# Progress in Easy-to-use 3D Programming Environments

Bruce Sherwood (North Carolina State University)
Steve Spicklemire (University of Indianapolis)

## VPython

**3D programming for ordinary mortals**

- Python + IDLE + visual (3D graphics module)
- Easy to learn
- Vector computations
- Navigable 3D animations as a side effect
- Object-oriented
- Free, open source, runs on all platforms
  - Uses OpenGL graphics library

http://vpython.org

## The Newtonian synthesis

Momentum principle + force law + initial conditions

Iterative update of momentum and position

Intro level: Simple Euler-Cromer algorithm is adequate: simply decrease step size if behaviour is erratic.

Matter & Interactions, 3d Edition

Programs written by students include

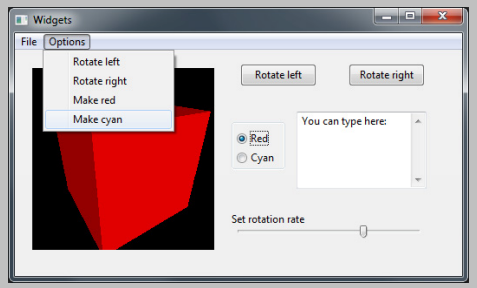
Motion of a fancart (constant force)
Spacecraft orbit
Spacecraft orbit with energy graphs
3D Mass-spring
3D Mass-spring with energy graphs
Rutherford scattering with momentum graphs
Statistical mechanics of Einstein solid
Electric field of point charge
Electric field of dipole
Electric field of uniformly charged rod
Magnetic field of moving proton
Charge motion in uniform magnetic field
Positron motion in electromagnetic plane wave

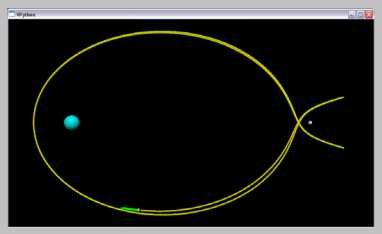
http://matterandinteractions.org
http://www.compadre.org (search for VPython)

## New with VPython 6

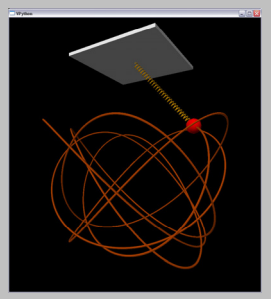
**Based on wxPython, cross-platform GUI library; makes possible adding buttons, sliders, etc.**

**Support for 64-bit Python**

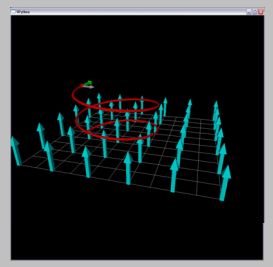


*Student program to compute and display the motion of a spacecraft near a fixed Earth and Moon. The green arrow represents momentum.*



*Student program to compute and display the motion of a 3D mass-spring system.*



*Student program to compute and display the motion of a proton in a uniform magnetic field. Arrows represent momentum and magnetic force.*

**Vector algebra ⇒ `VPython code`**

$$\vec{r} = \vec{r}_{\text{Earth}} - \vec{r}_{\text{Sun}}$$

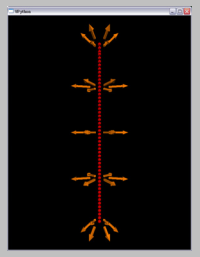
$$\hat{r} = \frac{\vec{r}}{|\vec{r}|}$$

$$\vec{F} = -\frac{GM_E M_S}{|\vec{r}|^2}\hat{r}$$

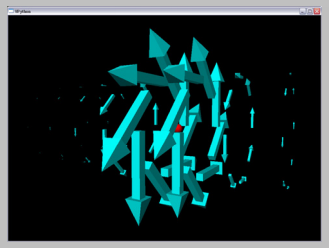
$$\vec{p} = \vec{p}_i + \vec{F}\Delta t$$

$$\vec{r} = \vec{r}_i + \left(\frac{\vec{p}}{m}\right)\Delta t$$

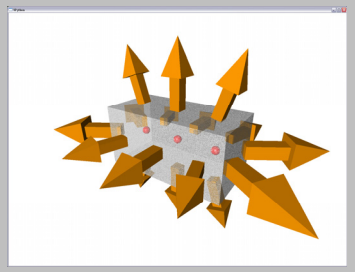
```
r = Earth.pos - Sun.pos

rhat = r/mag(r)

F = -(G*Earth.mass*Sun.mass/mag(r)**2)*rhat

Earth.p = Earth.p + F*deltat

Earth.pos = Earth.pos + (Earth.p/Earth.mass)*deltat
```



*Student program to calculate and display the electric field due to a uniformly charged rod at various locations. The rod is approximated by a number of discrete point charges; students vary the number of charges, and study the effect on their computed results.*



*Student program to compute and display the magnetic field of a moving proton at many locations, dynamically.*



*Lecture demo program illustrating the electric field on a transparent Gaussian surface.*

## GlowScript

- Similar to VPython, but runs in modern browsers
  - Requires modern graphics card with GPUs
- Program in JavaScript or in CoffeeScript
- Easy to learn
- Vector computations
- Navigable 3D animations as a side effect
- Object-oriented
- Free, open source, runs on all platforms
  - Uses WebGL graphics library

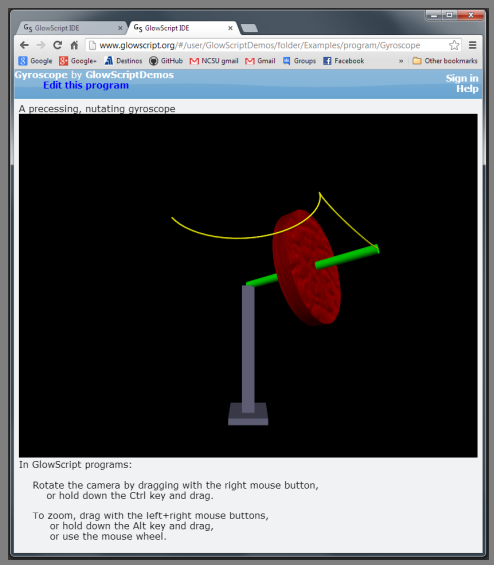
http://glowscript.org

## Sharing web programs

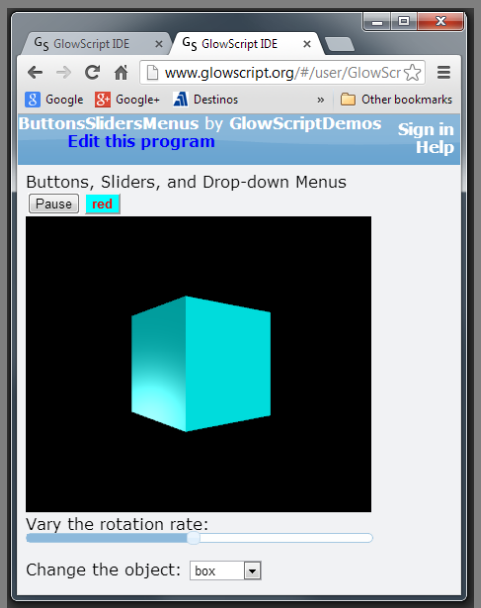
**Can write and run programs in the cloud at http://glowscript.org**

**Email a link to friends who can run your program in their browser with a simple click; no installation**

**For example, here is an animation of a gyroscope: http://www.glowscript.org/#/user/GlowScriptDemos/folder/Examples/program/Gyroscope**




**GlowScript supports standard web elements such as buttons, sliders, and drop-down menus:**




**Can also embed 3D animations in your own web page**

**Special thanks to David Scherer and Ruth Chabay**

Supported in part by the National Science Foundation.