRE

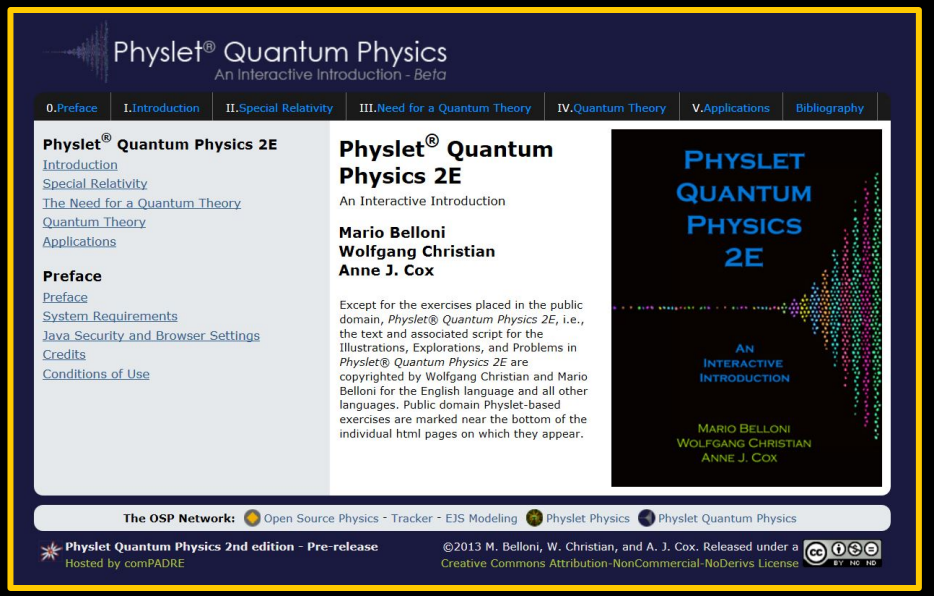
Physlets—"Physics applets"—are small, flexible Java applets that are ...

- **Visual and interactive**
- **Flexible (modular and scriptable with JS)**
- **Uniform with over 2,000 exercises**
- **Pedagogically adaptable**
- **Web based (Mac OSX, Windows, Unix).**
- **Free for noncommercial use.**



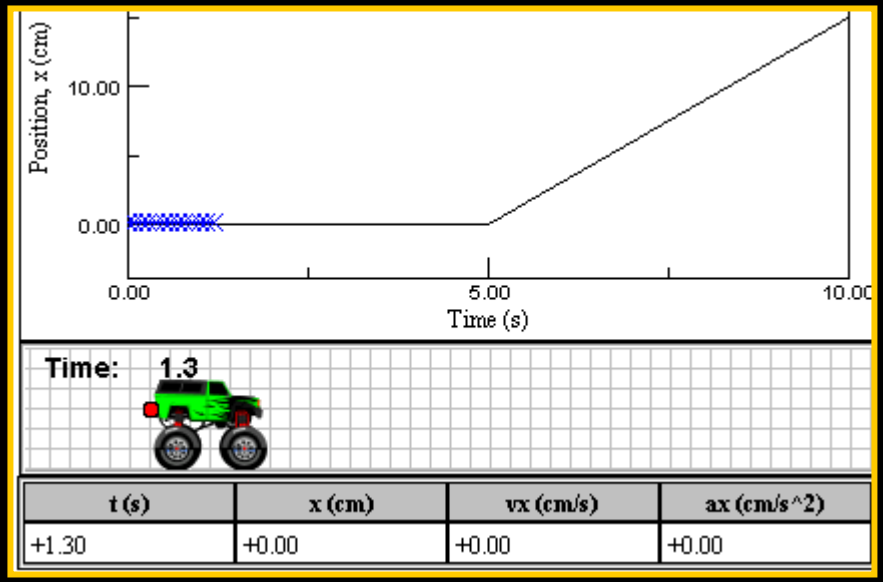
www.compadre.org/Physlets

www.compadre.org/PQP



JiTT Example: Physlets

Use the mouse to drag the rear bumper of the toy monster truck (the red ball). The goal of this exercise is to match the position/ velocity/acceleration vs. time graphs as shown in the animation (**position is given in centimeters and time is given in seconds**). There is some smoothing for the velocity and acceleration matching animations.



t (s)	x (cm)	vx (cm/s)	ax (cm/s^2)
+1.30	+0.00	+0.00	+0.00

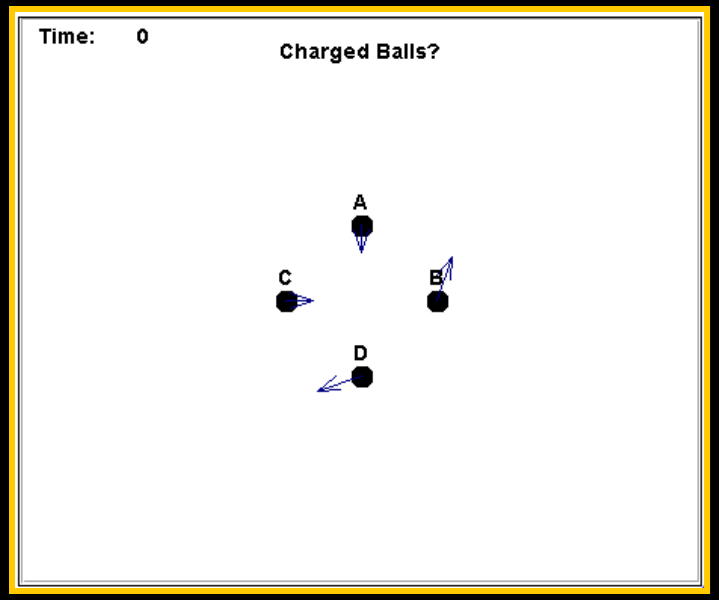
Answer the following questions after trying to match the motion:

a. Which of the graphs was the easiest to match and which one was the hardest to match?

b. Why? Base your answer on physics and mathematics.

http://www.compadre.org/Physlets/mechanics/prob1_3.cfm

JiTT Example: Physlets

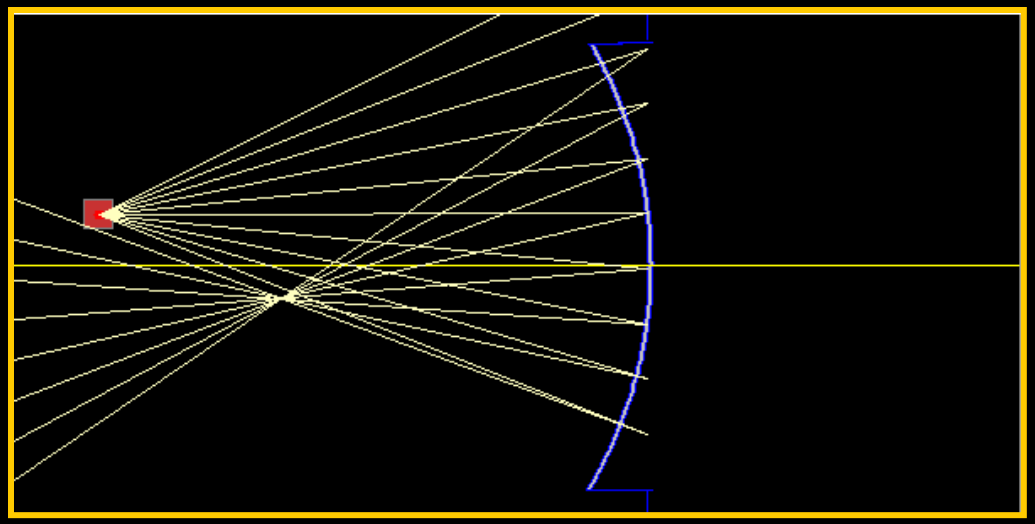


Four charged objects are shown on the screen along with vectors representing the forces on each object. You can click-drag on any object to change its position **(position is given in meters)**.

- What, if anything, is wrong with the animation?

http://www.compadre.org/Physlets/electromagnetism/prob22_2.cfm

JiTT Example: Physlets

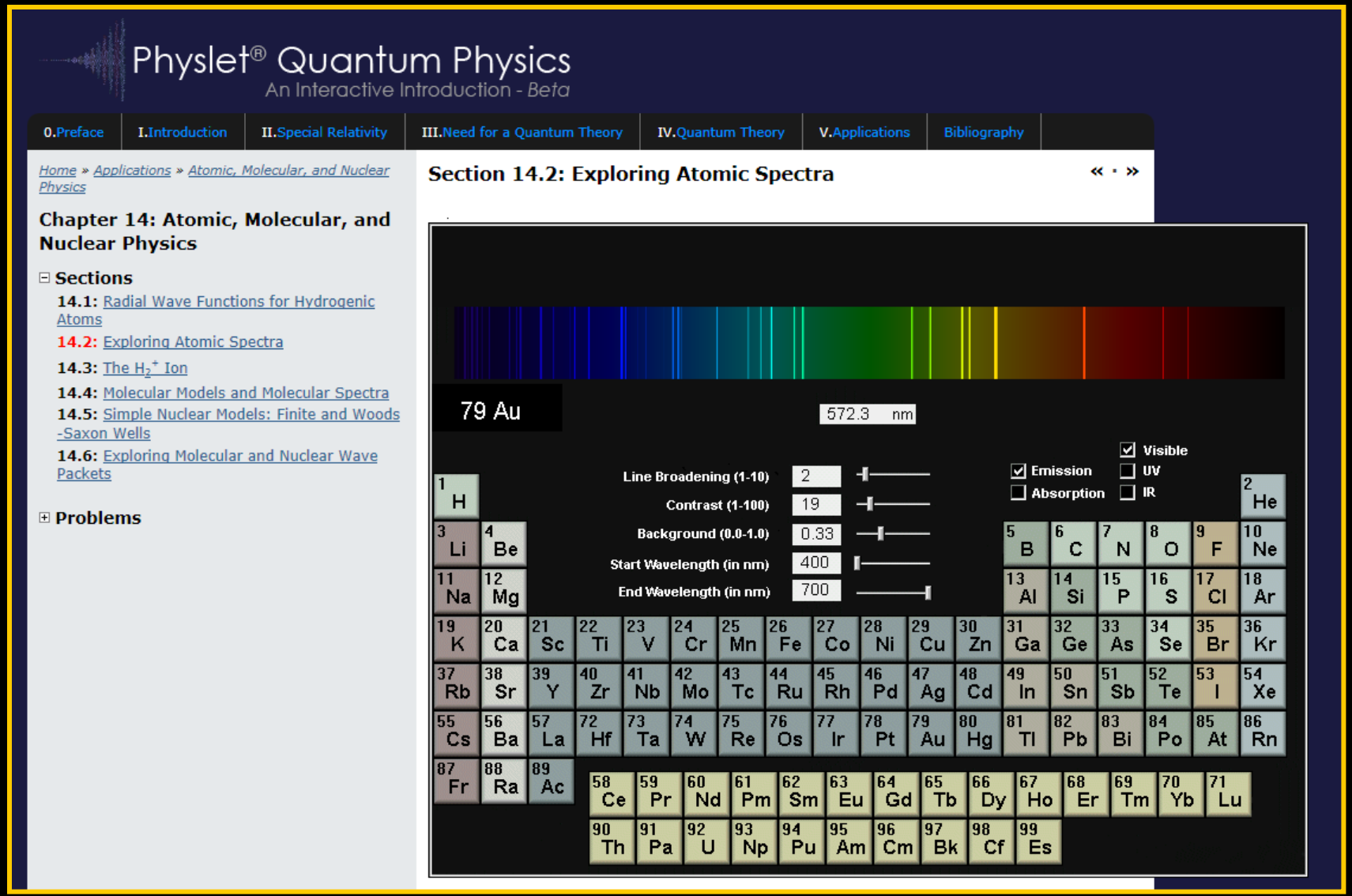


A point source is located to the left of a mirror. You can drag this point source to any position **(position is given in meters and angle is given in degrees)**.

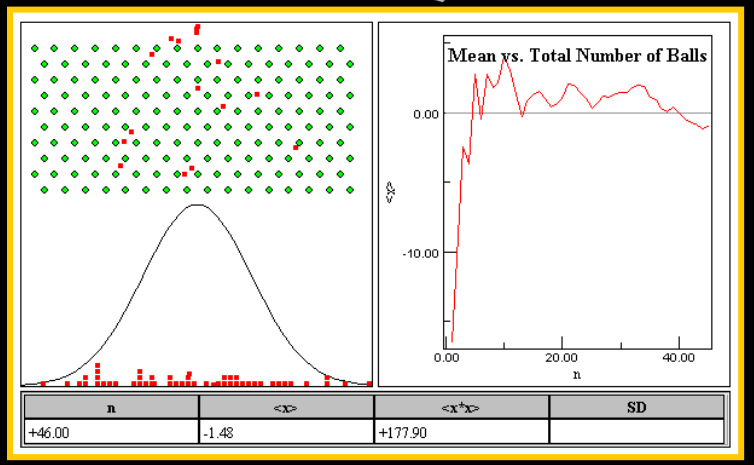
1. Find the focal length of the mirror.
2. Describe the technique you used to determine the focal length.

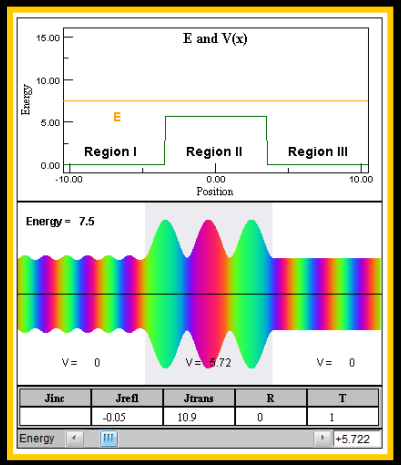
http://www.compadre.org/Physlets/optics/prob33_1.cfm

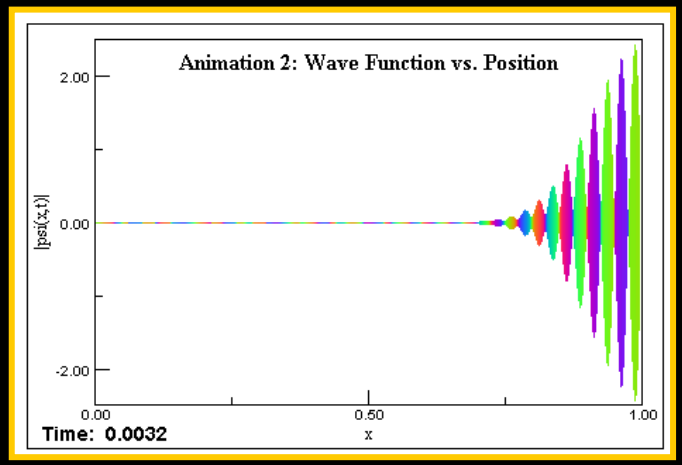
PQP 2E on ComPADRE

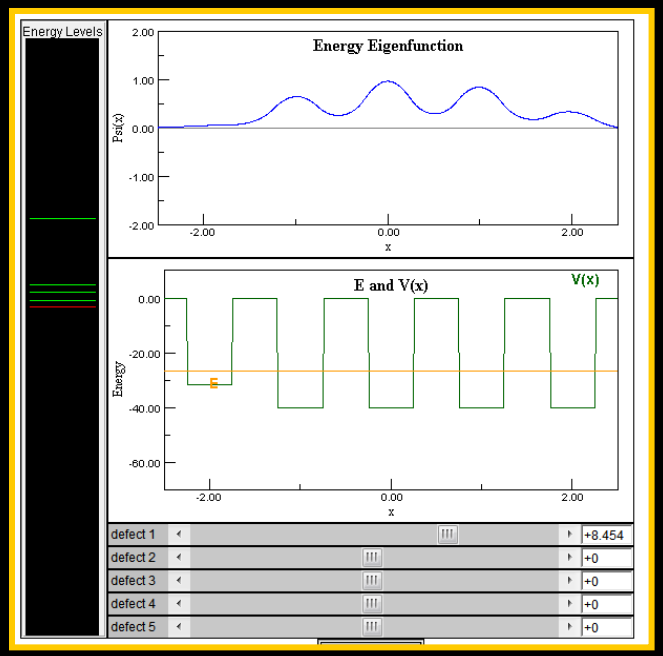


PQP 2E on ComPADRE

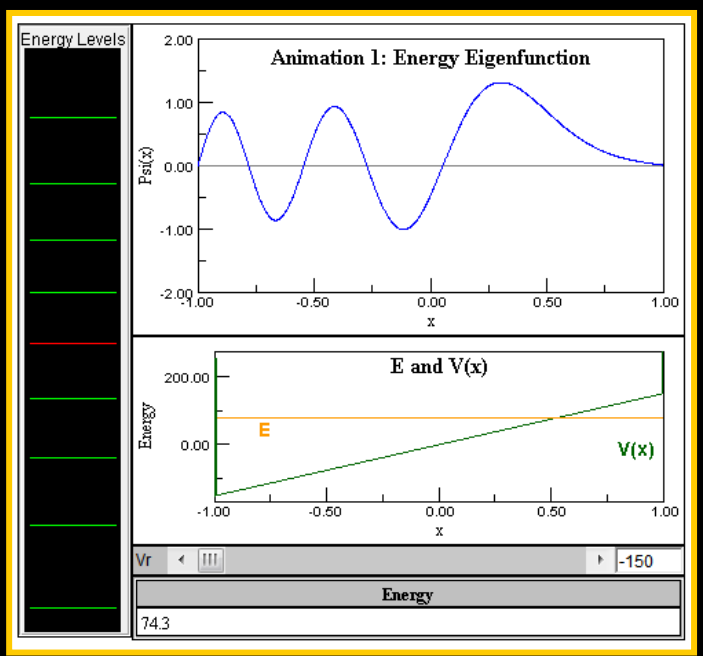




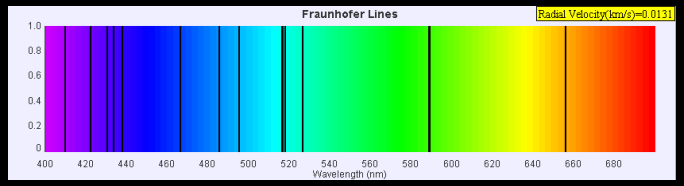


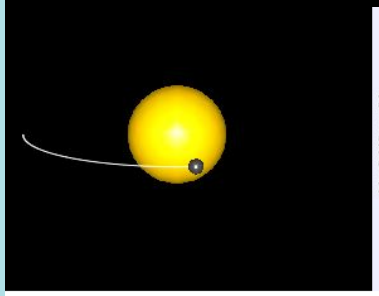


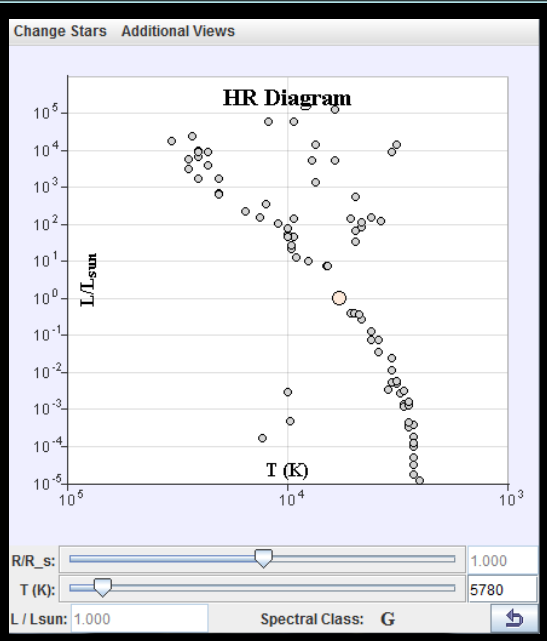
- sp. relativity
- modern physics
- classical/qm
- eigenfunction shape
- time evolution
- scattering
- non-standard qm wells
- atomic physics
- stat mech



JiTT Example: EJS Astronomy







Over 50 simulations for JiTT in intro astronomy on

OSP Collection on ComPADRE

Also organized in

M.B. Astro Filing Cabinet

Shared Folders

Astronomy 105 (2 resources, 14 subfolders)
Materials in Support of a College-Level Introductory Astronomy Course at Davidson College.

A survey of the current scientific view of the Universe. Emphasis on the physical and mathematical principles necessary to understand how astronomers observe and interpret phenomena. Topics include the historical development of major astronomical theories, the interaction of light and matter, the life cycle of stars, and the structure and evolution of the Universe. No laboratory.

Astronomy 105 Course Home Page

This website is the course homepage for the Davidson College Astronomy (PHY 105) course from the Spring of 2012 taught by Mario Belloni. Many of the following materials were used in the teaching of this course during the spring of 2012.

website

Davidson College Astrophotography Project

For the past year, as part of teaching the astronomy class (PHY 105), we have been taking astrophotographs. Follow the link to see both our personal and student photos taken in Davidson, NC either on campus or at the Pine Road Observatory.

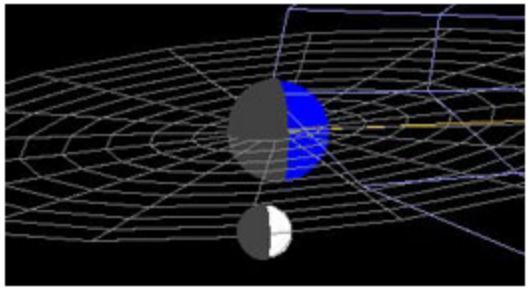
website

Copy selected into:

Astronomy 105 Subfolders

- Naked Eye Astronomy (12)
- Optical (Classical) Astronomy (0)
  - Solar System Models (10)
  - Orbits (7)
  - Optics (11)
- Modern Astronomy (0)
  - Stars and Stellar Properties (6)
  - Stellar Aberration and Parallax (5)
  - Exoplanets (2)
  - Galaxies (9)
  - General Relativity (5)
    - Classical Simulations (3)
    - Schwarzschild Metric Simulations (9)
    - Kerr Metric Simulations (4)

Featured Curriculum Package



Introductory Astronomy Models
A shared file folder of Astronomy models designed for a college-level introductory astronomy course. This shared folder contains over 50 EJS models and is broken up into three parts: naked-eye astronomy, classical astronomy (optics and orbits), and modern astronomy.

OSP Example: EJS Moon Phases

[Moon Phases Simulation]

http://www.compadre.org/OSP/items/detail.cfm?ID=9308

Moon Phases

Computer Program Detail Page Save into folder: Save

Phases of Moon Model

written by Dr. Todd Timberlake

The EJSPhases of Moon model displays the appearance of Moon and how it changes depending on the position of Moon relative to Earth and Sun. The main window shows Earth (at the center) and Moon, as well as a circle tracing out Moon's orbit. Sun is far to the right in this picture and therefore the right side of Earth and Moon are bright while the left sides are dark. By using the Options Menu the Moon View window shows the appearance of Moon as seen from Earth when Moon is in the position shown in the main window. You can modify this simulation if you have Ejs installed by right-clicking within the plot and selecting "Open Ejs Model" from the pop-up menu item.

EJS Phases of Moon model was created using the Easy Java Simulations (Ejs) modeling tool. It is distributed as a ready-to-run (compiled) Java archive. Double clicking the ejs_astronomy_MoonPhases.jar file will run the program if Java is installed. Ejs is a part of the Open Source Physics Project and is designed to make it easier to access, modify, and generate computer models. Additional Ejs models for astronomy are available. They can be found by searching ComPADRE for Open Source Physics, OSP, or Ejs.

Please note that this resource requires at least version 1.5 of Java (JRE).

http://www.compadre.org/OSP/document/ServeFile.cfm?ID=9308&DocID=1...

⊟ View the supplemental documents attached to this resource (3)

- **Phases of Moon Model: Student Version**
  EJS Phases of Moon Model: Student Version is a simulation for physical science (middle and high) school students. It is distributed as a ready-to-run (compiled) Java archive. Double clicking the ejs_MoonPhasesStudent.jar file will run the program if Java is installed.
- **Phases of Moon Model: Lesson Plan**
  A pdf file with a teacher lesson plan for use with the Phases of Moon Model.
- **Phases of Moon Model: Homework Exploration**
  A pdf file with a college-level homework exploration for use with the Phases of Moon Model.

Manage Record

OSP Status: **R**
Control Menu
Documents

Contribute

Make a Comment
Create a Relation
Contact us

Related Materials

Is Based On
Easy Java Simulations Modeling and Authoring Tool

See details...

Similar Materials

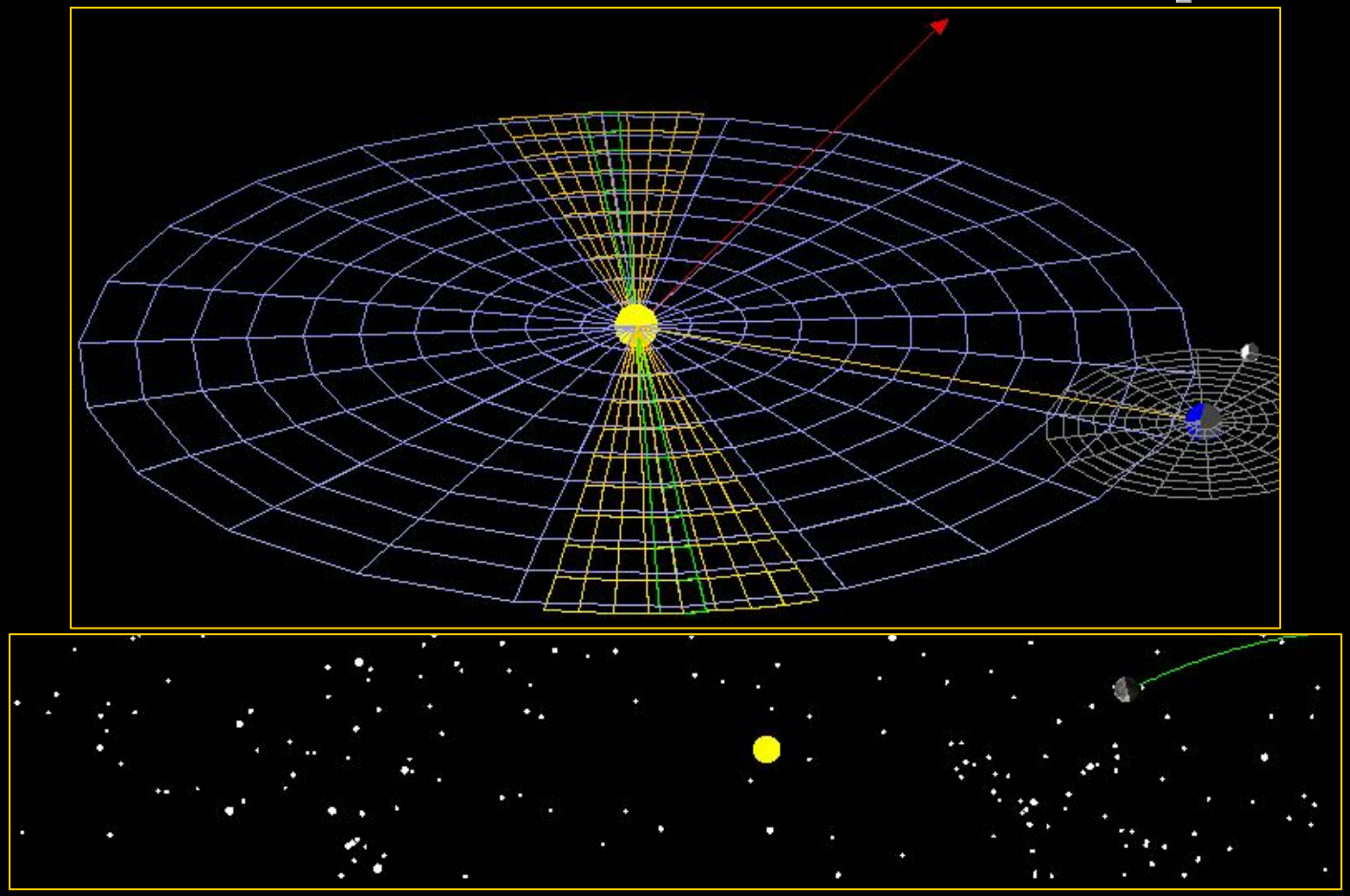
Phases of Venus Model
Superior Ptolemaic Model
Copernican System Model

More...

Featured By

Open Source Physics
Jul 31 - Aug 31, 2010

Solar and Lunar Eclipses



[Solar and Lunar Eclipses Simulation]

On OSP ComPADRE

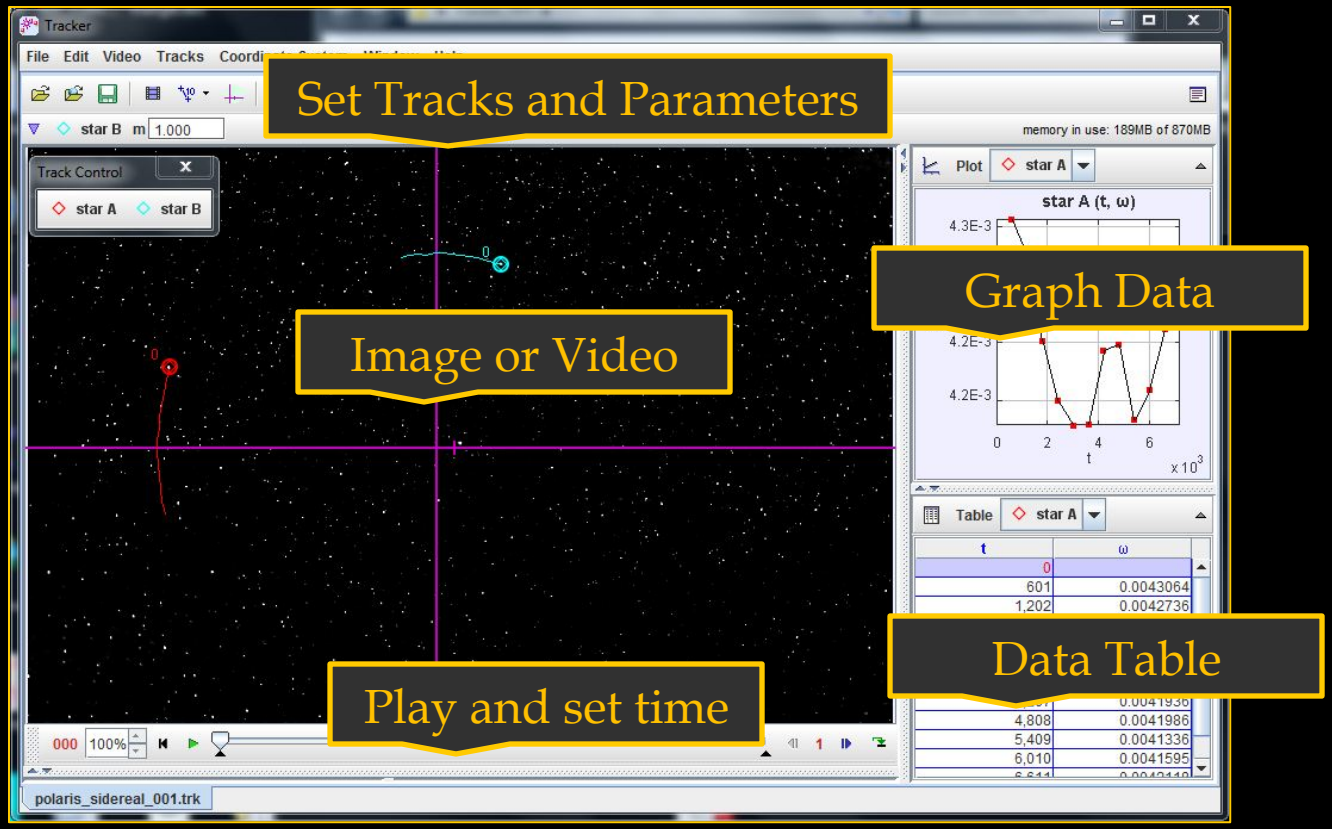
OSP: Coupling Simulation with Pedagogy

Modeler: Students (intro, classical mechanics, computational physics) are given access to a software package with a simple user interface. Students simulate the physics of a problem by modeling at a high level of abstraction. For example, adding the physics in the form of differential equations (rates of change) and initial conditions.

- **Tracker:** is a free and open source video analysis software program.

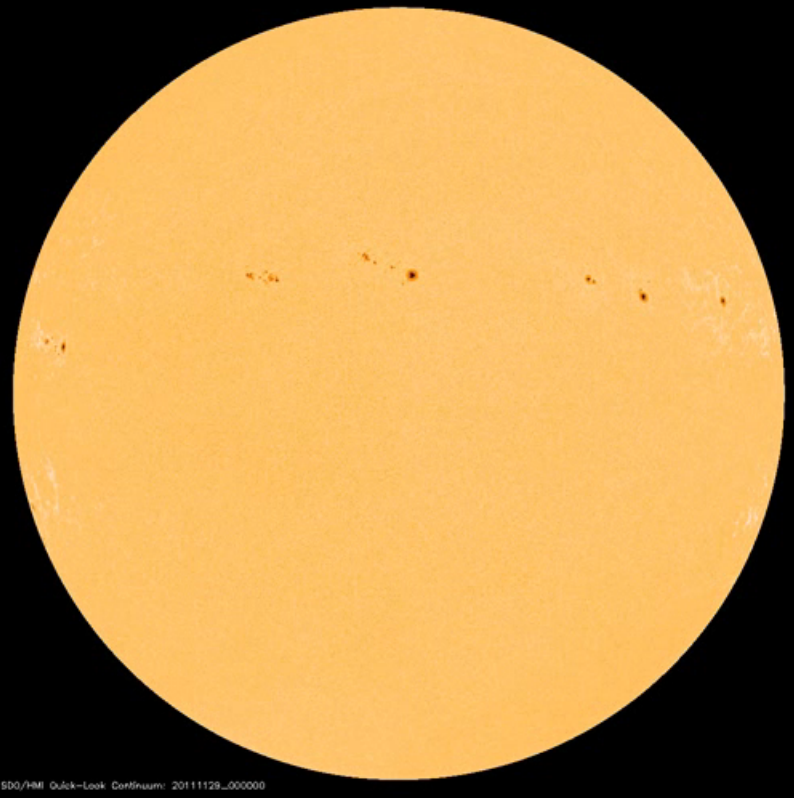
- **Easy Java Simulations:** (EJS) is free open source software that is designed to create interactive simulations in Java (applications and applets) without the necessity of prior programming knowledge to quickly and easily prototype, test, and distribute packages of Java simulations. EJS allows students, teachers, and curriculum authors to easily write and/or change simulations. Can also be used to teach computational physics.

Tracker

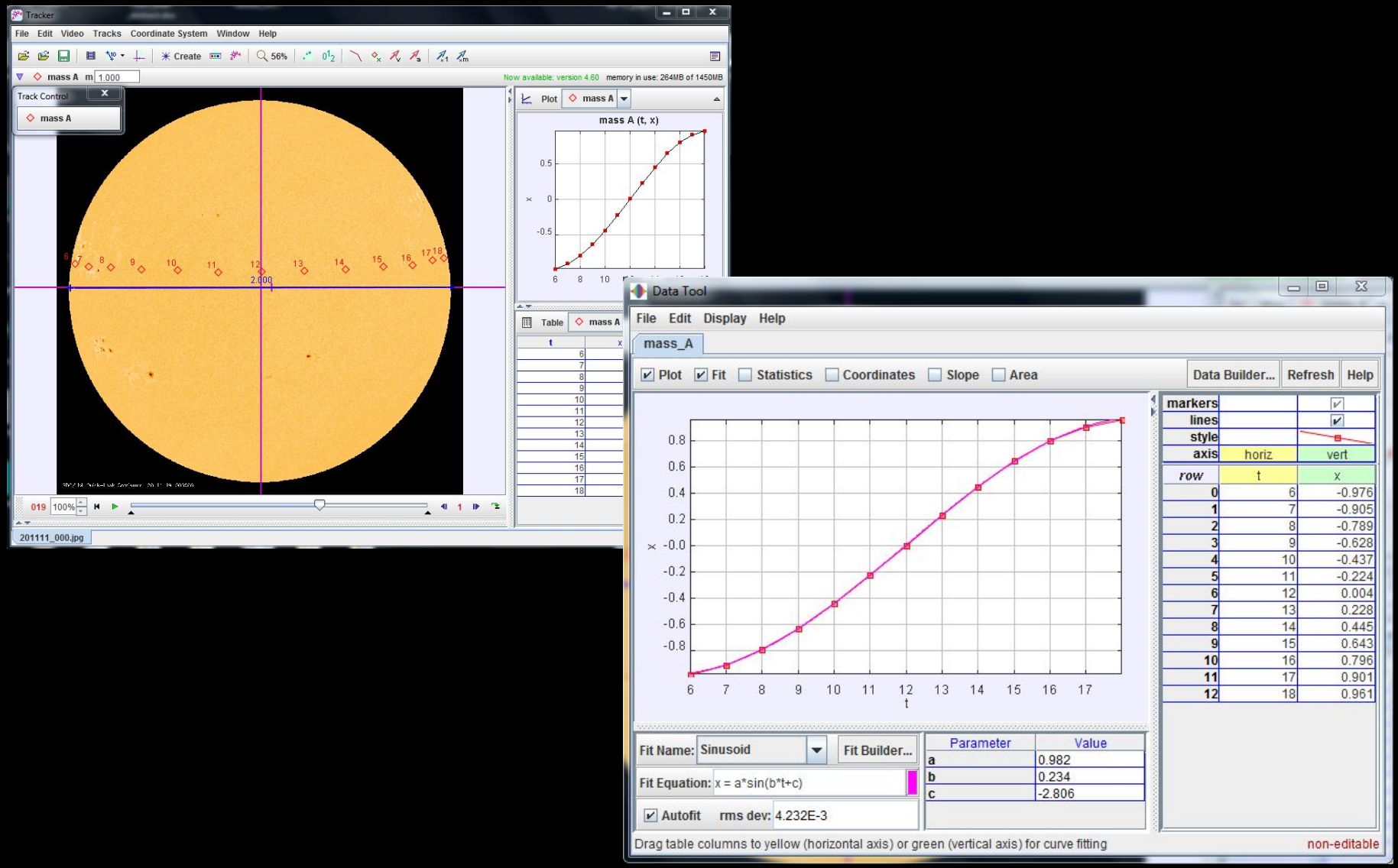


OSP-based free and open source (hmm..) video analysis program. On the OSP Collection there are numerous examples from Angry Birds to solar rotation rates.

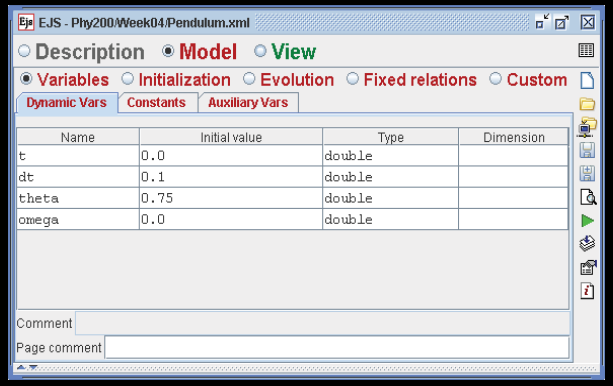
Tracker: Solar Rotation

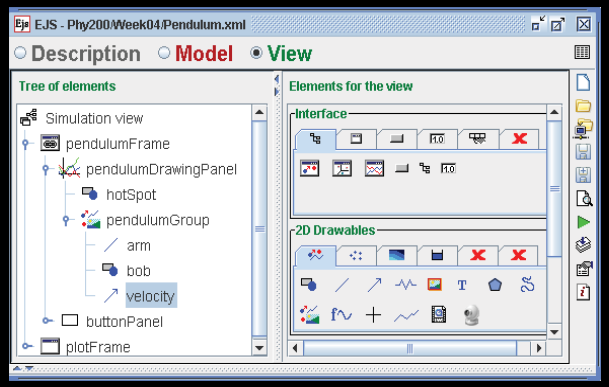


Tracker: Solar Rotation

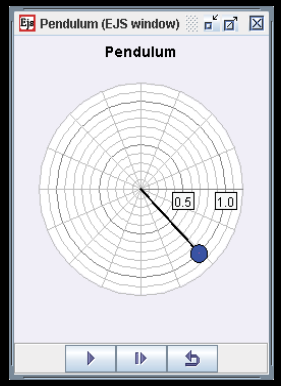


Creating an EJS Simulation



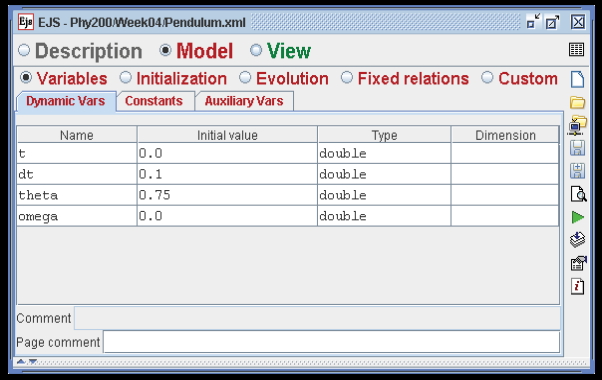


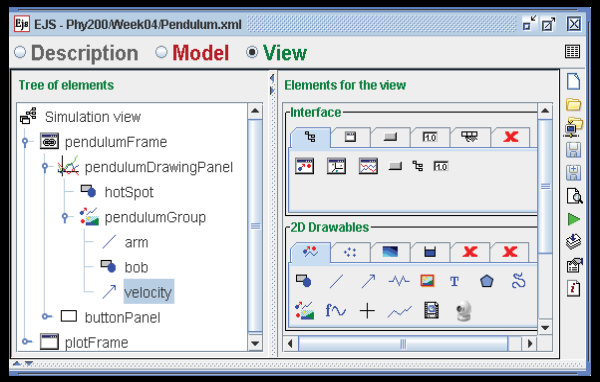
Easy Java Simulations: **(EJS) is free open source software that is designed to create interactive simulations in Java (applications and applets) without the necessity of prior programming knowledge.**

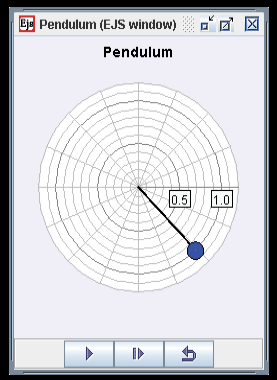


The advantage of EJS for physics teaching is it separates the model into logical parts (variables and evolution) and it separates the model from the view (the visualization of the simulated model).

EJS Simulation in 5 Minutes







Ready, Set, Go….

Characteristics of EJS Models

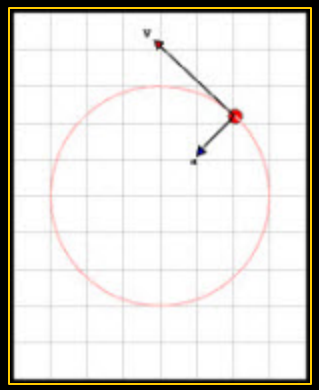
- Can be run:
  - As stand alone programs
  - As applets in HTML pages
  - In an single executable package that bundles many simulations together with curricular materials.
- Are written in Java and are distributed under the GNU GPL license.
- Can be modified with EJS
- Can be distributed from a digital library to a teacher or student desktop authoring tool via the internet.
- Over 400 simulations on the OSP Collection on ComPADRE

But it gets better

- Teacher Modifiable Simulations:
  - Modify the model and save
  - Open or close the user interface as desired
  - Add new curricular materials

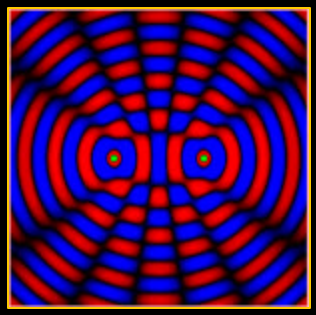
Function Visualizer

Analytic Trajectory

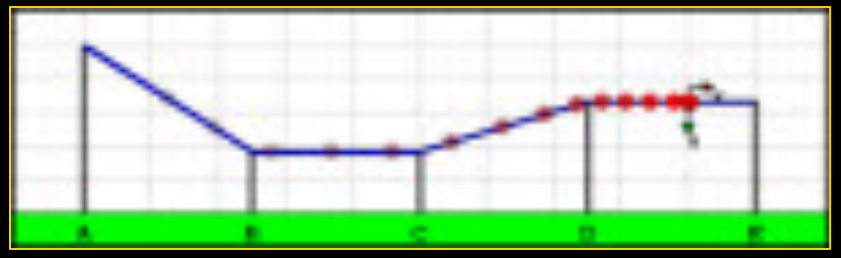


Dynamic Trajectory

Wave Superposition

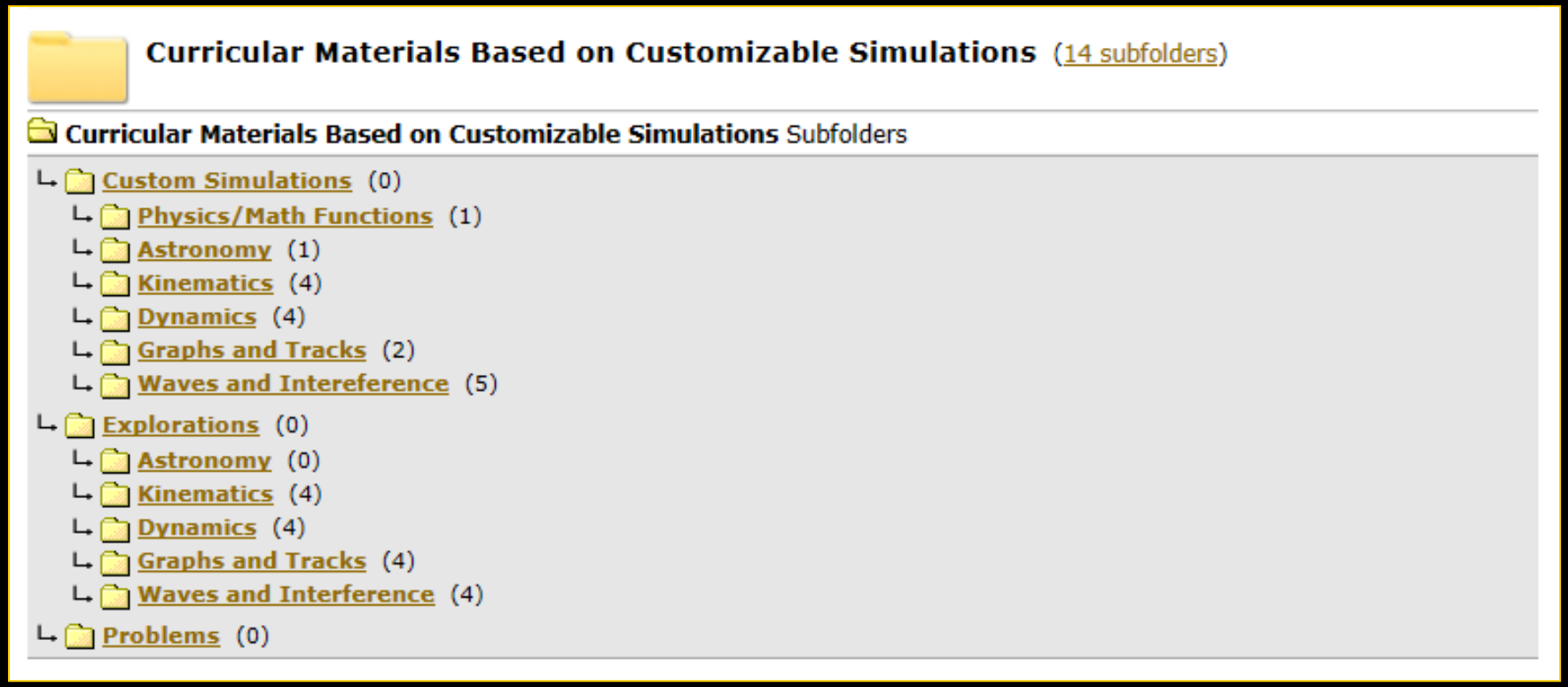


Ripple Tank



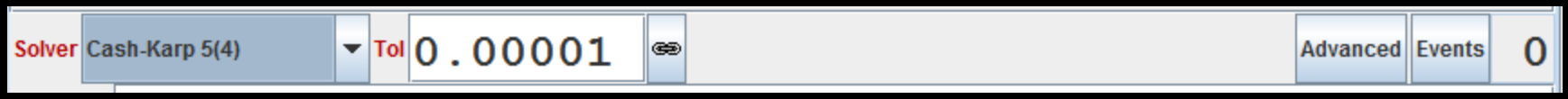
Graphs and Tracks

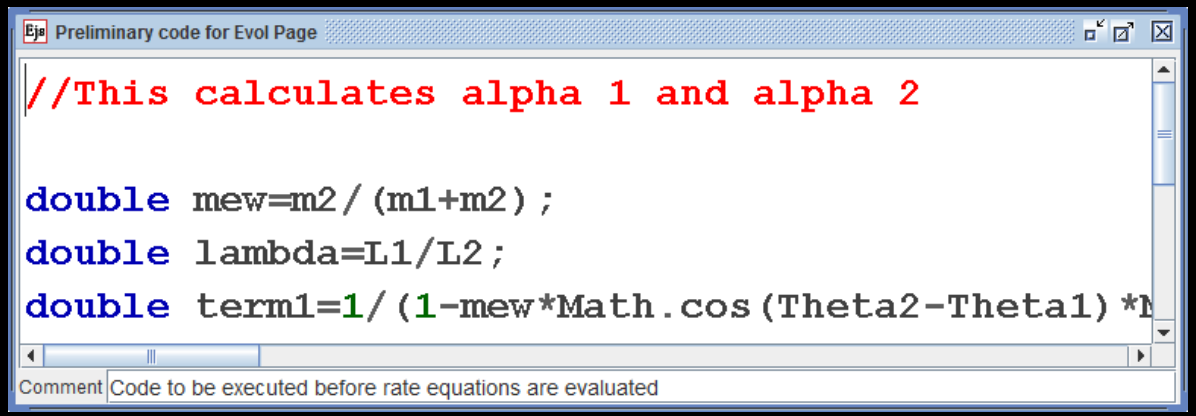
Teacher Modifiable Materials Filing Cabinet

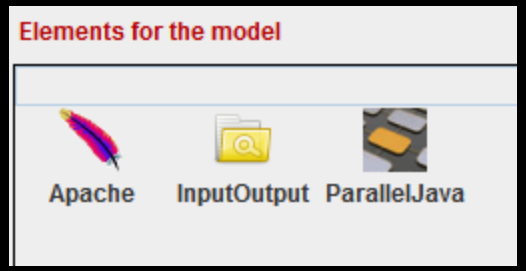


OSP: Coupling Simulation with Pedagogy

Programmer: Students are given tools to program a physics example using traditional computational physics techniques. We use **Easy Java Simulations** since it has as many advanced tools

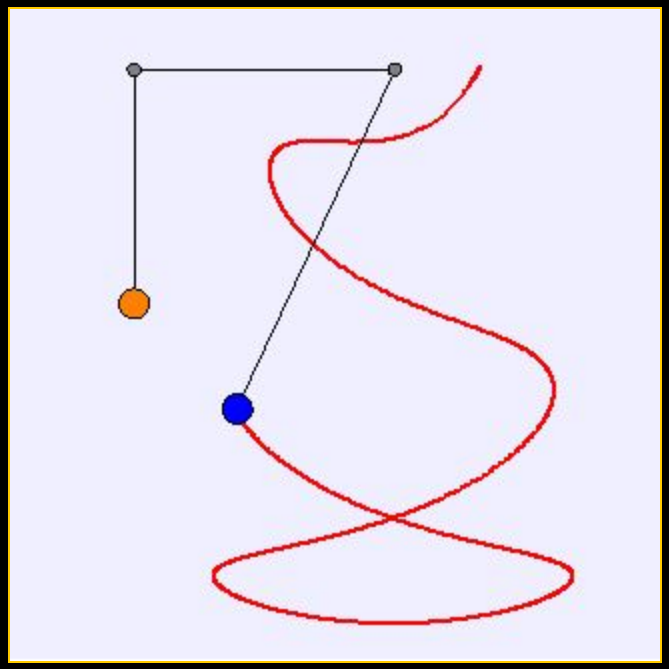


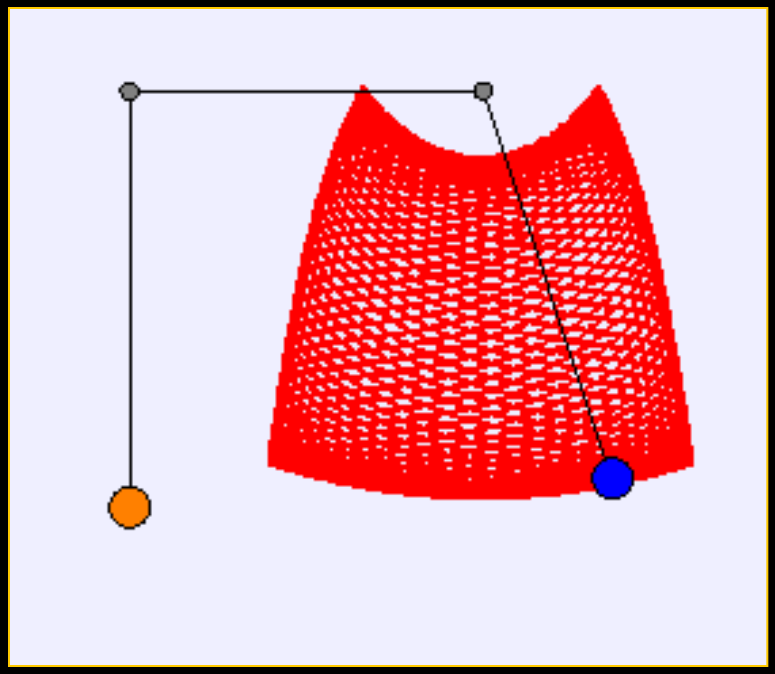





```
public void poincarePanelAction() {
  double x =_view.poincarePlottingPanel.getMouseX();
  double y =_view.poincarePlottingPanel.getMouseY();
  if(!checkValues(x,y,E0)) return;
  if(t!=0){
```

Example of Computational Physics and Experimental Physics: The Swinging Atwood's Machine





[SAM Simulation] [SAM Video] [fSAM Simulation]

On OSP ComPADRE

More Examples of Comp Physics

Wolfgang's Shared Folder — My Folders

Wolfgang's Shared Folders

└ **An Introduction to Computer Simulation Methods -- EJS Edition**

An Introduction to Computer Simulation Methods -- EJS Edition (9 resources, 2 subfolders)

The *Easy Java Simulations* (EJS) adaptation of an *An Introduction to Computer Simulation Methods* by Harvey Gould, Jan Tobochnik, and Wolfgang Christian emphasizes physics modeling by example. We have chosen EJS for this edition because its dynamic and highly interactive user interface greatly reduces the amount of programming required to implement an idea. EJS is a Java program that enables both programmers and novices to quickly and easily prototype, test, and distribute packages of Java simulations. EJS gently introduces students to Java syntax but even experienced programmers find it useful because it is faster and easier program in EJS than in other environments.

EJS CSM Textbook Chapter 1: Introduction to modeling

Chapter 1 introduces the *Easy Java Simulations* (EJS) ed
Simulation Methods and discusses the importance of con
computer simulation.

detail page - download

EJS CSM Textbook Chapter 2: Creating simulations

Chapter 2 introduces Java syntax and EJS elements in th
particles near the Earth's surface.

detail page - download

EJS CSM Textbook Chapter 3: Simulating Particle M

Chapter 3 presents several numerical methods needed t
Newton's laws and introduces the Ordinary Differential E
to select different numerical algorithms for ODE-based m
to model motion in three dimensions.

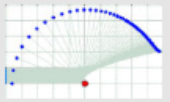
detail page - download

EJS CSM Textbook Chapter 4: Oscillations

Chapter 4 explores the behavior of oscillatory systems, i
simple pendulum, and electrical circuits. We introduce t
the EJS ODE editor is used to solve arrays of differential

detail page - download

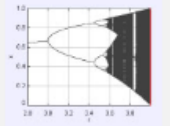
EJS CSM Textbook Chapter 5: Few-Body Problems



Chapter 5 applies Newton's laws of motion to planetary motion and other systems of a few particles and explores some of the counter-intuitive consequences.

detail page - download

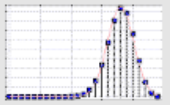
EJS CSM Textbook Chapter 6: The Chaotic Motion of Dynamical Systems



Chapter 6 studies simple nonlinear deterministic models that exhibit chaotic behavior. We will find that the use of the computer to do numerical experiments will help us gain insight into the nature of chaos.

detail page - download

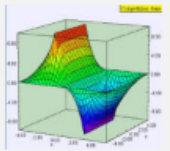
EJS CSM Textbook Chapter 7: Random Processes



Chapter 7 introduces Random processes in the context of several simple physical systems, including random walks on a lattice, polymers, and diffusion controlled chemical reactions. The generation of random number sequences also is discussed.

detail page - download

EJS CSM Textbook Chapter 10: Electrodynamics



Chapter 10 computes the electric fields due to static and moving charges, describes methods for computing the electric potential in boundary value problems, and solves Maxwell's equations numerically.

detail page - download

EJS CSM Textbook Chapter 17: Visualization and Rigid Body Dynamics

Chapter 17 studies affine transformations in order to visualize objects in three dimensions. We then solve Euler's equation of motion for rigid body dynamics using the quaternion representation of rotations.

detail page - download

Possible Issues?

OSP Network Team

- **OSP Managing Editor:**
  - Wolfgang Christian - Davidson College
- **OSP, Physlets, and EJS Curriculum Authors:**
  - Mario Belloni - Davidson College
  - Anne Cox - Eckerd College
  - Fu-Kwun Hwang – National Taiwan Normal University
  - Harvey Gould - Clark University
  - Jan Tobochnik - Kalamazoo College
- **Tracker and OSP Tools Developer:**
  - Doug Brown - Cabrillo College
- **Easy Java Simulations Developer:**
  - Francisco Esquembre - Universidad de Murcia
- **OSP Java Library Editor:**
  - Wolfgang Christian - Davidson College
- **ComPADRE :**
  - Bruce Mason – University of Oklahoma
  - Lyle Barbato- AAPT
  - Matt Riggsbee- AAPT
  - Caroline Hall- AAPT

AAPT American Association of Physics Teachers | AMERICAN INSTITUTE OF PHYSICS

AAS American Astronomical Society | APS physics American Physical Society

The Open Source Physics Project

Visit us at:

www.compadre.org/osp
www.compadre.org/Physlets
http://www.compadre.org/PQP
www.um.es/fem/Ejs/
www.cabrillo.edu/~dbrown/tracker/

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