

INTERACTIVE JAVA COMPUTER SOFTWARE FOR EARTH SCIENCE EDUCATION AND RESEARCH

Glenn A. Richard

Glenn.Richard@sunysb.edu

Earth Science Educational Resource Center

Center for High Pressure Research

Department of Geosciences

SUNY at Stony Brook

Stony Brook, NY 11794-2100

Janet L. Kaczmarek

Janet.Niebling@sunysb.edu

Earth Science Educational Resource Center

Center for High Pressure Research

Department of Geosciences

SUNY at Stony Brook

Stony Brook, NY 11794-2100

Summary

The Earth Science Educational Resource Center (ESERC) cooperates with other offices on the SUNY Stony Brook campus to offer several programs that generate software designed for use in earth science education and research. These programs are:

- Project Java
- GEO 327: Computerized Modeling of Geological Phenomena
- GEO 511: Computer Programming for the Geosciences

The software is written in the Java programming language that was developed by Sun Microsystems. As a platform independent language, Java is an excellent tool for providing interactive content on the World Wide Web.

In addition to its educational activities that actually create software, ESERC offers numerous other programs, some of which utilize this software as educational tools. In contrast to static printed documents, software enables students and researchers to interact with models of phenomena, equipment, and mathematical content by entering input and observing results.

ESERC, established in 1994, is the educational outreach component of the Center for High Pressure Research, which is a National Science Foundation Science and Technology Center.

Rationale

Computer technology is a rapidly evolving educational medium. It provides us with extremely powerful learning environments not otherwise available at a reasonable cost. The interactive use of computers can be of major value in science education. One major advantage of Java applets as compared to platform-specific software is their accessibility via the ubiquitous World Wide Web. Every teacher can incorporate this learning tool into their classroom to complement and facilitate lessons. Under such circumstances, students can take a more active role in learning.

With declining percentages of students showing an interest in science, stimulating interest is an important national and international concern (Bork, 1993). Interactive computer programs hold the interests of students for longer periods of time, increasing the time devoted to learning, and decreasing the time consumed by discipline. These are extremely useful tools for the topics that are more difficult to address with static overheads or illustrations.

Making learning more interactive and more individualized are critical challenges in all science education. With resources that facilitate interactive learning that focus on analytical content, teachers can move away from memory-based science and focus on students' thinking processes. The devotion of individual attention to students is a key to the success of interactive learning material. Few students receive this quality of attention in standard lecture-based classes. With good technology-based material, we can provide such aid for all students, facilitating full mastery of scientific topics for all young people, regardless of background (Bork, 1993).

Science education in the United States is currently held in low esteem by many people. Insufficient attention is devoted to science in the elementary and middle schools, and much emphasis is on rote learning. However, new approaches to science education using highly interactive learning modules made possible by widespread availability of the personal computer, have increased the potential for improving science education. The future outcome will be a "scientifically literate" public that understands science and is equipped to make informed decisions within a societal infrastructure that is heavily dependent upon science and engineering.

The Java Programming Language

The [Java](#) programming language was developed by [Sun Microsystems](#) starting in 1991, originally as a means to control consumer devices. Initially James Gosling, one of the developers of the language, called it "Oak" because he liked a particular oak tree that was right outside his window at Sun Microsystems (Horstmann and Cornell, 1999). However, it was later discovered that there was already a programming language by that name. While a group of employees from Sun were visiting a local coffee shop, the name "Java" was suggested and its stuck (Deitel and Deitel, 1999).

Java is designed to be platform independent, therefore it is well-suited for developing programs that can be run on the World Wide Web. The [Java 2 SDK](#) is a development environment that is available as a free download from Sun Microsystems, but it does not

offer a visually-based development interface like some commercial products. It does, however, include all of the standard Java library packages and is actually a better learning environment than the commercial products. In fact, Java programs created using the Java 2 SDK can include graphics, images, buttons, sliders, menus, icons and any of the other features that appear in programs developed using more sophisticated tools.

A number of Web sites are entirely devoted to Java. Some of the most noteworthy ones that include an abundance of applets with source code are the following:

- [Gamelan](#)
- [JARS](#)
- [Java Boutique](#)

The Java Computing Laboratory

In 1998, Sun Microsystems awarded Dr. Martin Schoonen and Glenn Richard an academic equipment grant consisting of computer hardware and software to equip ESERC with a state-of-the-art Java Computing Lab. This laboratory, supervised by Glenn Richard, consists of a server, five workstations and two JavaStations, and provides ESERC with the capability to develop, test, and host applets that serve to explain scientific and, in particular, geological concepts.

Project Java

The World Wide Web is an effective means of making educational software available to students of all educational levels (Richard, 1997). [Project Java](#) was initiated in the summer of 1996 by ESERC and the Long Island Consortium for Interconnected Learning (LICIL) in order to create educational software that could be posted on the World Wide Web. Like CHiPR, LICIL is funded by the National Science Foundation. Some of the software was to be especially designed to serve as educational tools for existing ESERC programs. In order to provide an initial software development team for Project Java, Dr. Peter Henderson, then of the [Department of Computer Science](#), arranged for three undergraduate computer science majors to receive a combination of academic credit and pay to work with Glenn Richard, Educational Coordinator of ESERC, during the summer of 1996 and the following academic year. These students were Konstantin Lukin, Yevgeniy Miretskiy, and Mikhail Sabaev. Between June, 1996 and May, 1997, they learned Java and developed the following programs that were posted on the Java Web site:

- Radioactive Decay
- The Diamond Thief Game
- Soil Moisture
- Impact Craters
- Earthquake Epicenters
- Bragg's Law and Diffraction
- Plate Tectonics
- Waveform Interaction

- Lines: A Game of Strategy
- Pollutant in a Pond System
- High Pressure Laboratory

These programs are all examples of applets, which are "Java programs that work on Web pages" (Horstmann and Cornell, 1999). Alternatively, Java programs can be written as applications, which are executed independently of Web browsers. In order to create a Java applet or application, a programmer writes Java source code and saves it as text in appropriately named files. The programmer then uses a Java compiler to translate the source code into bytecode, which is stored in additional files. After the bytecode files are created, they can be copied onto a Web server, along with HTML files that represent the documents that are designed to display the applets. The HTML files invoke the applets via tags that refer to the appropriate bytecode files.

The High Pressure Laboratory, Waveform Interaction, and Bragg's Law and Diffraction applets were intended to serve as educational tools for ESERC's Let's Make Diamonds! educational program. Participants in this program conduct an experiment designed to convert graphite into diamond, using high pressure and temperature instruments that CHiPR uses in much of its research.

One of the most important concepts that students in the Let's Make Diamonds! program need to understand is that states of matter depend on temperature and pressure. The temperature-pressure spaces of particular substances can be described by phase diagrams. Many of the high pressure experiments conducted by CHiPR are aimed at collecting data that can be used to develop phase diagrams of materials that occur in the earth's mantle. Before a Let's Make Diamonds! experiment begins, the High Pressure Laboratory applet (figure 1) is used to present participants with the opportunity to conduct virtual high pressure and temperature experiments. A material can be chosen from a list in the upper-right hand corner of the applet window. The arrows in the lower-left corner are used to navigate the temperature-pressure space, while the temperature, pressure, and stable phase of the substance are updated and indicated along the bottom margin. We have found that this applet helps students greatly in their understanding of phase diagrams.

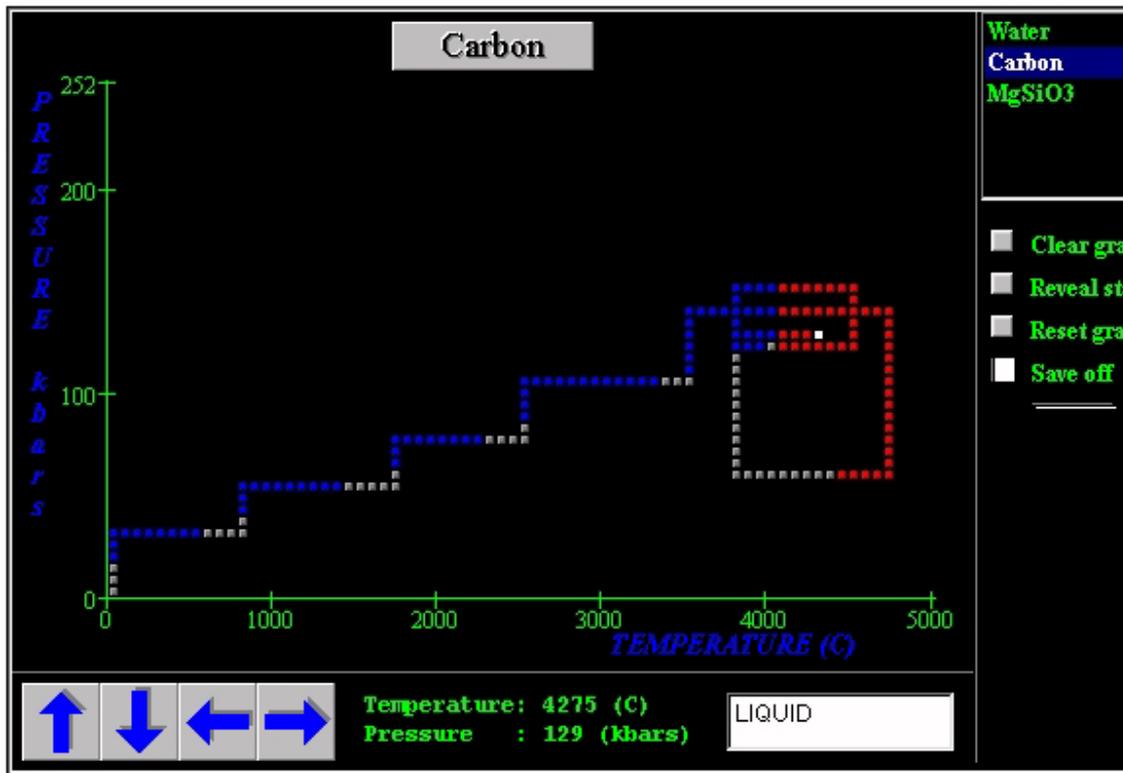


Figure 1. The High Pressure Laboratory applet

The Bragg's Law and Diffraction applet (Figure 2) is designed to demonstrate principles of x-ray diffraction. After the Let's Make Diamonds! experiment, students test the resultant product in order to determine whether they did, in fact, produce diamonds. Some participating groups of students use x-ray diffraction in order to determine what materials are present in the sample after they retrieve it from the press. The applet enables students to choose an x-ray wavelength and a d-spacing, which is the distance between planes of atoms in a crystal. Assuming they have chosen the actual wavelength of the x-rays that they will be using for analysis and a d-spacing that is characteristic of a candidate product of the experiment, they can then incrementally vary the theta angle, and obtain a result that will be useful in evaluating the experiment. The scattered beams illustrated in the applet are in phase at a theta angle that will correspond to a peak in the x-ray output chart for the candidate substance. By matching the actual printouts from the x-rays of the experimental product with the results from the applet, students can determine what substances are present in the product sample.

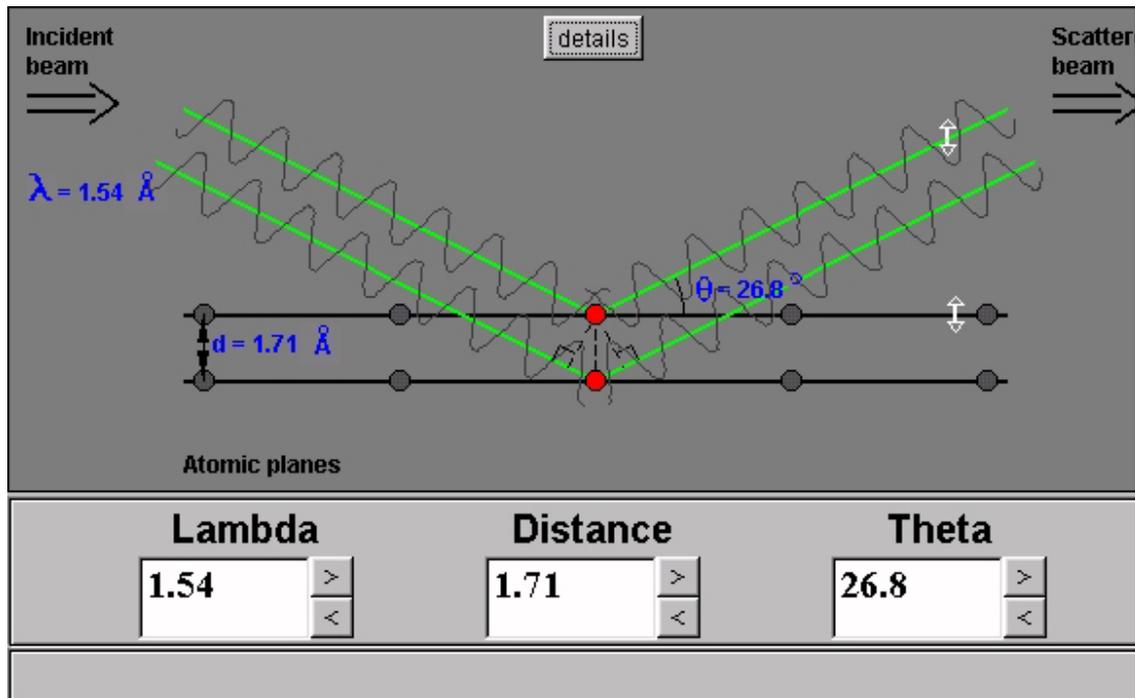


Figure 2. The Bragg's Law and Diffraction applet

In order for participants in Let's Make Diamonds! to gain an understanding of how waves interact to produce constructive or destructive interference during the x-ray analysis, they can use the Waveform Interaction applet (figure 3). Two sine waves and their sum are illustrated in the applet window. Using sliders, users can alter the wavelengths, amplitudes, and phases of the summed waves. The sum is adjusted along with these waves to reflect the new configuration. Sliders and a start and stop button can be used to animate the waves.

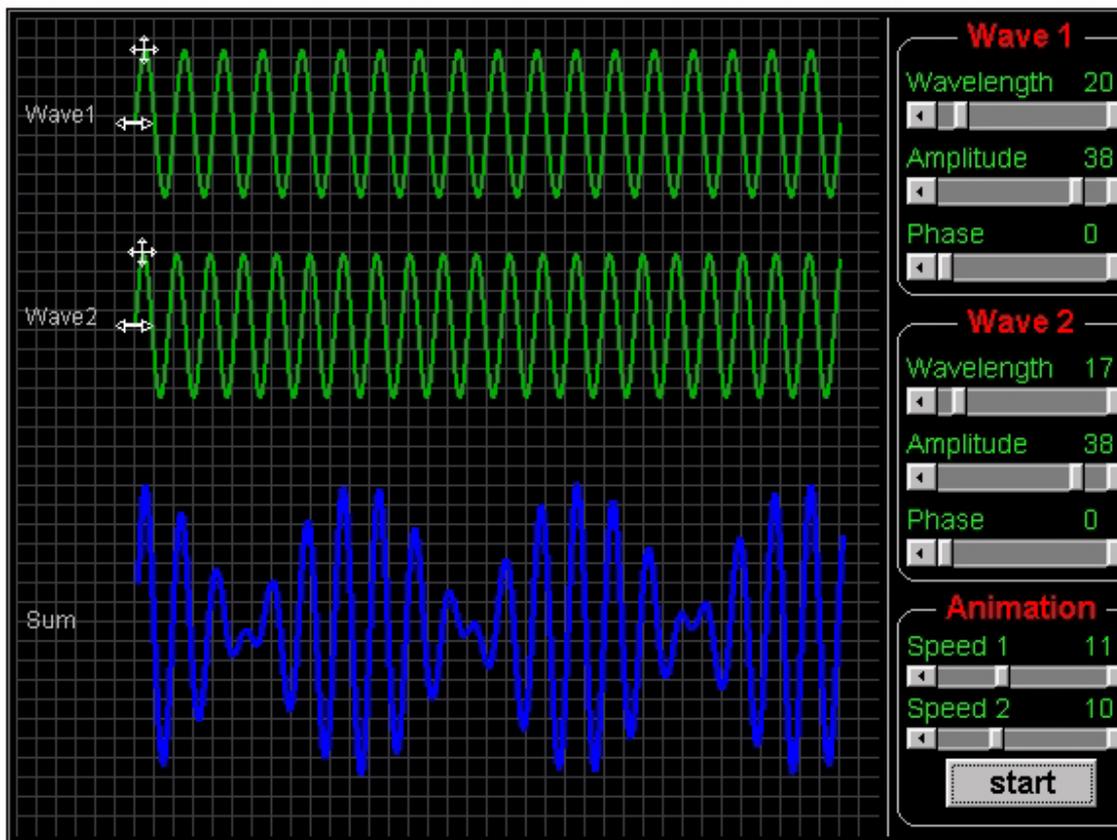


Figure 3. The Waveform Interaction applet

The Pollutant in a Pond System applet (figure 4) was modeled after a spreadsheet that Dr. Martin Schoonen of the Department of Geosciences at SUNY Stony Brook has used in some of his hydrogeology courses. It is designed to model a simple pond system with input and output streams. Parameters that can be specified by the user are:

- $C(s)$: concentration of a pollutant in the input stream
- $C(0)$: initial concentration of the pollutant in the pond
- V : volume of the pond
- Q : flow rate of the streams
- k : decay rate of the pollutant

Through text and charts, the applet outputs the change in the concentration of the pollutant in the pond over time.

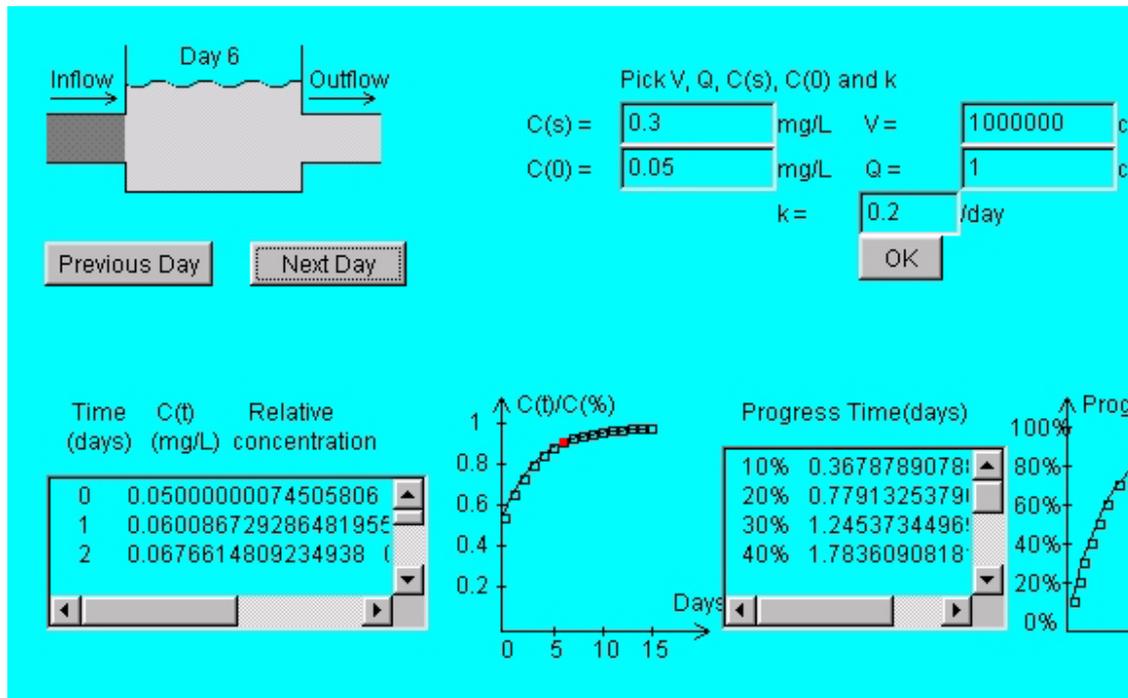


Figure 4. The Pollutant in a Pond System applet

The tr660 program (figure 5) models the mineralogy and physical properties for various chemical compositions in the Earth's mantle where a transition occurs in these properties at about 660 km depth.

The chemical system is modeled in terms of 5 oxides, MgO, FeO, CaO, SiO₂, and Al₂O₃. For most models of the Earth, these oxides constitute over 95% of the bulk composition. Variables XMG, XFE, XCA, XSI, XAL represent the number of the respective cation in the calculation. The appropriate number of oxygens are added automatically in order to maintain charge neutrality. The absolute value of these variables is not important, as they must simply be in the appropriate ratio that the user chooses. The default value is appropriate for the pyrolite model of the earth. All of these variables can be changed by the user to test alternate models.

The program calculates the stable phases at each pressure-temperature condition along an adiabatic temperature gradient, starting at 'P', 'T' for "Iterations" steps of 'DTAP' in pressure. At each condition, it then calculates the composition of each phase and it calculates the volume percents of each phase. Subsequently, it calculates the properties of the phases, and finally the properties of the bulk mineralogy. Output is graphical and tabular. The graphing output is versatile, allowing the user to view any of the calculated variables as a function of depth. The output can be imported to a file in the user's platform for further analysis.

The tr660 program was originally written in Fortran 77 by Don Weidner of the Center for High Pressure Research. The tr660 Java code was created by Eric Nuzzi, a computer

science undergraduate student at SUNY Stony Brook, as part of Project Java under the supervision of Glenn Richard.

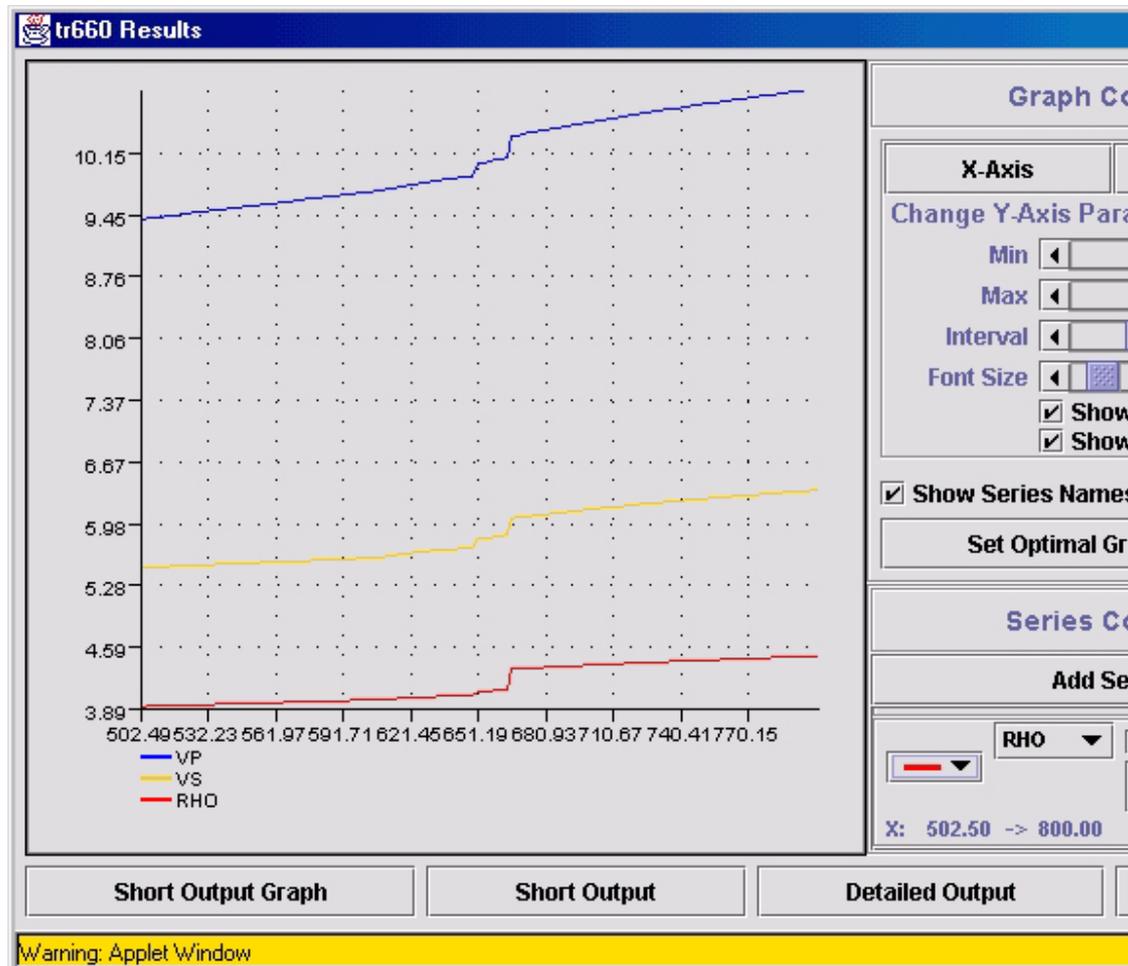


Figure 5. The tr660 applet

Since the summer of 1997, the Department of Computer Science and the Department of Chemistry have joined Geosciences in mentoring their own groups of Project Java computer science majors. Each year since 1997, Project Java groups or individuals have entered their software in the Long Island Software Awards (LISA) competition. In 1997, the three students who participated during the initial year of the project won first place in the undergraduate category as a group. In 1998, the award went to the Project Java group mentored by the Department of Computer Science, and in 1999, Eric Nuzzi won with the tr660 applet.

Project Java applets have also received recognition from sites devoted to Java, such as Gamelan and JARS. For instance, from April 18 to May 2, 1997, Radioactive Decay was Gamelan's featured applet. JARS rates Project Java as being among the top 5% of all Java Web sites.

During the spring, 2000 semester, ESERC, aided by a grant from the Center for Excellence and Innovation in Education (CEIE) on the Stony Brook campus, initiated a Project Java program for students at Brentwood High School. Kenneth Darling, through a Teaching Assistanceship established in the Department of Geosciences with the CEIE funds, is serving as an instructor for the project. About two dozen students and several Brentwood faculty are participating in the project, and meet with the instructor as a group for three hours each week.

GEO 511: Computer Programming for the Geosciences

[GEO 511](#) is designed to provide graduate students with the opportunity to learn how to create interactive graphically-oriented software that supports research, instrumentation, mathematical modeling, or educational presentations in their chosen area of interest. It is assumed that students entering the course know little or nothing about programming. Accordingly, emphasis is placed initially on having students learn how to program in Java. Subsequently, each student works on an individual programming project, with the focus of the course in the latter part of the semester placed on implementing and testing this software. Since the course is designed for geoscience graduate students, participants are encouraged to draw upon their primary areas of interest within the geosciences as a basis for their projects. Meetings consist of a combination of lecture, discussion, computer-based hands-on activities, and student project demonstrations.

Five graduate students participated in this course during its initial offering in the spring of 1999. In order to jump-start the software development process, the instructor, Glenn Richard, provided each participant with a starter applet after the project topics were selected. One of the students, Li Li, developed a program designed to analyze images of materials under pressure in order to calculate strain rates (figures 6 and 7). The first set of images she used represented magnesium oxide, which contained thin sheets of gold foil as reference features. The gold foil appears as diffuse vertical dark lines in the images. The mouse is used to select areas for the software to compare within the images. After the areas are chosen, the software uses various mathematical strategies to compute the locations of the lines and their displacements within a series of images. Figure 6 shows two images taken one minute apart. Figure 7 shows output computed using what we called an inner difference method, which computes the displacement that results in the minimum sum for the differences in gray scale values for corresponding pixels when the two images are compared. Least squares and other schemes were also used, eventually resulting in resolution on the scale of a fraction of a pixel.

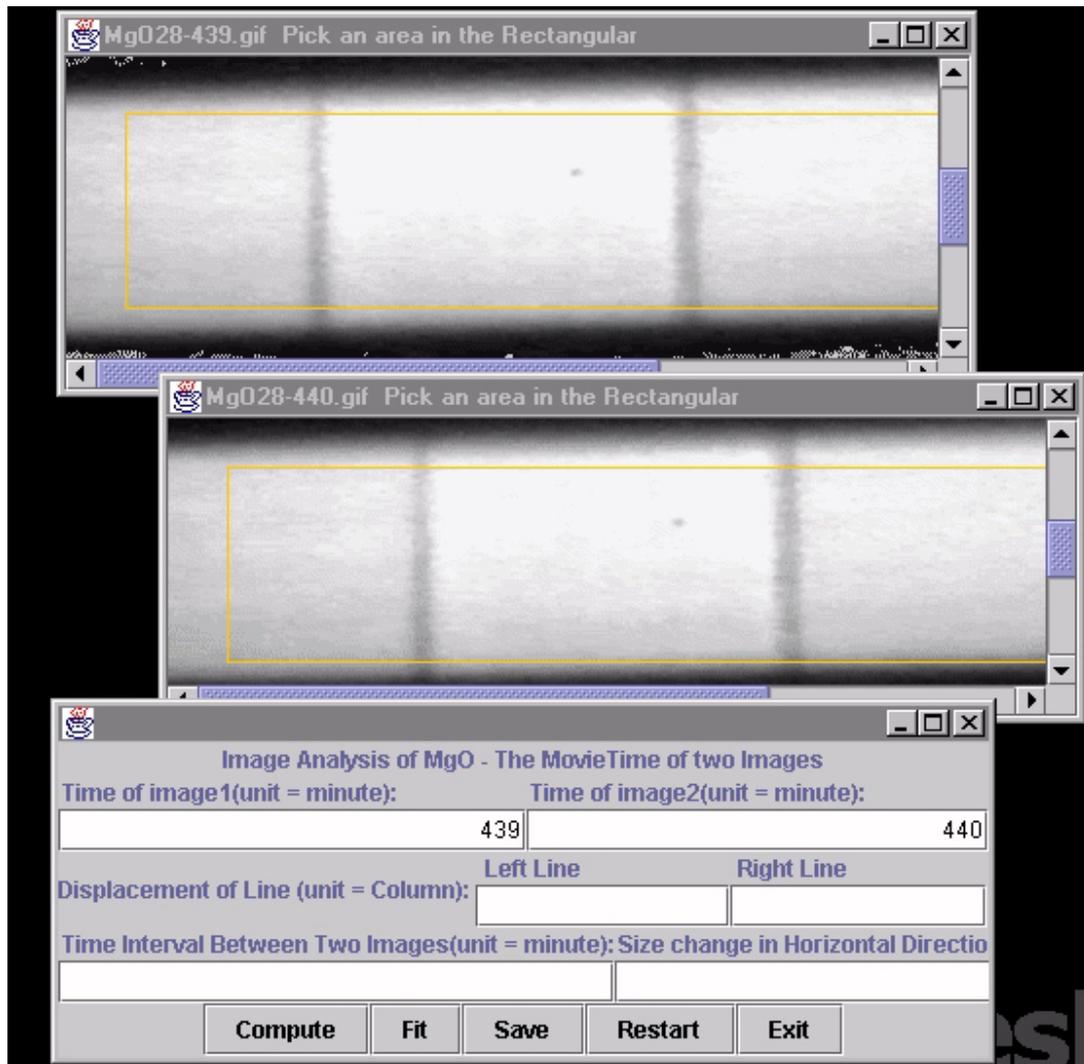


Figure 6. The MgOAnalyzer application showing images of MgO and gold foil

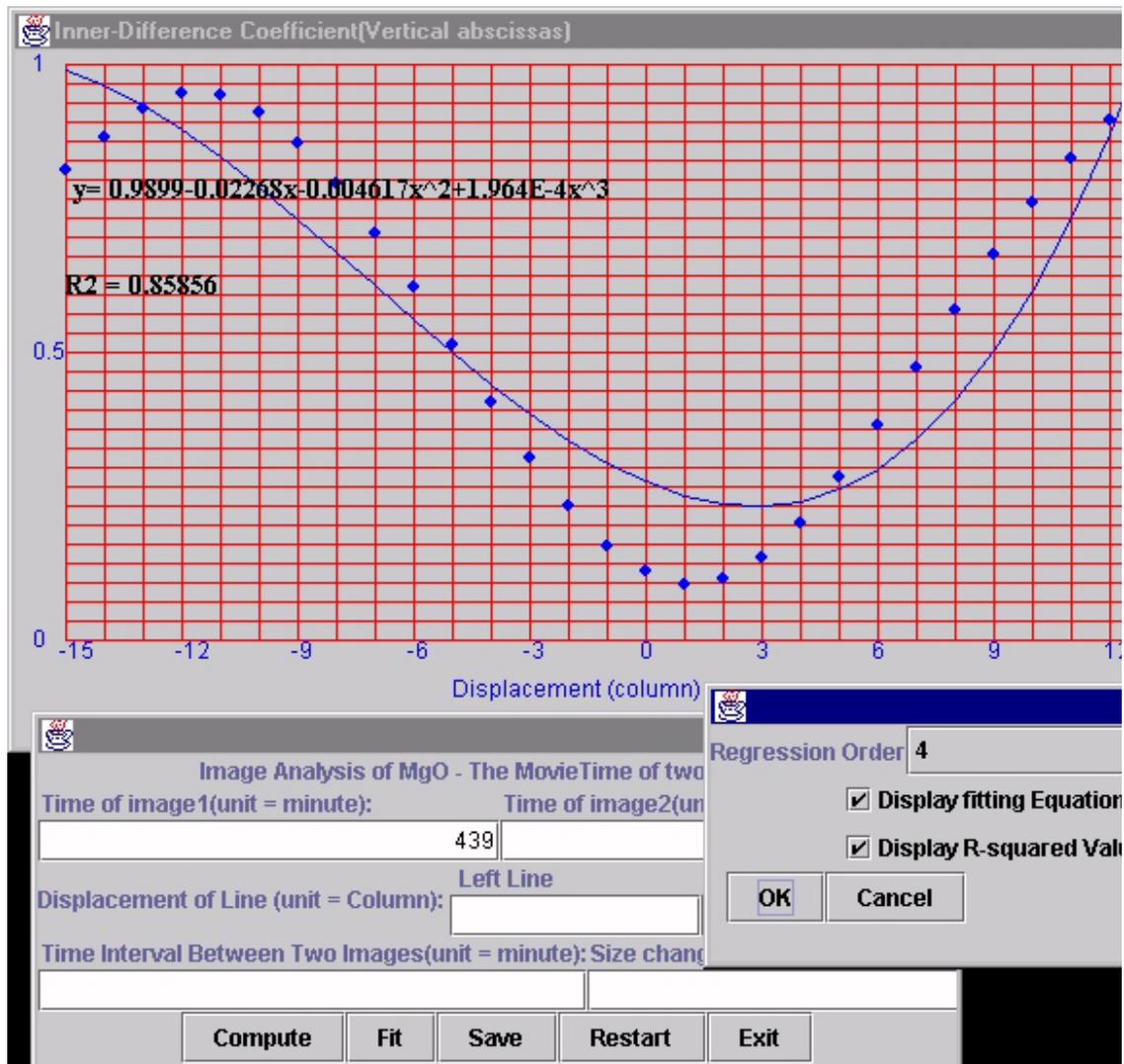


Figure 7. The MgOAnalyzer application output windows

GEO 327: Computerized Modeling of Geological Phenomena

[GEO 327](#) provides students with practical experience in creating computer software that implements mathematical models of selected geological phenomena. Each semester that the course is offered, a particular phenomenon is selected for study, and the students are provided with lectures and engage in discussions and homework exercises designed to familiarize them with the details of a mathematical model that describes the phenomenon. Subsequently, the participants are given an introduction to programming in which the selected phenomenon is used as a source of examples. The class then develops specifications for a software user interface, and an outline for the design of a software model. As a team, the class develops, tests, and refines the software, with each student writing a particular portion of the software according to the specifications. This course requires a significant amount of computer work outside of class time.

During the spring, 2000 semester, participants in GEO 327 are using Java to implement an interactive graphically-oriented mathematical model of a geyser. Three undergraduate students from the Department of Geosciences are enrolled. In addition, Philip Nachreiner, a Project Java participant, is serving as the project leader of the software development team that consists of the students in the course. The Java Computing Laboratory is being used as a classroom and work area for the course. Ultimately, the software developed during this semester will become part of a display at Yellowstone National Park that will be used by millions of visitors each year. It will also be posted on the World Wide Web in order to provide access to educators.

Portability of Java-Based Software Development Projects

It is not difficult to establish a software development project similar to Project Java, GEO 511, or GEO 327, provided that at least one person who is knowledgeable in Java is prepared to conduct the program, and that there is a source of students available for participation. The Java 2 SDK version 1.2 is free as a download and easy to install on systems running Solaris, Linux, or Windows. Macintosh systems can be used as well, but as of this writing, version 1.1 is the most recent release available for that platform. The first step in establishing a mentored software development program is to speak at length with a person with experience in conducting this type of project. This type of activity has tremendous potential for creating software for education and research, and for providing students with a working knowledge of programming and an opportunity for analytical creativity of a very intricate nature.

Other Science Education Java Websites

[Dip and Strike by VisualEntities](#)

Understand the concepts of dip and strike with an easy to use Java applet.

[Contour Map by VisualEntities](#)

Try your hand at drawing an elevation contour map.

[GeoAstro Applet Collection](#)

A collection of 12 Java applets concerning various aspects of the system sun-moon-earth: day and night, rise and set of the sun and the moon, horizon view (elevation, azimuth), daylight hours, hour angle, equation of time, and additional details.

[Stig's Sky Calendar](#)

The Sky Calendar displays sunrise/set, moonrise/set and twilight data for any location on earth using a graphical display. On top of this display, astronomy and space related events are displayed using small icons. Moving the mouse over an event will bring up information on the event. Clicking on an event will take you to a Web site containing more information. This applet was tested and found to run properly on Netscape 4.03 (and higher) and Internet Explorer 3.02 (and higher) on the Windows 95 platform.

[Periodic Table by VisualEntities](#)

Learn about the elements with an easy to use Java applet. Click on an element for detailed information.

[Biomer](#)

Biomer is a free molecular modeling program that can run as both an applet and an application. Version 1.0 alpha has the following features: model builders for polynucleotides (DNA/RNA), polysaccharides and proteins, interactive molecule editor, AMBER force-field based geometry optimization, simulated annealing with molecular dynamics, and the ability to save gif, jpeg, and ppm images.

[General Physics Java Applets](#)

Applets cover some basic math required in physics, kinematics, gravitation. Great applets covering Kepler's Laws of Motion are included.

[Crystallography applets](#)

This website contains more than 20 (including some eye-catching) Java applets related to crystallography, physics, chemistry and mathematics.

[Matters of Gravity](#)

Matters of Gravity is an electronic orrery, a model of a system of planets that orbit based upon the physical principles of gravity. This is an interactive model in which you can define your own planets, move them around, set their speeds, and even fly a little rocket ship around.

[Flight of a Projectile](#)

This Java applet demonstrates the classic kinematics problem of the flight of a cannonball shot over flat ground. While there is no pretty cannon, it does use correct units and gives a clear picture of trajectories. There are options to show trails of the projectile or to turn on a simple air friction. You have control over the initial velocity, initial angle and mass.

References

Bork, A. (1993). Learning Scientific Reasoning with the Interactive Computer. *Journal of Science Education and Technology*, Vol. 2, No. 1: 335-348.

Deitel, H.M. and Deitel, P.J. 1999. *Java How to Program*. Prentice Hall, Upper Saddle River, NJ.

Gamelan. <http://www.gamelan.com>

Horstmann, C.S. and Cornell, G. 1999. *Core Java Volume I - Fundamentals*. Sun Microsystems Press/Prentice Hall, Upper Saddle River, NJ.

JARS. <http://www.jars.com>

Java Boutique. <http://www.javaboutique.com>

Richard, G.A. 1997. The World Wide Web as a Resource for Earth Science Education. *Geology of Long Island and Metropolitan New York, Program with Abstracts*, April 19, 1997, p. 88 - 110. Long Island Geologists, Stony Brook, NY.