



Black Art of Java Game Programming

by Joel Fan

Sams, Macmillan Computer Publishing

ISBN: 1571690433 Pub Date: 11/01/96

[Table of Contents](#)

Introduction

Gaming on the Web is the next blockbuster business. Have any doubts? The video game industry, which already eclipses Hollywood in terms of revenue, targets the Web as the next huge growth area. Software developers are busily porting games to the online environment, or developing entirely new ones. And numerous popular Web sites, many of which charge fees, are devoted to game entertainment.

With *Black Art of Java Game Programming*, you'll learn how to create your own dynamic online entertainment using the Java programming language. Java heralds a revolutionary shift in gaming, away from the desktop and into the network. Using Java, you'll create games that people can download through the Web and play. Using Java, you'll enable players from around the world to compete against one another. As Java enters its maturity, the possibilities of creating innovative entertainment are endless, and the potential for profit unbounded.

This book provides you with the foundations for creating Java games of all types, such as board games, video games, and networked/multiplayer games. We recommend that you have some basic knowledge of C, C++, or Java before diving in. The underlying thread of this book is that good object-oriented design and good Java games go hand in hand, so we devote the early chapters of the book to covering and applying object-oriented principles to graphics and games. Once the foundations have been laid, we build a wide variety of games. As you will see, game programming is almost a game in itself, and once you learn the basics, you'll be well equipped to write innovative games on your own.

Organization

Black Art of Java Game Programming is divided into three parts:

Part I, Fundamentals of Java Game Development

Part II, Advanced Game and Graphics Techniques

Part III, Game Gallery

Part I, Fundamentals

Part I takes you on a whirlwind tour of Java, graphics, and object-oriented game programming. This section is for you if you're learning Java, if you're new to object-oriented design, or if you want to see how to build a game step by step.

This is what's covered in the Fundamentals section:

Chapter 1, "Fundamental Java," delivers a turbocharged introduction to the Java language, the API, and object-oriented design. By the end, you'll create graphics applets.

Chapter 2, "Using Objects for Animation," shows you how classes, inheritance, and dynamic method binding help you animate objects. You'll also learn about clipping and double-buffering, two basic graphics techniques.

Chapter 3, "Animating Sprites," teaches you to create a simple abstract class for representing graphics objects called sprites. In addition, you'll learn about interfaces, bitmap animation, and sound.

Chapter 4, "Adding Interactivity," shows you how to create applets that respond in real time to player input.

Chapter 5, "Building a Video Game," shows you how to apply what you've learned in the first four chapters to create a shoot-'em-up video game. What you learn here can be applied to creating many other types of games.

Chapter 6, "Extending Your Video Game," shows you how to take a game that you've developed and add new features to it without starting from scratch.

Chapter 7, "Creating Customizable Games with the AWT," demonstrates how Java's Abstract Windowing Toolkit allows players to change parameters in your games. What you learn here about the AWT will be applied throughout the rest of the book.

Part II, Advanced Game and Graphics Techniques

In Part II, you'll learn the skills necessary to bring your games into the next dimension, such as multithreading, networking and multiplayer techniques, and 3D.

Chapter 8, "Implementing a High Score Server on a Network," takes you through Java's networking and GUI facilities, and teaches you to build a high score server for your games.

Chapter 9, "Advanced Networking and Multiplayer Gaming Concepts," illustrates techniques for enabling multiuser game play over the Web. In addition, you'll deepen your understanding of Java's networking capabilities by implementing a chat room.

Chapter 10, “Advanced Techniques,” covers features of Java and the Java API that are useful in writing games and organizing programs.

Chapter 11, “Into the Third Dimension,” demonstrates the process of defining, transforming, projecting, and painting three-dimensional models, and builds classes that can be used to make a simple 3D engine.

Chapter 12, “Building 3D Applets with App3Dcore,” shows how the App3Dcore (a set of classes) works and how it can be used to develop some simple 3D applets and an advanced 3D game.

Part III, Game Gallery

In Part III, you’ll apply the skills you’ve learned in earlier chapters as leading Java game designers take you step by step through the creation of a wide spectrum of cool games.

Chapter 13, “Building the JAVArroids Game,” shows you how to create a Java version of the video game classic Asteroids.

Chapter 14, “Daleks!,” takes you through the creation of an enhanced Java version of a classic computer game.

Chapter 15, “NetOthello,” builds on your networking skills learned in earlier chapters to create a networked implementation of the classic game Othello.

Chapter 16, “WordQuest,” takes you through the creation of a Java game specifically designed to teach vocabulary, but which could easily be extended to teach a plethora of other concepts, demonstrating Java’s potential as a learning tool.

Chapter 17, “The Magic Squares Puzzle,” is an example of a deceptively simple, yet challenging puzzle game that will delight Rubik’s Cube enthusiasts (and many others).

Chapter 18, “The Internet MahJong Server,” demonstrates a software package that allows people to play the classic Chinese strategy game MahJong with each other online.

Chapter 19, “Slider Puzzle,” shows you how to write a Java applet for a simple slider puzzle enhanced with animation and sound.

Chapter 20, “The Game of Worm,” develops a game in which you control the direction of a virtual worm on a rectangular playing surface, collecting treats while avoiding collision with solid objects.

[Table of Contents](#)