

Tricks

of the

Java Programming Gurus

by *Glenn L. Vanderburg. et al.*

C O N T E N T S

[Introduction](#)

Chapter 1 [Communication Between Applets](#)

- [getApplet: The "Official" Mechanism](#)
- [Static Variables and Methods](#)
- [Network Communication](#)
- [Thread-Based Communication](#)
- [Summary](#)

Chapter 2 [Using the Media Tracker](#)

- [Java Media Objects and the Internet](#)
- [Keeping Up with Media Objects](#)
- [The MediaTracker Class](#)
- [Tracking Images with the Media Tracker](#)
- [Tracking Other Types of Media](#)
- [Summary](#)

Chapter 3 [Exploiting the Network](#)

- [Retrieving Data Using URLs](#)
- [Posting Data to a URL](#)
- [Communication Using Sockets](#)
- [Summary](#)

Chapter 4 [Using Java's Audio Classes](#)

- [Digital Audio Fundamentals](#)
- [Java Audio Support](#)
- [Playing Audio In Java](#)

The Future of Java Audio

- [Summary](#)

Chapter 5 [Building Special-Purpose I/O Classes](#)

- [Stream Classes](#)
- [Non-Stream I/O Classes](#)
- [Highly Structured Files](#)
- [Summary](#)

Chapter 6 [Effective Use of Threads](#)

- [Using Threads](#)
- [Performance](#)
- [Inside Threads](#)
- [Summary](#)

Chapter 7 [Concurrency and Synchronization](#)

- [Concurrency](#)
- [Monitors](#)
- [Advanced Monitor Concepts](#)
- [Synchronization](#)
- [A Thread Coordination Example](#)
- [Advanced Thread Coordination](#)
- [Summary](#)

Chapter 8 [All About GridBagLayout and Other Layout managers](#)

- [Automated Layout and the AWT Layout Manager](#)
- [Basic Layout Classes](#)
- [The GridBagLayout Class](#)
- [Creating Your Own Layout Manager](#)
- [Summary](#)

Chapter 9 [Extending AWT Components](#)

- [Components-an Overview](#)
- [New Components from Old](#)
- [A Self-Validating TextField](#)
- [A Multi-State Toggle Button](#)
- [Overview](#)
- [Summary](#)

Chapter 10 [Combining AWT Components](#)

- [Component, Container, Panel](#)
- [E Pluribus Unum: Out of Many-One](#)
- [Panels Are Components Too](#)
- [Layouts](#)

- [Whose Event Is It Anyway?](#)
- [The Panel as a Component Manager](#)
- [A Scrolling Picture Window Example](#)
- [Overview](#)
- [Class Construction](#)
- [Event Handling](#)
- [Summary](#)

Chapter 11 [Advanced Event Handling](#)

- [Basic Event Handling](#)
- [The Event Class](#)
- [Key Events](#)
- [Mouse Events](#)
- [Displaying Events](#)
- [Events with Methods](#)
- [Generating Events](#)
- [Fixing Broken Event Handling](#)
- [A Complete Example](#)
- [Major Surgery to the Event Model](#)
- [Summary](#)

Chapter 12 [Image Filters and Color Models](#)

- [Understanding Color](#)
- [Color Images in Java](#)
- [Color Models](#)
- [The Color Model Classes](#)
- [Working with Color Models](#)
- [Image Filters](#)
- [The Image Filter Classes](#)
- [Writing Your Own Image Filters](#)
- [Using Image Filters](#)
- [Summary](#)

Chapter 13 [Animation Techniques](#)

- [What Is Animation?](#)
- [Types of Animation](#)
- [Implementing Frame Animation](#)
- [Eliminating Flicker](#)
- [Implementing Sprite Animation](#)
- [Testing the Sprite Classes](#)
- [Summary](#)

Chapter 14 [Writing 2D Games](#)

- [2D Game Basics](#)
- [Scaling an Object](#)
- [Translating an Object](#)
- [Rotating an Object](#)

- [2D Game Engine](#)
- [The Missile Class](#)
- [Asteroids](#)
- [The Asteroids Applet Class](#)
- [The Asteroids](#)
- [The Ship](#)
- [The Photons](#)
- [Final Details](#)
- [Summary](#)

Chapter 15 [A Virtual Java-Creating Behaviors in VRML 2.0](#)

- [Going Beyond Reality](#)
- [Making the World Behave](#)
- [Overview of VRML](#)
- [The VRML Script Node](#)
- [VRML Datatypes in Java](#)
- [Integrating Java Scripts with VRML](#)
- [The Browser Class](#)
- [The Script Execution Model](#)
- [Creating Efficient Behaviors](#)
- [Dynamic Worlds-Creating VRML on the Fly](#)
- [Creating Reusable Behaviors](#)
- [The Future: VRML, Java, and AI](#)
- [Summary](#)

Chapter 16 [Building Stand-Alone Applications](#)

- [Writing and Running a Java Program](#)
- [Properties](#)
- [Application Instances](#)
- [The BloatFinder Application](#)
- [Using Java's Special Features](#)
- [Summary](#)

Chapter 17 [Network-Extensible Applications with Factory Objects](#)

- [How Factories Work](#)
- [Factory Support in the Java Library](#)
- [Factory Object Implementation Considerations](#)
- [Supporting a New Kind of Factory](#)
- [Security Considerations](#)
- [Summary](#)

Chapter 18 [Developing Database Applications and Applets](#)

- [Storing Data for the Web](#)
- [Providing Access to Data](#)
- [The JDBC API](#)
- [Simple Database Access Using the JDBC Interfaces](#)
- [Result Sets and the Meta-Data Interfaces](#)

- [Other JDBC Functionality](#)
- [Building a JDBC Implementation](#)
- [Extending JDBC](#)
- [Designing a Database Application](#)
- [Summary](#)

Chapter 19 [Persistence](#)

- [What Is Persistence?](#)
- [Forms of Persistence \(in Java\)](#)
- [Implementing a Simple File-Based Persistent Store](#)
- [The PersistentJava \(PJava\) Project](#)
- [Summary](#)

Chapter 20 [A User's View of Security](#)

- [Users Need to Understand](#)
- [The Kinds of Attacks](#)
- [Which Resources Are Dangerous?](#)
- [Cultural Change](#)
- [Summary](#)

Chapter 21 [Creating a Security Policy](#)

- [The Java Security Model](#)
- [The Java Security Manager](#)
- [Security Manager Decisions](#)
- [Which Resources Are Protected?](#)
- [Understanding Security Risks](#)
- [Keeping the Security Policy Manageable](#)
- [Implementing Class Loaders](#)
- [Implementing Security Managers](#)
- [Summary](#)

Chapter 22 [Authentication, Encryption, and Trusted Applets](#)

- [Cryptography Basics](#)
- [Security Mechanisms Provided by java.security](#)
- [Enabling Trusted Applets](#)
- [Cryptographic Security Solves Everything, Right?](#)
- [Summary](#)

Chapter 23 [Pushing the Limits of Java Security](#)

- [Introducing Hostile Applets](#)
- [Challenges for the Hacker](#)
- [A Very Noisy Bear](#)
- [A Gluttonous Trio](#)
- [Throw Open a Window](#)
- [Survival of the Fittest, Applet Style](#)

[Port 25, Where Are You?](#)

- [A Java Factoring-By-Web Project](#)
- [Summary](#)

Chapter 24 [Integrated Development Environments](#)

- [The Examples Used in This Chapter](#)
- [Symantec's Cafe Lite](#)
- [ED for Windows, The Java IDE](#)
- [Object Engineering Workbench](#)
- [Comparison of Environments](#)
- [Other Products Under Development](#)
- [Summary](#)

Chapter 25 [Class Organization and Documentation](#)

- [Java Packages](#)
- [Documentation Generation Using javadoc](#)
- [Class Disassembly Using javap](#)
- [Summary](#)

Chapter 26 [The Java Debugging API](#)

- [Remote Debugging](#)
- [Java Debugger](#)
- [The Sun Java Debugger API](#)
- [Simple Types](#)
- [Some Examples](#)
- [Summary](#)

Chapter 27 [Alternatives to Java](#)

- [Nuts and Bolts Languages](#)
- [General-Purpose Languages](#)
- [Scripting Languages](#)
- [Secure Languages](#)
- [Summary](#)

Chapter 28 [Moving C and C++ Code to Java](#)

- [File Organization](#)
- [The Preprocessor](#)
- [Structures and Unions](#)
- [Functions and Methods](#)
- [Procedural-to-OOP Conversion](#)
- [Operator Overloading](#)
- [Automatic Coercions](#)
- [Command-Line Arguments](#)
- [I/O Streams](#)
- [Strings](#)

[Pointers](#)

- [Multiple Inheritance](#)
- [Inheritance Syntax](#)
- [Access Modifiers](#)
- [Friends and Packages](#)
- [Booleans](#)
- [Summary](#)

Chapter 29 [Using Tcl with Java](#)

- [Introduction to Tcl](#)
- [What Does This Have to Do with Java?](#)
- [The TclJava API](#)
- [User Menu Configuration](#)
- [Other Useful Roles for Tcl](#)
- [Tcl Extension Packages](#)
- [Summary](#)

Chapter 30 [When and Why to Use Native Methods](#)

- [What Is a Native Method?](#)
- [Uses for Native Methods](#)
- [Benefits and Trade-Offs](#)
- [How Does This Magic Work?](#)
- [Summary](#)

Chapter 31 [The Native Method Interface](#)

- [A Java Class with Native Methods](#)
- [Accepting and Returning Java Classes](#)
- [Accessing Arrays of Classes](#)
- [Accessing a Float Array](#)
- [Summary](#)

Chapter 32 [Interfacing to Existing C and C++ Libraries](#)

- [Interfacing to Legacy C Libraries](#)
- [Developing Java Interface Class Libraries with Legacy C++ Libraries](#)
- [Special Concerns and Tips](#)
- [Summary](#)

Chapter 33 [Securing Your Native Method Libraries](#)

- [Security in Native Method Libraries](#)
- [Avoiding the Problem](#)
- [Identifying Security-Sensitive Resources](#)
- [Security Checks](#)
- [Language Protection Mechanisms](#)
- [Summary](#)

Chapter 34 [Client/Server Programming](#)

- [Java's Suitability for Client/Server Programming](#)
- [Client and Servers](#)
- [Merging the Client and Server](#)
- [Java's Deployable Code Advantage](#)
- [Java TCP/IP Sockets](#)
- [Using Datagram for UDP Sockets](#)
- [Using Socket and ServerSocket for TCP Sockets](#)
- [Summary](#)

Chapter 35 [Taking Advantage of the Internet in Development](#)

- [Ending the Isolation of the pc](#)
- [Working and Playing in Groups](#)
- [Creating and Using Resource Libraries](#)
- [Distributing and Maintaining Software](#)
- [Creating Alternative Revenue Schemes](#)
- [Summary](#)

Chapter 36 [Fulfilling the Promise of Java](#)

- [The Java Ideals](#)
- [The Java Community](#)
- [Setting Your Sights](#)

Appendix A [API Quick Reference](#)

- [java.applet](#)
- [java.awt](#)
- [java.awt.image](#)
- [java.awt.peer](#)
- [java.io](#)
- [java.lang](#)
- [java.net](#)
- [java.util](#)

Appendix B [Class Hierarchy Diagrams](#)

[Credits](#)

Copyright © 1996 by Sams.net Publishing

FIRST EDITION