CD-ROM INCLUDED

ADVANCED FLASH™ 5

ActionScript In Action

- Master Flash interactivity using ActionScript!
- Build complete Web apps, e-commerce front-ends, and games
- Use XML structured data in your Flash-based Web applications
- Learn hands-on through expert examples
- CD-ROM: Includes all Flash movies from the book

DAN LIVINGSTON

Advanced Flash 5 ActionScript in Action

江苏工业学院图书馆
藏书章

Dan Livingston

PH PTR

Prentice Hall PTR
Upper Saddle River, NJ 07458
www.phptr.com

ISBN 0-13-093127-6

90000

9 780130 931276

Library of Congress Cataloging-in-Publication Data

Livingston, Dan
Advanced Flash 5 ActionScript in Action / Dan Livingston.
p. cm. -- (Advanced Topics for Web Professionals)
Includes bibliographical references and index.
ISBN 0-13-093127-6
1. Computer animation. 2. Flash (Computer file) 3. Computer Graphics. 4. Web sites--Design. I. Title. II. Series.

TR897.7 .L58 2001
006.6'96--dc21

2001021125

Editorial/Production Supervision: *Jan H. Schwartz*
Acquisitions Editor: *Karen McLean*
Cover Design Director: *Jerry Votta*
Cover Design: *Design Source*
Manufacturing Manager: *Alexis R. Heydt*
Buyer: *Maura Zaldivar*
Marketing Manager: *Bryan Gambrel*
Editorial Assistant: *Rick Winkler*
Art Director: *Gail Cocker-Bogusz*
Series Interior Design: *Meg VanArsdale*

© 2001 Prentice Hall PTR
Prentice-Hall, Inc.
Upper Saddle River, NJ 07458

Prentice Hall books are widely used by corporations and government agencies for training, marketing, and resale.

The publisher offers discounts on this book when ordered in bulk quantities.
For more information, contact: Corporate Sales Department, Phone: 800-382-3419;
Fax: 201-236-7141; E-mail: corpsales@prenhall.com; or write: Prentice Hall PTR,
Corp. Sales Dept., One Lake Street, Upper Saddle River, NJ 07458.

All rights reserved. No part of this book may be
reproduced, in any form or by any means, without
permission in writing from the publisher.

All product names mentioned herein are trademarks or registered trademarks of their respective owners.

Printed in the United States of America

10 9 8 7 6 5 4 3 2

ISBN 0-13-093127-6

Prentice-Hall International (UK) Limited, *London*
Prentice-Hall of Australia Pty. Limited, *Sydney*
Prentice-Hall Canada Inc., *Toronto*
Prentice-Hall Hispanoamericana, S.A., *Mexico*
Prentice-Hall of India Private Limited, *New Delhi*
Prentice-Hall of Japan, Inc., *Tokyo*
Prentice Hall Asia Pte., Ltd.
Editora Prentice-Hall do Brasil, Ltda., *Rio de Janeiro*

Foreword

I've written a few books on some pretty fun topics (PhotoShop, DHTML, and JavaScript), but I had a great time writing this book—Flash 5 is a hoot. Flash 5 can do some pretty amazing things, and I hope I've been able to show you some of these things well enough to get you excited too.

The Goal of This Book

When you're through with this book, you should have a thorough grasp of what Flash 5 and ActionScript can do, and know how to create a wide diversity of really engaging, fun, and useful interactive Flash movie. I'm talking fantastic navigation, awesome games, and efficient, robust applications.

Hopefully, given the foundation you'll get in this book, you'll be able to create new kinds of Flash movies that no one else has thought of.

The focus of this book is more technical than artistic, but I believe it provides a good springboard for programmers, animators, and artists alike.

Who This Book Is For

This book is for anyone who wants to include any level of complexity and interaction in their Flash movies. This can include navigation, games, and interaction with middleware and databases.

You can be a right-brained Flash designer who's making the reluctant leap into ActionScript or a seasoned, left-brained programmer who wants to create a custom interface to a back-end e-commerce system.

To get the most out of this book, you should already know some Flash basics:

- drawing
- tweening
- creating symbols
- creating buttons

That's it. If you know how to do these things at all, then you're ready for this book.

How This Book Teaches

Both my parents were teachers, and they shared the same philosophy. "There are three ways to teach," they said, "example, example, example." I think they're right on this one, and this book uses complete, fully functional examples for every aspect of interactivity and ActionScript in this book. All examples can be found on the accompanying CD and on Web sites at *www.phptr.com/advancedweb* and *www.wire-man.com/flash5/*.

I've tried to avoid using overly simple squares and circles to explain concepts (I mostly succeeded). Instead, I'll ask you to load small movies and manipulate them. You will see a number of examples using Jake the Fish, and I hope you'll forgive my small indulgence—it's the only way I can put my marine zoology degree to use.

此为试读,需要完整PDF请访问: www.ertongbook.com

Acknowledgments

First, thanks go to Karen McLean, my editor at Pearson PTR, for giving me the opportunity to write this book. It's been a treat to write this little tome. And, of course, thanks go to my wife Tanya, who let me stay home and write full-time for a few months (I recommend that everyone do this at least once).

I'd also like to thank Joshua Davis for graciously letting me use his open-source Flash movies that he's posted on *praystation.com* (I recommend visiting it at least once a week).

Extra thanks to James of *presstube.com* and to Branden of *figleaf.com* for the movie used in Chapter 7, "Complex Scripting."

And finally, to my reviewers, M.D. McDowell and Leon Atkinson, for slapping me into shape—the book is better for it.

Contents

此为试读,需要完整PDF请访问: www.ertongbook.com

1

Introduction to ActionScript

IN THIS CHAPTER

- What Is ActionScript?
- What Is ActionScript Good For?
- What ActionScript Can't Do
- Variables
- Objects and Object-Oriented Scripting
- Objects and Frame Actions
- Dot Syntax
- Properties
- Methods
- Functions

What Is ActionScript?

ActionScript is the scripting language Flash uses to control its movies and objects within the movies. If you want to do anything interactive in Flash, you'll need to use ActionScript. It allows you to execute different actions in a movie depending on what a user does or on what frame of the movie is being played.

ActionScript looks a lot like JavaScript, which Macromedia (the folks who wrote Flash) did on purpose. A specification called ECMA-262 was written to provide an international standard for the JavaScript language. ActionScript in Flash 5 is based on the ECMA-262 specification, so if you've used JavaScript before, a lot of ActionScript will look familiar to you. If you haven't used JavaScript before, don't worry—you'll get it.

Throughout this book, I'll be referring to *actions*. This is a general term, and an action roughly means "a chunk of ActionScript code that does something."

What Is ActionScript Good For?

Here's some of what you can do using ActionScript:

- Create multiplayer games
- Create engaging, user-aware navigation
- Send data to middleware like PHP and Cold Fusion
- Create and parse XML objects
- Communicate with JavaScript or ActiveX objects
- About a billion other things. You'll see.

What ActionScript Can't Do

- ActionScript can't talk directly to a database—you'll still need to use middleware like PHP, Cold Fusion, or ASP to do that.
- Unicode isn't supported, but ISO-8859 and Shift-JIS are.
- You can't use exception handling with `try`, `throw`, or `catch`.

Variables

If you're completely new to programming, it'll take a little while to master the fundamentals, depending on how linearly you can make your brain work. One of the most basic program-

ming concepts is that of the variable. This is the same variable you saw in algebra class in junior high. Some simple examples:

```
// "x" is the variable
x = 3;

// "message" is a variable that holds a string,
// i.e., usually text
message = "Please press the next button."
```

You'll probably use variables mostly to keep track of what the user is doing and what state certain movie clips are in. If this isn't clear right now, keep reading—as you see more examples, it should become clearer.

Objects and Object-Oriented Scripting

Both ActionScript and JavaScript are called *object-oriented* scripting languages. Let's go over what this means, since it's an odd concept if you haven't been exposed to it before.

Scripting languages organize information into groups called *classes*. You can create multiple instances of these classes, which are called *objects*.

Classes and objects are just ways to group together chunks of information. To use a real-world analogy, you could create a "bicycle" class. That bicycle has lots of different properties to it, such as how many gears there are, how big the tires are, and what color it is. You can also perform some actions with the bike, such as pedal, brake, and turn (these are called *methods*).

In Flash 5, all movie clips are objects (also called *instances*) of the class MovieClip. Since all movie clips are objects of the same class, they all have the same properties and the same methods.

Buttons also act like objects. While there are some significant differences between buttons and movie clips, which we'll cover later, I consider it—as do many other Flash folk—useful to think of buttons as a kind of movie clip.

Flash has a number of predefined objects you can access: Array, Boolean, Color, Date, Key, Math, Mouse, Number, Object, Selection, Sound, String, XML, and XMLSocket. We'll be seeing many of these objects in the tutorial section of this book. These objects are treated with excruciating detail in Appendix B, "ActionScript Reference."

Creating a Class

You don't have to restrict yourself to using only classes that Flash 5 has provided, such as Movie Clips. You can create your own classes using constructor functions. This is pretty advanced stuff, and if you can't think of why you'd want to create a new class, don't worry about it—usually, only advanced programmers build their own classes. This section is for them. Say you want a to create a 1980s band:

```
function HairBand(p,s)
{
     this.hair = "big";
     this.hair_dye = true;
     this.number_members = p;
     this.number_synthesizers = s;
}

function Breakup()
{
     this.hair_dye = false;
     this.hair = "crew cut";
}

// Now, actually create two objects using
// the HairBand constructor function.
kajagoogoo = new HairBand(3,4);
softcell = new HairBand(2,1500);

// Create a method for a hairband
HairBand.kajagoogoo.partyover = Breakup;
```

Object and Frame Actions

Here's the basic structure of an action:

```
whenSomethingHappens(input variables)
{
    do stuff
}
```

We'll be elaborating on this basic structure significantly.

There are two kinds of actions: frame actions and object actions.

Object Actions—Movie Clips

Object actions are actions, or chunks of ActionScript code, that are attached to an object. Most of the time, an object is a symbol that's either a button or a movie clip. Graphic symbols can't have actions, nor can shapes you draw on that stage that aren't symbols. You can create your own objects, as we just saw.

An object action is associated with an instance of a symbol, not with the symbol itself. Here's an example (see Figure 1-1).

1. Load the movie *chapter1/fish_drag.fla*.
2. Control ➠ Test Movie.
3. Notice that by positioning the cursor over each of the fish and dragging your mouse, you can move the fish on the left, but not the one on the right. In fact, no matter where you click, you move the fish on the left and the right one stays put.

Both fish are instances of the same movie clip symbol, but only one of them has an action attached to it. Let's see what that action looks like. Exit from the movie test and return to the Flash editor (the main program). Option-click (Mac) or right-click (Windows) on the fish that's draggable (it's on the left). Choose **Actions**.

The Object Actions panel appears, as shown in Figure 1-2.

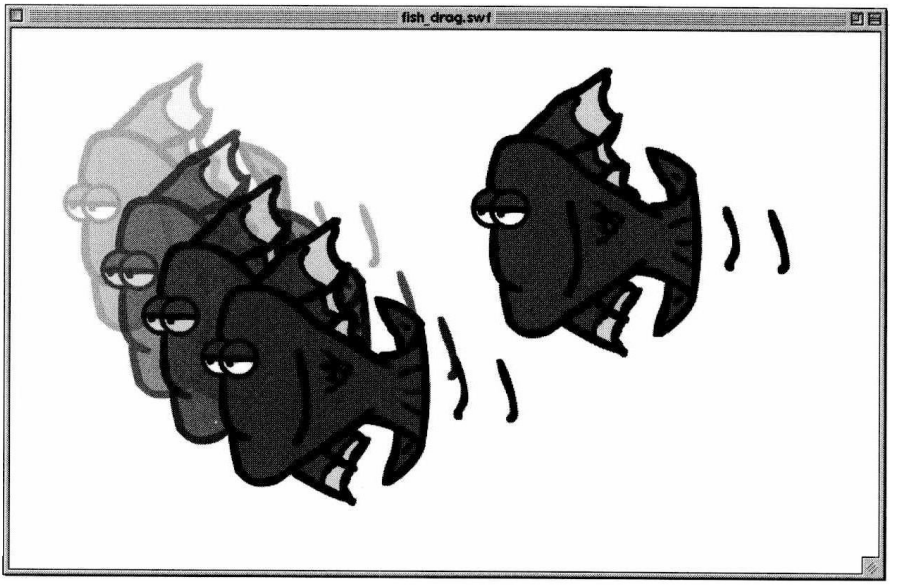

FIGURE 1–1 Dragging one of the fish

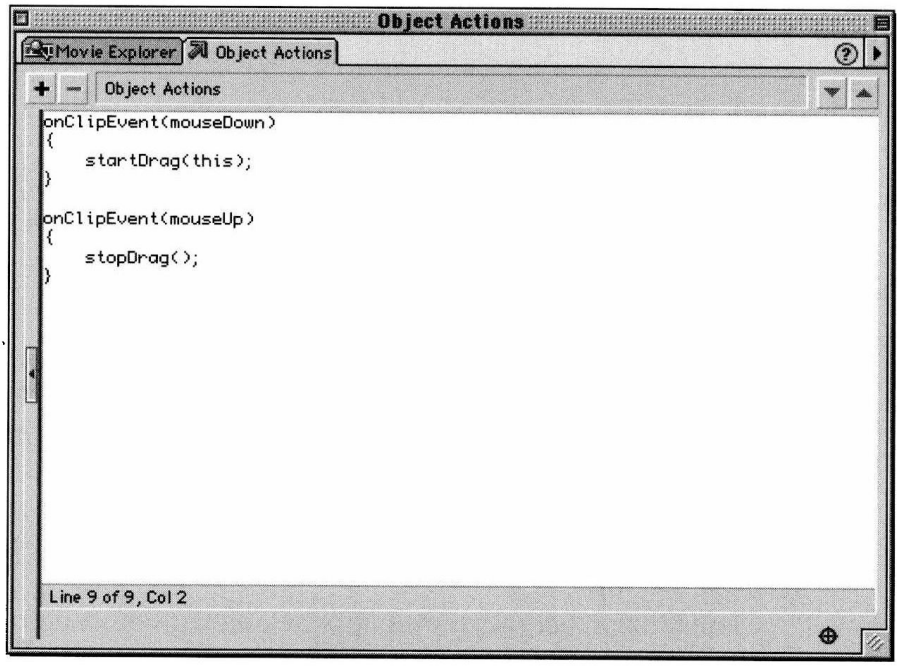

FIGURE 1–2 The Object Actions panel (Expert Mode)

此为试读,需要完整PDF请访问: www.ertongbook.com