

The Game Maker's Apprentice

Game Development for Beginners



Jacob Habgood
Mark Overmars



Apress®

The Game Maker's Apprentice: Game Development for Beginners

Copyright © 2006 by Jacob Habgood and Mark Overmars

All rights reserved. No part of this work may be reproduced or transmitted in any form or by any means, electronic or mechanical, including photocopying, recording, or by any information storage or retrieval system, without the prior written permission of the copyright owner and the publisher.

In purchasing this book, the authors and publisher grant you permission to use the electronic resources from the accompanying CD for commercial or noncommercial use in your own games made with Game Maker. However, redistribution of the original games or their resources is prohibited and the authors retain full copyright of all the original game concepts and the intellectual property associated with them.

ISBN-13 (pbk): 978-1-59059-615-9

ISBN-10 (pbk): 1-59059-615-3

Printed and bound in the United States of America 9 8 7 6 5 4 3 2 1

Trademarked names may appear in this book. Rather than use a trademark symbol with every occurrence of a trademarked name, we use the names only in an editorial fashion and to the benefit of the trademark owner, with no intention of infringement of the trademark.

Lead Editor: Chris Mills

Development Editor: Adam Thomas

Technical Reviewer/Additional Material: Sean Davies

Editorial Board: Steve Anglin, Ewan Buckingham, Gary Cornell, Jason Gilmore, Jonathan Gennick,

Jonathan Hassell, James Huddleston, Chris Mills, Matthew Moodie, Dominic Shakeshaft,

Jim Sumser, Keir Thomas, Matt Wade

Project Manager: Richard Dal Porto

Copy Edit Manager: Nicole LeClerc

Copy Editor: Liz Welch

Assistant Production Director: Kari Brooks-Copony

Production Editor: Ellie Fountain

Compositor: Dina Quan

Proofreader: Lori Bring

Indexer: Present Day Indexing

Artist: Kinetic Publishing Services, LLC

Illustrations and Cover Art: Kevin Crossley

Game Artists: Kevin Crossley, Matty Splatt and Ari Feldman

Cover Designer: Kurt Krames

Manufacturing Director: Tom Debolski

Distributed to the book trade worldwide by Springer-Verlag New York, Inc., 233 Spring Street, 6th Floor, New York, NY 10013. Phone 1-800-SPRINGER, fax 201-348-4505, e-mail orders-ny@springer-sbm.com, or visit <http://www.springeronline.com>.

For information on translations, please contact Apress directly at 2560 Ninth Street, Suite 219, Berkeley, CA 94710. Phone 510-549-5930, fax 510-549-5939, e-mail info@apress.com, or visit <http://www.apress.com>.

The information in this book is distributed on an “as is” basis, without warranty. Although every precaution has been taken in the preparation of this work, neither the author(s) nor Apress shall have any liability to any person or entity with respect to any loss or damage caused or alleged to be caused directly or indirectly by the information contained in this work.

*To halcyon days
with a frog,
a parrot,
and a talented bunch of gremlins.*

Contents at a Glance

Foreword	xiv
About the Authors	xvi
About the Technical Reviewer	xvii
About the Illustrator	xviii
Acknowledgments	xix
Introduction	xx

PART 1 ■ ■ ■ Getting Started

■ CHAPTER 1	Welcome to Game Maker	3
■ CHAPTER 2	Your First Game: Devilishly Easy	9

PART 2 ■ ■ ■ Action Games

■ CHAPTER 3	More Actions: A Galaxy of Possibilities	41
■ CHAPTER 4	Target the Player: It's Fun Being Squished	65
■ CHAPTER 5	Game Design: Interactive Challenges	85

PART 3 ■ ■ ■ Level Design

■ CHAPTER 6	Inheriting Events: Mother of Pearl	101
■ CHAPTER 7	Maze Games: More Cute Things in Peril	127
■ CHAPTER 8	Game Design: Levels and Features	149

PART 4 ■■■ Multiplayer Games

■ CHAPTER 9	Cooperative Games: Flying Planes	169
■ CHAPTER 10	Competitive Games: Playing Fair with Tanks	191
■ CHAPTER 11	Game Design: Balance in Multiplayer Games	211

PART 5 ■■■ Enemies and Intelligence

■ CHAPTER 12	GML: Become a Programmer	225
■ CHAPTER 13	Clever Computers: Playing Tic-Tac-Toe	245
■ CHAPTER 14	Intelligent Behavior: Animating the Dead	259
■ CHAPTER 15	Final Words	291
■ BIBLIOGRAPHY	297
■ INDEX	299

Contents

Foreword	xiv
About the Authors	xvi
About the Technical Reviewer	xvii
About the Illustrator	xviii
Acknowledgments	xix
Introduction	xx

PART 1 ■■■ Getting Started

■ CHAPTER 1	Welcome to Game Maker	3
	Installing the Software	3
	Registration	5
	The Global User Interface	6
	Running a Game	6
	How to Get More Information	8
	What's Next?	8
■ CHAPTER 2	Your First Game: Devilishly Easy	9
	Designing the Game: Evil Clutches	9
	Sprites	10
	Objects	13
	The Boss Object	13
	Events and Actions	14
	The Dragon Object	18
	Rooms	20
	Save and Run	22
	Instances and Objects	24
	Demons, Baby Dragons, and Fireballs	24
	The Fireball Object	24
	The Demon Object	27
	Summoning Demons	30
	The Baby Dragon Object	31

Backgrounds and Sounds	33
A Background Image	33
Background Music	34
Sound Effects	35
Congratulations	36

PART 2 ■■■ Action Games

CHAPTER 3	More Actions: A Galaxy of Possibilities	41
	Designing the Game: Galactic Mail	41
	Sprites and Sounds	42
	Moons and Asteroids	45
	Flying Around	50
	Winning and Losing	56
	An Explosion	56
	Scores	57
	Levels	58
	Finishing Touches	60
	A Title Screen	60
	Winning the Game	61
	Adding Some Visual Variety	62
	Help Information	62
	Congratulations	63
CHAPTER 4	Target the Player: It's Fun Being Squished	65
	Designing the Game: Lazarus	65
	An Animated Character	66
	A Test Environment	72
	Falling Boxes	73
	Finishing Touches	78
	No Way Out!	78
	Adding a Goal	79
	Starting a Level	80
	Sounds, Backgrounds, and Help	81
	Levels	82
	Congratulations	83

CHAPTER 5	Game Design: Interactive Challenges	85
	What Makes a Good Game?	85
	Game Mechanics	86
	Interactive Challenges	87
	Game Genres	87
	Challenges	88
	Difficulty	88
	Goals	89
	Rewards	90
	Subgoals	92
	Interactivity	92
	Choices and Control	93
	Control Overload!	93
	Unfair Punishment	94
	Audio Feedback	95
	Summary	96

PART 3 ■■■ Level Design

CHAPTER 6	Inheriting Events: Mother of Pearl	101
	Designing the Game: Super Rainbow Reef	101
	A Game Framework	103
	The Front-End	103
	The Completion Screen	106
	Bouncing Starfish	107
	Biglegs	113
	Parent Power	116
	Lives	117
	Blocks	120
	Normal Blocks	120
	Solid Blocks	120
	Special Blocks	121
	Polishing the Game	123
	Sound Effects	123
	Saving Games and Quitting	123
	A Slower Start	124
	Creating the Levels	125
	Congratulations	126

CHAPTER 7	Maze Games: More Cute Things in Peril	127
	Designing the Game: Koalabr8	127
	The Basic Maze	128
	The Game Framework	129
	A Moving Character	131
	Save the Koala	136
	Creating Hazards	137
	Tiles	140
	Adding Additional Hazards	143
	Locks and Switches	143
	A Detonator	144
	Rocks	145
	Finishing the Game	147
	Congratulations	148
CHAPTER 8	Game Design: Levels and Features	149
	Selecting Features	149
	Pie in the Sky	150
	Do You Have That in Blue?	151
	Starting an Arms Race	152
	One-Trick Ponies	152
	Emerging with More Than You Expected	153
	Designing Levels	154
	The Game Maker's Apprentice	155
	Learning Curves	156
	Difficulty Curves	158
	Saving the Day	160
	Applying It All	160
	Features	160
	Emerging Springs	161
	Training Missions	161
	Dividing Levels	162
	Summary	163

PART 4 ■■■ Multiplayer Games

CHAPTER 9	Cooperative Games: Flying Planes	169
	Designing the Game: Wingman Sam	169
	Variables and Properties	170
	The Illusion of Motion	173
	Flying Planes	174
	Enemies and Weapons	176
	Dealing with Damage	179
	Time Lines	182
	More Enemies	184
	End Boss	186
	Finishing Touches	188
	Congratulations	189
CHAPTER 10	Competitive Games: Playing Fair with Tanks	191
	Designing the Game: Tank War	191
	Playing with Tanks	192
	Firing Shells	195
	Secondary Weapons	199
	Views	205
	Congratulations	210
CHAPTER 11	Game Design: Balance in Multiplayer Games	211
	Competition and Cooperation	211
	Independent Competition	211
	Dependent Competition	212
	Independent Cooperation	213
	Dependent Cooperation	213
	Mix and Match	213
	Balanced Beginnings	214
	Equivalent Characters	214
	Balancing Differences	214

Balanced Choice	218
Weighting Choices	218
Cyclic Relationships	218
Balanced Computer Opponents	220
Artificial Stupidity	220
Summary	221

PART 5 ■■■ Enemies and Intelligence

CHAPTER 12 GML: Become a Programmer	225
Hello World	226
Variables	228
Functions	230
Conditional Statements	232
Repeating Things	234
Arrays	237
Dealing with Other Instances	239
Scripts As Functions	240
Debugging Programs	242
Congratulations	244
CHAPTER 13 Clever Computers: Playing Tic-Tac-Toe	245
Designing the Game: Tic-Tac-Toe	245
The Playing Field	246
Let the Computer Play	251
A Clever Computer Opponent	254
Adaptive Gameplay	256
Congratulations	257
CHAPTER 14 Intelligent Behavior: Animating the Dead	259
Designing the Game: Pyramid Panic	259
The Basic Framework	260
Creating the Maze and the Explorer	263
Expanding Our Horizons	265
Reactive Behavior	267
Time for Treasure!	269

Movable Blocks	270
Rule-Based Behavior	271
Walking Around	273
Moving Toward the Explorer	275
Dealing with States	277
Scarabs	280
Let There Be Light	284
Looking to the Future	289
 CHAPTER 15 Final Words	 291
Creating Resources	291
Artwork: The GIMP	291
Music: Anvil Studio	292
Sound Effects: Audacity	294
The Game Maker Community	294
Note to Teachers	295
Good Luck	296
 BIBLIOGRAPHY	 297
 INDEX	 299

Foreword

Way back when Mario was still a mere twinkling in Miyamoto's eye, I was the proud owner of a state-of-the-art Commodore 64 microcomputer. It came with a game development system called "The Quill," which allowed anyone to create their own text-based adventure games. It may have been incredibly crude, but it suddenly put at my fingertips the thrill of entertaining my nearest and dearest by devising "interactive challenges" of my own. Unfortunately, I knew little about game design, and rather than easing my players into a new and alien world, I treated them as opponents that had to be defeated before they could reach the end. Their spirits crushed, they left, never to return . . .

It took me years of playing a variety of good (and bad) games to eventually learn how to treat the player to the game-playing experience that their investment of time and money deserved. It took just hours of reading this book to wish I'd had its invaluable guidelines and the accompanying Game Maker tool to help me take my own first steps into game development all those years ago.

Two decades later, I now work for Real Time Worlds as the producer of Crackdown, an imminent Xbox 360 title developed exclusively for Microsoft. Crackdown is the result of over three years of development from a team that's now nearly 70 strong in Dundee (Scotland), with many more contributors across North America and Eastern Europe. This game has cost millions of pounds to create, and already consists of over two and half million lines of programming code! Blood, sweat, and tears have been poured into this title to provide cutting-edge graphics technology, stunning art assets, and dramatic surround sound. We've spent days (and nights) wrestling with new technologies to provide the player with a "playground" and "toy set" that was previously only the stuff of dreams.

Nonetheless, once you strip away the gloss, Crackdown boils down to a handful of game-play linchpins, or what we term the "pillars of play." Take it from me that when charged with building such a grand gaming monument, it is vitally important to have absolute faith in the basic foundations! I was therefore very pleased to see that this book encourages you to identify these pillars (or game mechanics) and discover how a system of simple rules can combine in unique and compelling ways to create a spellbinding experience.

As you progress through the book you'll build a series of excellent games that you might never have even dreamed you could be capable of creating right now. The instructions are clear and concise, but also encourage you to experiment with your own designs. For example, your version of the captivating and original Koalabr8 game (Chapter 7) will almost certainly be a unique piece of software. The crazy devices you invent, and the way you lay out your levels, will certainly differ from mine. Watch out for Lazarus too (Chapter 4)—it may interest you to know that this eponymous hero first appeared in Jacob's student portfolio, and was partly responsible for securing his first programming job in the industry!

Mark and Jacob have brought together decades of game development expertise in this book. As well as being a professor of computing, and the creator of Game Maker, Mark first cut his game-programming teeth creating versions of games like Super Breakout for the Atari ST. Jacob has a string of titles to his name, and his in-depth knowledge of “the craft” consistently yields outstanding results. Never more so was this the case than when I had the pleasure of working with him on the team that created the PlayStation hit *Hogs of War* (also mentioned in this book). Where Jacob differs from his peers is in his mastery of all four of the fundamental game development disciplines: programming, sound, art, and, of course, design. Now, thanks to Mark’s Game Maker software, you can find out what it feels like in their world!

One of the key messages I hope you’ll take away from this book is that there’s a world of difference between having a great idea for a game and being a great game designer. The initial idea is simply the seed from which the game grows, or the stone from which the pillars are hewn. The role of a designer is to fully realize the vision: conceiving and continually refining the various supporting mechanisms to make them mesh like the components of a Swiss time-piece. As is repeatedly stated in these pages, there is no correct solution to game design—only a great idea, well executed and injected with personal flair and enthusiasm. Even if you’re struggling to pin down that idea right now, I’m sure you will have wrestled it onto the screen and into the hands of friends and family before finishing the final chapter of *The Game Maker’s Apprentice*.

Good luck!

Phil Wilson
Producer, Real Time Worlds



Crackdown

About the Authors

JACOB HABGOOD is 30 years old and has been writing computer games since he was 10. He wanted to be a psychologist when he grew up, but somehow he ended up with a computer science degree and went into the games industry instead. He worked as a professional game developer for seven years, programming console games for Gremlin Interactive and Info-frames/Atari in the north of England. During this time he contributed to a range of successful titles and led the programming teams on Micro Machines (PlayStation 2, Xbox, and Nintendo GameCube) and Hogs of War (PlayStation).

Jacob is now a doctoral student at the University of Nottingham, researching the educational potential of computer games. As part of this research, Jacob runs clubs and workshops teaching children and teenagers how to make their own computer games and provides free teaching resources through his website: gamelearning.net. All being well, this work will soon earn him a Ph.D. from Nottingham's Department of Psychology so that he can finally consider himself grown up.



MARK OVERMARS is a full professor in computer science at Utrecht University in the Netherlands. There he heads the research center for Advanced Gaming and Simulation (www.gameresearch.nl) in which researchers from different disciplines collaborate on all aspects of gaming and simulation. One of Mark's prime research domains is computer games. He is also one of the founders of the Utrecht Platform for Game Education and Research (www.upgear.nl), a collaboration of different game-related educational programs in the Netherlands. For many years he has taught courses on computer game design at Utrecht University, and has given lectures on game design to many types of people (high school kids, teachers, researchers, and politicians). Mark is the author of a number of popular software packages, in particular, the Game Maker software package used as the development tool in this book.

About the Technical Reviewer

SEAN DAVIES is 28 years old and has been fascinated by computer games from an early age. He grew up fairly certain that he would become a novelist—or possibly a rock star, but eventually came to a number of important realizations:

1. He's probably never going to be a rock star.
2. Game programming is quite cool, though.
3. Badgers are just really big weasels (the exchange rate is approximately 20 weasels to the badger if you're interested).
4. Any attempts to construct a serious calculus of the family Mustelidae are probably best kept to yourself—people think you're strange (see above).

Having made these startling realizations at such an early stage, the rest of his career path was pretty much decided. After graduating with a degree in computer science, he joined Infogrames in Sheffield, UK, and has worked in the games industry ever since. When Infogrames Sheffield closed its doors in 2002, he joined Sumo Digital, where he still is today. Sean is currently Xbox platform lead on *Outrun 2006: Coast to Coast*, which Sumo is developing for SEGA, and is looking forward to working on some next-generation console programming in the near future. He currently has no intention at all of ever becoming properly grown up.



Outrun 2006: Coast to Coast

About the Illustrator

The first things **Kev Crossley** remembers drawing as a child were some Daleks and the Incredible Hulk, and he knew from that point on that he would grow up to be one or the other. When Kev was five, his dad brought home the videogame Pong, and Kev has been trying to come to grips with it ever since. Nonetheless, he has managed to get through a couple of Zelda games and has spent many a happy hour blasting the head off one of this book's authors with a Rail Gun.

Kev spent some time in a university, but eventually realized that an art degree was not going to give him access to Time Lords or gamma rays, so he decided to work in a bakery instead. Eventually he got sick of eating coconut macaroons and biscuits and applied for a job making videogames for a little green monster. For the next third of his life, he had a great time producing graphics and animation for over 20 titles—some of which were quite good.

These days Kev is a concept artist at Core Design in Derby, UK, where he has occasionally been known to hang out with women obsessed with tombs. He has also done copious amounts of freelance illustration and writing for publishers all over the world, including a series of instructional drawing books and sequential work for Rebellion's sci-fi comic *2000AD*. His one regret is that he can't ride a skateboard, because a cross between the Daleks and Tony Hawk would be unstoppable.



Acknowledgments

By rights, this book really shouldn't exist, because it's required far too much effort from far too many people to make it a profitable endeavor. Nonetheless, it does exist, and as a result there are a lot of people who need to receive our heartfelt thanks in helping us to realize this labor of love.

First and foremost, we need to thank all those people closest to the individuals directly involved in bringing this book into existence. Their influence may not be obvious to the reader, but projects like this could never happen without the support and understanding of the wives, girlfriends, and families of all the people involved in this book. In particular, Jacob would like to thank Jenny, Fiona, Michelle, and Amelia, all of whose names should adorn the credits of many a videogame for the sacrifices that they regularly make to indulge the creative passions of their loved ones.

The next biggest thanks should go to Matty Splatt, who has shied away from a full billing in the "About the" sections but did a fantastic job of bringing Koalabr8 and Pyramid Panic to life with his comical and beautifully polished graphics. We would also like to give a special thanks to Ari Feldman for allowing us to use and modify his game sprites for the Wingman Sam game.

More thanks go to Jenny and Marguerite Habgood for their grammatical critiques, and the following people for all their comments and input into the project: Sarah Peacock, Judy Robertson, John Sear, and Phil Wilson. Additional thanks goes to all the staff and students at Sheffield West City Learning Centre, who suffered the book's instruction in its earliest form.

Quick thanks also to everyone who enthusiastically play-tested the games in the book, including Gail Clipson, Fiona Crossley, Katie Fraser, Giulia Gelmini, Jasmin Habgood, Martijn Overmars, Ronald Overmars, and Stuart Reeves.

Finally, we would like to thank everyone at Apress for their support, and for sticking with us even after it became plainly obvious that both authors were far too busy to write a book!



Introduction

Who wouldn't want to make computer games? It's creative, rewarding, and these days even pretty darn cool too. You can make them to share with your school friends, your work colleagues, your grandchildren, or even the entire gaming world. This book is not specifically for the young or old, but anyone who loves computer games and wants to have a go at making them for themselves. We've all painted a picture, written a story, and made a wobbly piece of pottery at some point in our lives, so it's now time to embrace the art form of the future and try making computer games too.

This book provides a collection of engaging tutorials that introduce you to the Game Maker tool and teach you how to use it. The first four parts of the book take you step by step through seven different projects using Game Maker's simple drag-and-drop programming system. By the time you've finished making Evil Clutches, Galactic Mail, Lazarus, Super Rainbow Reef, Koalabr8, Wingman Sam, and Tank War, you'll have a well-rounded experience of making games with Game Maker. Parts 2, 3, and 4 also end with game design chapters that encourage you to stand back from your creations and consider how principles of game design can be used to make them more fun. Moreover, we don't just talk about it, but we provide new versions of the games with improved features so that you can experience for yourself how solid game design can lead to good gameplay.

Game Maker provides a simple environment that allows complete beginners to quickly start building games, using an icon-based system of events and actions (see Figure 1). This drag-and-drop programming technique provides an easy way to learn about game development and allows you to create complete games without going near a traditional programming language.

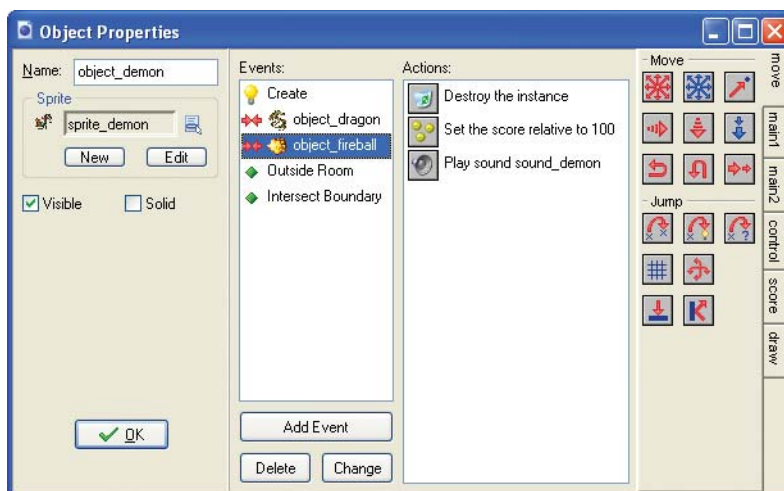


Figure 1. Game Maker's simple drag-and-drop system uses iconic events and actions to program computer games.

However, once you become more experienced, traditional languages can provide a more powerful way to program games. Consequently, Game Maker also provides the Game Maker Language (GML), which underpins Game Maker and makes it such a powerful tool (see Figure 2). The last part of the book uses several simple examples to introduce you to GML, before we demonstrate how you can use it to create artificial intelligence for undead creatures in the Pyramid Panic game.

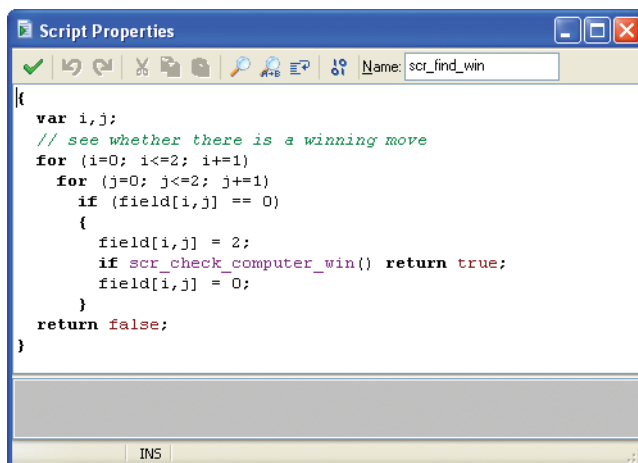


Figure 2. Game Maker Language (GML) provides extra power for advanced users.

The example games in this book have been brought to life with graphics and illustrations by real games industry artists. Furthermore, you can use all the professional resources provided on the CD in your own Game Maker projects with the blessing of the publisher and authors. We only ask that you share your creations with the online Game Maker community so that we can see what you have created with them. We want you to enjoy the creative journey ahead and hope that it will help you to share in our passion and enthusiasm for creating computer games!