

Jeroo – A Tool for Introducing Object-Oriented Concepts

Contact Information

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

Brian Dorn
Iowa State University
dorn@cs.iastate.edu

Problem Statement

The first programming course has always been difficult to teach. Whether we are teaching procedural or object-oriented programming, the first few weeks can be intimidating as students work to master abstract concepts and a myriad of details. This is compounded by textbooks, examples, and ancillary materials that are rooted in text-based applications, even though the students live in a highly visual computing environment. Many lose interest and self-confidence early in their first programming course. The desire for a teaching tool that would address these concerns led us to develop Jeroo. Our goal was to create a microworld and programming language that would (1) be immediately understandable by novices, (2) engage all students, (3) be rich enough for sophisticated problems, (4) feature animated execution with code highlighting, (5) focus on objects, methods, and control structures, (6) support a smooth transition to Java, and (7) run on a variety of platforms with minimal resources.

Solution Overview

Jeroo is an integrated development environment and microworld that is reminiscent of Karel the Robot and its descendants. A Jeroo is a kangaroo-like animal living on a remote island in the South Pacific. A Jeroo can pick flowers, plant them, or give them to an adjacent Jeroo. Jeroos move by hopping in the four main compass directions. As they hop, they must avoid water, nets, and each other. They can toss flowers to disable the nets. Students immediately relate to Jeroo's friendly anthropomorphic nature and its non-technical, non-violent metaphor. Students write programs to control up to four Jeroos and their interactions with their environment.

The Jeroo application has four major components: the user interface, the Jeroo programming language, editors, and a runtime module. The user interface is a single window in which all components are visible at all times. Menus and a toolbar provide a familiar look and feel. The left-hand side of the screen contains tabbed panes for editing source code. The right-hand side contains a representation of the island. The lower part of the screen is used to display status information. One status area continually monitors the state of every Jeroo in a running program. A second status area reports the state of a program and displays error messages. Syntax and runtime errors highlight the offending line and generate a carefully worded message. A typical message has the form "SYNTAX ERROR: Expected XXX but found YYY."

The syntax of the Jeroo language is identical to Java, save for a generic method header. This helps smooth the transition from Jeroo to Java. The language has only one class – the Jeroo class. This class has six constructors that provide a gentle introduction to overloading and the concept of state. The language has six action methods to control the Jeroo's behavior and six sensor methods through which the Jeroos examine their immediate surroundings. Students can write additional methods, even recursive ones, to extend the Jeroos' behavior. The language supports three control structures: while, if, and if-else as well as the Boolean operators !, &&, and ||. The language is small enough to be mastered by novices, but rich enough for sophisticated problems.

Jeroo includes a runtime module that uses animation and code highlighting to illustrate the connection between source code and runtime behavior. A program may be executed stepwise or continuously at five different speeds. Students can switch between modes and change speeds during execution. The combination of source code highlighting, animation, and status information creates a rich learning environment. Both syntax and runtime errors can be located quickly; the semantics of the control structures are readily apparent; and the interaction between methods is revealed.

The Jeroo tool provides a straightforward way to edit both the source code for a Jeroo program and the layout of the island. The source code editor supports all common operations as well as block indent/unindent, block comment/uncomment, and optional auto-indent. The island editor employs a point-and-click process to place and remove flowers, nets and water features.

The current version is 2.1.2. Version 2.2 will be released in the summer of 2004. That release will have two additional features, the ability to change the editor's font, and the ability to remember login information. No enhancements are anticipated beyond version 2.2.

Experience with the Solution

Jeroo is being used in high schools, colleges, and universities. It is used in computer literacy courses, pre-CS1 courses, and CS1-type courses. Written in Java, Jeroo has been used on Linux, Solaris, Mac OS-X, and various Windows platforms. There are some international users.

Jeroo has been used since August 2002 at Northwest Missouri State University (NWMSU). Jeroo is used in a Java-based course that is comparable to CS101o in Computing Curriculum 2001. As of this writing, Jeroo has been used by five instructors and approximately 200 students. Typical student comments are "Jeroo is fun" and "Everything goes back to Jeroo."

The University of Wisconsin – Parkside offers a one-credit-hour, pre-CS1 course for those who are not yet ready for the Java-based CS1 course. Jeroo was adopted for that course in the Fall semester of 2003. Since switching to Jeroo, the withdrawal rate for the pre-CS1 course has fallen dramatically, and the percentage of students who continue to CS1 has increased. The instructor for that course has reported a significant increase in the student's enthusiasm.

Informal reports have been very positive. Typical comments are that students love Jeroo, that it's an effective teaching tool, and that it's much better than Karel. Several perceived benefits have been reported including: (1) Jeroo engages students immediately, (2) students master important topics faster than they do without Jeroo, (3) Jeroo encourages experimentation.

Objective data from NWMSU shows that the students report statistically significant increases in confidence and comfort in the course after using Jeroo; the increase is greater among female students; and after using Jeroo, there is no significant difference between females and males. Jeroo levels the playing field with respect to these measures.

API Documentation

Since Jeroo is a stand-alone tool, API documentation is not appropriate. However, Jeroo's help menu includes a language summary. Documentation, in the form of a rudimentary book, is available at www.Jeroo.org.

Supplemental Material

The software, documentation, and copies of related publications are available at www.Jeroo.org.