Title
NGP as a tool for creating Object-Oriented, 2D graphical applications in Java

Contact Information
Andries van Dam, Brown University (avd@cs.brown.edu)

Problem Statement
We are teaching an introductory course on Object-Oriented design and programming in Java. We emphasize interactive graphical programs, but felt that Java's AWT and Swing frameworks are too complex for beginning students, and the Java APIs don't always adhere to the strict Object-Orientation we try to foster.

Solution Overview
NGP is a simple graphics package of approximately 50 classes and interfaces, which contains GUI components and 2-D graphical objects. It hides the complex AWT/Swing painting hierarchy and event handling mechanisms from the student, allowing graphical components to be treated as self-contained objects.

For example, a student can draw an ellipse in a panel by simply creating an instance of `FilledEllipse` and associating it with a `DrawingPanel`. The ellipse's properties can be manipulated through methods like `setLocation`, `setDimension`, `setColor`, etc. The `DrawingPanel` maintains a collection of the objects that have been added to it and takes care of repainting them with the appropriate calls to an instance of `java.awt.Graphics`. The `FilledEllipse` also encapsulates mouse interaction by providing methods like `mouseClicked`, which can be overridden in a subclass to add functionality.

The encapsulation of complex painting and event handling hierarchies makes for a gentler introduction to graphical programming than the standard Java libraries. In addition, NGP encourages good Object-Oriented programming principles and code reuse by incorporating visual properties and event handling into complete objects that can be extended and customized.

Experience with the Solution
NGP has been in use at Brown since 1999, and it has been adopted by at least one other institution for introductory programming classes. Beginning next year, we will actually be abandoning NGP in favor of a new library, OOPS, in the interest of providing more comprehensive exposure to Swing. Also developed at Brown, OOPS is a minimal library that aims to ease introductory students into the complexity of Swing without completely hiding it (see Java Task Force submission "OOPS as a tool for teaching Object-oriented graphics in Java"). The primary reason for making the change from NGP to Swing is not dissatisfaction with NGP but that we'll be using a new textbook based on this course (by Sanders and van Dam), which, for purely commercial reasons, uses Swing rather than the NGP "homebrew" library. Nevertheless, NGP remains a useful tool for classes seeking a more feature-rich alternative to Swing/AWT if ultimate proficiency in the native Java

libraries is not required. At Brown alone, approximately 700 students have taught with NGP.

Using NGP, students are able to begin creating simple graphical programs with a basic understanding of parameters. Within 2 weeks of being introduced to the library, students create a fairly sophisticated GUI, using a variety of components like sliders, push buttons, radio buttons, and text boxes to manipulate graphical objects. By mid-semester, students can create advanced graphical applications, incorporating images, sound, keyboard interaction, and animation.

API Documentation

Full documentation for NGP can be viewed at http://www.cs.brown.edu/courses/cs015/2003/docs/NGP/. Code for the library can be downloaded from http://www.cs.brown.edu/courses/cs015/2003/lib/NGP.zip.