KlassroomSwing – A Stepping Stone To javax.swing

Contact Information
Dean Sanders
Northwest Missouri State University
sanders@mail.nwmissouri.edu

Problem Statement
Computer science educators have always sought the best way to teach programming. Current discussions include consideration of how early object-oriented techniques should be introduced and how early students should be expected to create graphical user interfaces (GUIs). This appears to be a situation where no one is wrong, but everyone needs to make a choice. Regardless of which approach is chosen, two things are clear. First, the students live in a world of multimedia and graphical user interfaces, but many textbooks, examples, and ancillary materials are rooted in text-based interactions. Second, support materials are important, regardless of the approach that is used. KlassroomSwing was designed as a support tool for those that use an objects-first, GUI-early approach to teaching Java.

KlassroomSwing is a Java package that allows beginning programmers to construct GUIs and to incorporate media into their programs. KlassroomSwing grew out of frustration with existing packages. Various packages provide some of this capability, but none are wholly suitable. Some (e.g. Swing, Java Power Tools) have steep learning curves, some (e.g. BreezySwing) differ from Swing in fundamental ways, and many are too limited in scope. KlassroomSwing was designed to rectify many of these problems.

Solution Overview
KlassroomSwing is a self-contained package that parallels a subset of Java’s Swing. Most of the classes and methods are mirror images of those in Swing, but seldom-used methods have been eliminated, some things have been streamlined, and a few new classes and methods have been added. Rather than extending Swing classes, KlassroomSwing makes extensive use of delegation and the façade pattern; thereby simplifying the API and significantly reducing the clutter in the API documentation.

Several design goals guided the development of KlassroomSwing. Understanding the design goals gives insight into the package as a whole. KlassroomSwing should (1) be simple and to the point, (2) be easy to learn, (3) be uncluttered with extraneous (for our purposes) methods, (4) provide support for common widgets, (5) provide support for images, audio, and video, (6) provide support for graphics, (7) be completely self-contained, (8) mirror Java’s Swing as much as practical, (9) incorporate additional classes and methods to simplify common tasks, and (10) borrow good ideas from related packages.

As of this writing, KlassroomSwing can be partitioned into five sets of features: (1) containers, (2) analogues to Swing components, (3) special components, (4) media support, and (5) utility classes. Additional features are planned for development.

Two classes play the role of containers, KSFrame and KSPanel. The KSFrame has a default size, built-in window closing logic, and specific methods to set the background color and to center the frame on the screen. One adds KSComponents to a KSFrame.
directly, rather than to its content pane. The default layout manager for KSFrame and
KSPanel is a GridBagLayout with a hidden GridBagConstraints.

KlassroomSwing contains analogues to most of Swing’s user interface components.
These are simplified in that they only support a subset of the Swing methods; seldom
used methods have been eliminated. Whenever possible the prototypes for
KlassroomSwing methods are identical to those for the Swing methods. Some
KlassroomSwing components contain additional methods that are not in their Swing
counterparts. For example, there are straightforward methods to add borders to a
component.

KlassroomSwing has a few special components to provide capabilities that are
harder to implement in Swing. The KSDataField is an enhanced JTextField with
methods to get and set text, int values, or double values. It’s also possible to specify a
label as part of a KSDataField. The KSControl component provides a synchronized
menu item and toolbar button. The KSDoubleSpinner and KSIntSpinner classes provide
standard spinner components (thumbwheels). Neither is directly available in Swing.
KlassroomSwing has straightforward support for images and audio clips. An image
can be in GIF, JPEG, or PNG format. An audio file can be in AU, WAV, or AIF format.
KlassroomSwing has some classes that simplify common tasks. Classes
KSCalendarDate and KSClockTime provide easier ways to work with and format dates
and times, respectively. Class KSDecimalFormat provides methods for formatting
numbers as integers, fixed point decimals, floating point decimals, and percentages.

Additional features are planned for summer 2004. These include mouse and keyboard
adapters, dialog windows, and support for playing video clips.

Experience with the Solution
KlassroomSwing has been used since August 2003 at Northwest Missouri State University.
KlassroomSwing is being used in two Java-based courses that are comparable to CS101o and
CS102o in Computing Curriculum 2001. As of this writing, KlassroomSwing has been used by
five instructors and approximately 120 students.

We have sought no empirical evidence regarding the effectiveness of KlassroomSwing, but
the anecdotal evidence is encouraging. Comments from faculty members have been very
positive. Upper-division students who work as tutors and teaching assistants have been envious,
wishing that KlassroomSwing had been available when they were freshmen. Those who have
taught the courses have observed that novices can use KlassroomSwing with little difficulty, and
that students do not outgrow KlassroomSwing. Perhaps the most important observation is that
students learn to use KlassroomSwing’s simplified API documentation effectively. Next fall we
will learn whether or not KlassroomSwing really provides a good entry point for learning Swing.

API Documentation
The current API classes and documentation are available at
www.nwmissouri.edu/~sanders/KlassroomSwing/KS.html.

Supplemental Material
Preliminary teaching materials are will be available by August 2004.