



TECHNOLOGY IN ACTION™

The long-awaited sequel to
The Game Maker's Apprentice

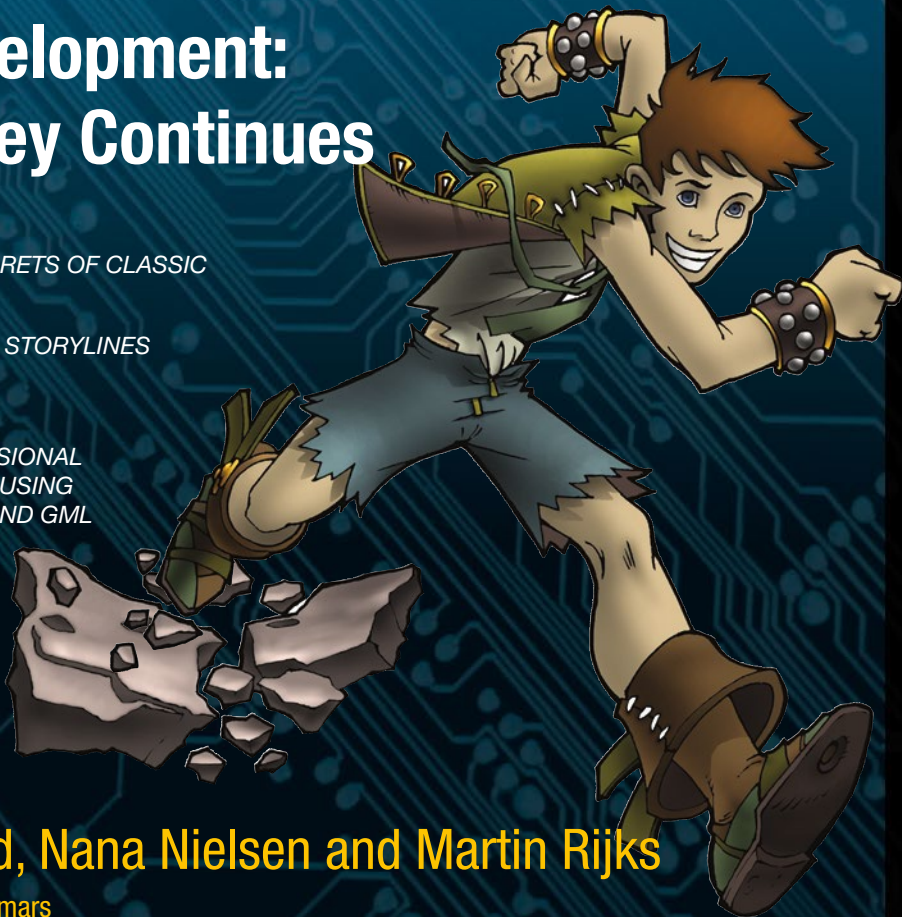


The Game Maker's Companion



Game Development: The Journey Continues

- DISCOVER THE SECRETS OF CLASSIC PLATFORM GAMES
- CREATE ENGAGING STORYLINES WITH BELIEVABLE CHARACTERS
- PROGRAM PROFESSIONAL PLATFORM GAMES USING DRAG-AND-DROP AND GML



Jacob Habgood, Nana Nielsen and Martin Rijks

with foreword by Mark Overmars
Artwork by Kevin Crossley

Praise for The Game Maker's Companion

Game Development: The Journey Continues

"The Game Maker's Companion is packed with expert advice, practical information and technical knowledge, which together provide a thorough overview of the nuts and bolts of video game creation."

Ade Carless – Designer of Zool

"The Game Maker's Companion brings back the days where you could pick up a book and learn all about creating games, fantastic!"

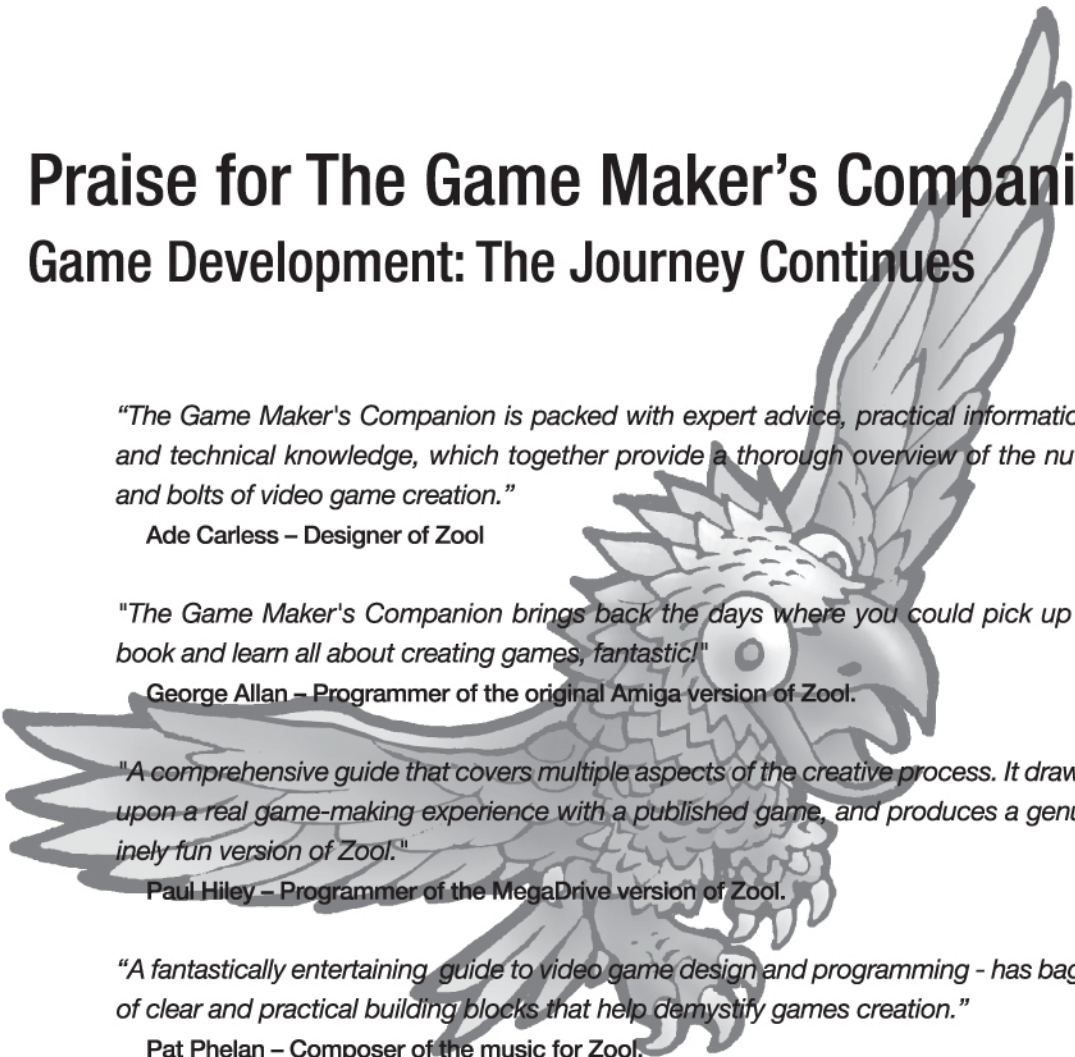
George Allan – Programmer of the original Amiga version of Zool.

"A comprehensive guide that covers multiple aspects of the creative process. It draws upon a real game-making experience with a published game, and produces a genuinely fun version of Zool."

Paul Hiley – Programmer of the MegaDrive version of Zool.

"A fantastically entertaining guide to video game design and programming - has bags of clear and practical building blocks that help demystify games creation."

Pat Phelan – Composer of the music for Zool.



The Game Maker's Companion

Game Development: The Journey
Continues



Jacob Habgood

Nana Nielsen

Martin Rijks

Kevin Crossley

Apress®

The Game Maker's Companion: Game Development: The Journey Continues

Copyright © 2010 by Jacob Hagbood, Nana Nielsen, Martin Rijks, Kevin Crossley

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.

ISBN-13 (pbk): 978-1-4302-2826-4

ISBN-13 (electronic): 978-1-4302-2827-1

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

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

The use in this publication of trade names, trademarks, service marks, and similar terms, even if they are not identified as such, is not to be taken as an expression of opinion as to whether or not they are subject to proprietary rights.

President and Publisher: Paul Manning

Lead Editor: Steve Anglin

Development Editor: Matthew Moodie

Technical Reviewer: Sean Davies

Editorial Board: Clay Andres, Steve Anglin, Mark Beckner, Ewan Buckingham, Gary Cornell, Jonathan Gennick, Jonathan Hassell, Michelle Lowman, Matthew Moodie, Duncan Parkes, Jeffrey Pepper, Frank Pohlmann, Douglas Pundick, Ben Renow-Clarke, Dominic Shakeshaft, Matt Wade, Tom Welsh

Coordinating Editor: Kelly Moritz

Copy Editor: Ralph Moore

Compositor: MacPS, LLC

Indexer: Toma Mulligan

Artist: April Milne

Cover Designer: Anna Ishchenko

Distributed to the book trade worldwide by Springer Science+Business Media, LLC., 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 www.springeronline.com.

For information on translations, please e-mail rights@apress.com, or visit www.apress.com.

Apress and friends of ED books may be purchased in bulk for academic, corporate, or promotional use. eBook versions and licenses are also available for most titles. For more information, reference our Special Bulk Sales–eBook Licensing web page at www.apress.com/info/bulksales.

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 the offspring of childish minds.

Please note: when the text of this book refers to a “CD” or “CD Rom,” please refer to the supplementary material on [apress.com](http://www.apress.com) to access that content. This supplementary material can be found at http://www.apress.com/downloadable/download/sample/sample_id/1197/.

Contents at a Glance

■ Contents	v
■ Foreword	xiii
■ About the Authors.....	xv
■ About the Artists	xvii
■ About the Technical Reviewer	xix
■ Acknowledgments	xx
■ Introduction	xxi
■ Part 1: Fishpod	1
■ Chapter 1: Greetings, Game Maker.....	3
■ Chapter 2: Platform Beginnings: An Idea with Legs	13
■ Part 2: Zool	45
■ Chapter 3: Zool: Taking It to the N th Dimension	47
■ Chapter 4: Empowerment: Sliding Ninjas.....	71
■ Chapter 5: Krool’s Forces: Sweetening the Challenge.....	103
■ Chapter 6: Fighting Talk: The Empower Strikes Back	133
■ Part 3: Shadows on Deck.....	163
■ Chapter 7: Game Design: “Shadows on Deck”	165
■ Chapter 8: Storytelling in Theory.....	181
■ Chapter 9: Storytelling Applied.....	199
■ Chapter 10: Of Mice and Pen: Pirate Art.....	219
■ Chapter 11: GML: From Ninja to Pirate	249
■ Chapter 12: Rogues’ Rendezvous: Vertically Sliced	273
■ Chapter 13: The Story Begins	307
■ Part 4: Reference	339
■ Chapter 14: Feature Reference	341
■ Appendix: Rogues’ Rendezvous: Dialogue.....	385
■ Index.....	395

Contents

■ Contents at a Glance.....	iv
■ Foreword	xiii
■ About the Authors.....	xv
■ About the Artists.....	xvii
■ About the Technical Reviewer	xix
■ Acknowledgments	xx
■ Introduction	xxi
■ Part 1: Fishpod	1
■ Chapter 1: Greetings, Game Maker.....	3
Resources	3
Instances and Objects.....	5
Variables	5
Local Instance Variables	6
Variables in Other Instances	7
Global Variables	8
Coordinates and Angles	8
Transparency	10
Congratulations.....	11
■ Chapter 2: Platform Beginnings: An Idea with Legs	13
A Fish Called Pod	13
Sprites.....	15
Collision Masks	16

Platform Objects	20
Platform Waiting Room	22
State Machines	23
State Objects.....	26
Collision	31
Problem 1: Just Passing Through	31
Problem 2: Get Your Inside Out.....	32
Walk On.....	32
Jump Up.....	34
Fall Down	36
Challenges	37
Goals	40
Finishing Touches.....	41
Congratulations.....	43
■ Part 2: Zool	45
■ Chapter 3: Zool: Taking It to the Nth Dimension	47
Designing a Ninjalien	47
Long-Term Challenge.....	49
Short-Term Challenges	49
Ready-Baked Resources.....	50
Kitchen Tiles	51
An Alien State	55
An Alien’s First Steps.....	57
Marching Order	61
Room with a View	62
Crash Landing a Ninja.....	64
Congratulations—Phase 1 Complete!.....	68
■ Chapter 4: Empowerment: Sliding Ninjas.....	71
Balance of Power	71

Empowering Up.....	73
You Cannae Break the Laws of Physics.....	75
A Floating Garage Worker.....	76
Underground, Overground.....	78
Driven up the Walls.....	79
Wall Polish.....	84
Ramping Up.....	87
Sliding Down.....	90
Sound Movement.....	96
Congratulations—Phase 2 Complete!.....	100
Chapter 5: Krool’s Forces: Sweetening the Challenge.....	103
Challenging Ingredients.....	104
Reheated Resources.....	105
Gotta Get ‘Em All.....	106
Spoil Your Dinner.....	107
A Solid Problem.....	108
Health Hazards.....	111
Spiky Fright.....	112
Bitter Sweets.....	114
Congratulations—Phase 3 Complete!.....	130
Chapter 6: Fighting Talk: The Empower Strikes Back.....	133
Ant Attack.....	134
Splitting Headache.....	134
Slide Kick.....	135
Top Spinning.....	138
Shooting Stars.....	139
Uplifting Platforms.....	140
Up and Down.....	141
Left and Right.....	142
Relativity.....	144

Not So Solid Ground	146
Parallax Universe	147
Keeping Score	151
Zoolaphobia	158
Programming Memories	159
Congratulations—100% Complete!	159
■ Part 3: Shadows on Deck.....	163
■ Chapter 7: Game Design: “Shadows on Deck”	165
A Good Beginning.....	166
The Hero.....	167
Make Him Invisible.....	167
Make Him Fantastic	168
Make Him Mortal.....	168
Flynn	168
The Beginnings of a Story	169
Game Play	170
Mechanics of “Shadows on Deck”	171
Challenges	175
Difficulty.....	179
Conclusion	180
■ Chapter 8: Storytelling in Theory.....	181
What Can a Story Do for My Game?.....	181
Little or Big Story?	182
Story Structures.....	182
The Three-Act Structure.....	183
The Hero’s Journey	183
The Structures Working Together	188
Pace	189
Characters.....	190
Hero.....	191

The Rest of the Crew.....	191
Filling Roles.....	193
Traits.....	193
Interactive Stories.....	194
Branching.....	194
Modifying Non-Player Character Behavior.....	196
Changing Character Appearance.....	196
Cosmetic Changes.....	197
Tools to Tell a Story Inside a Game.....	197
Conclusion: Where’s the Skip Button?.....	198
Chapter 9: Storytelling Applied.....	199
Applying the Theory to <i>Shadows on Deck</i>	199
The Ordinary World.....	199
The Call to Adventure.....	200
The Refusal to the Call.....	200
The Meeting with the Mentor.....	200
Crossing the First Threshold.....	201
Tests, Allies, and Enemies.....	201
Approach to the Innermost Cave.....	201
The Ordeal.....	201
The Reward.....	202
The Road Back.....	202
The Resurrection.....	203
Return with the Elixir.....	203
Archetypes for Our Story.....	203
Flynn.....	204
Father.....	205
Mother.....	206
The Parrot Archibald.....	207
The Pirate King.....	208
Level Bosses.....	209
Minor Characters.....	210

Titles Tantrum	210
Storytelling Tools for <i>Shadows on Deck</i>	212
Clues	212
Dialog	212
Cut Scenes	213
Conclusion	218
■ Chapter 10: Of Mice and Pen: Pirate Art.....	219
A Little Art History	219
Shadows On Spec	219
The Art Process	220
Phase 1: Conceptual Design	220
Phase 2: Asset Creation	229
Phase 3: Art Integration	238
We've Got It Covered.....	241
Conclusion: Save Progress and Continue!	248
■ Chapter 11: GML: From Ninja to Pirate	249
GML Babel Fish	249
Literal Translation	252
Translation Tips.....	255
The Power of Language	256
A Global Evil	256
Switching Sprites.....	258
The Ninja Elbow Walk	260
Scripts.....	262
Delayed Lift	266
Goodbye Ninja, Hello Pirate.....	267
The Problem with Giants	268
Move On Top	270
Ignoring Ledges in the Air	270
Empty Collisions.....	271
Congratulations.....	272

- Chapter 12: Rogues' Rendezvous: Vertically Sliced 273**
- Animated Driving 273
 - Best Foot Forward..... 274
 - Over the Top..... 277
 - Jump High, Fall Hard..... 283
 - Half a Pixel 287
- Lights, Camera, Action!..... 288
 - Pretty Useful Polly 289
 - Numb Skulls..... 295
 - Camera Focus 303
- Congratulations!..... 306
- Chapter 13: The Story Begins 307**
- Emotivation 307
 - Cutting the Scene..... 308
 - Fetch the Popcorn..... 311
- Level Design..... 311
 - Theory in Practice 312
 - Progression 312
 - Micro Curves 313
 - Walkthrough..... 313
 - Game Play Tweaking..... 318
- Traps and Puzzles..... 319
 - Traps 319
 - Puzzles..... 320
- Storytelling..... 322
 - Mood 322
 - Characters..... 322
 - Bigger Story 325
- Implementing Dialogue 326
 - Scripting Dialogues 326
 - Punctuating Text 331

Congratulations.....	336
Final Words	337
■ Part 4: Reference	339
■ Chapter 14: Feature Reference	341
360-Degree Movement	342
360-Degree Shooting	343
Cheat Codes	345
Fancy Buttons	347
Four- and Eight-Way Movement	348
Grid Movement.....	350
Homing Missiles	356
Mini-Map.....	358
Mouse Cursor.....	362
Moving Through Rooms	363
Orbiting Objects	367
Patrolling Enemies	368
Pausing the Game.....	372
Pushing Boxes	373
Scrolling Text (Horizontal).....	375
Scrolling Text (Vertical).....	377
Smoke Trail	378
Smooth Motion.....	379
Snap To Grid	380
Toggle Music and Sound Effects.....	381
■ Appendix: Rogues' Rendezvous: Dialogue	385
■ Index	395

Foreword

When I created the first version of Game Maker in 1999, I naturally had no idea that it would achieve its current status of the most-used game-creation package in the world. The program was still rather limited. You could only create very simple games with it and the package was downloaded only a few hundred times per month.

A lot has changed over the past ten years. New versions of Game Maker were released that made it possible to create sophisticated stand-alone games. The number of downloads rose to over 150,000 per month, and the company YoYo Games was formed that now develops and distributes the program. The site of YoYo Games (www.yoyogames.com) has already collected close to 100,000 games made with Game Maker and this number is rapidly increasing.

But one thing has not changed. Most of the games created with Game Maker are still very simple and do not use many of the advanced features that the program offers. They also often lack sophistication in their gameplay. One of the reasons for this is that there was no text book for the more advanced Game Maker users. That is, until this book came along. A book dedicated to those that have already created their first games and want to learn more.

Using the popular genre of platform games, the first part of the book introduces various advanced aspects of Game Maker. In an entertaining yet precise way, the authors show you how to create increasingly complicated versions of the game *Zool*, resulting in a game of commercial quality. Once you have worked your way through this part, you will be able to create many interesting platform games yourself.

To make great games, it is not enough to know the advanced aspects of the Game Maker program. It is equally important to understand the design principles on which good games are based. In the previous book, *The Game Maker's Apprentice*, that I wrote together with Jacob Habgood, we introduced some basic game design concepts such as challenges, level and feature design, and balance. In the second part of this new book, Jacob and his co-authors take this a step further.

In particular, they explain how to design fascinating storylines and characters and how to apply this in your games. As an example, you create a pirate game, *Shadows on Deck*, which also demonstrates the effect of using a very nice silhouetted graphical style. Good storylines will stimulate your players to continue playing your games. The players want to find out what is happening next. Good character design will create a bond between the player and the characters, making them care. They become attached to the characters, which further enhances the game play. Use it in your own games and players will appreciate your work a lot more.

The final part of the book consists of a large reference section with solutions to many features that you might want to put in your games. It handles such diverse topics as how to shoot to the mouse position, how to create fancy buttons, how to display a mini-map, how to make enemies patrol an area, and how to display scrolling text. It provides answers to many questions users might have, making this book a useful companion for every Game Maker user.

I am convinced that after reading this book, you will be an even better game maker than you are now and that you will be able to create games that many players will enjoy. Don't keep them to yourself, though. Share your creations with the rest of the world. You can publish them through the YoYo Games web site (www.yoyogames.com) so that everybody can play them for free. In 2010, YoYo games will also introduce the possibility to sell your games on the PC and PSP, preparing the way for other platforms in the future. So this book could actually be the start of your professional career as a console game designer. But, more importantly, it is a book that will help you enjoy creating exciting games.



*Mark Overmars
Creator of Game Maker*

About the Authors

Jacob Habgood

Jacob's career in the games industry spans 14 years, but he has been programming games as an amateur and professional now for a quarter of a century. During this time, he has worked on over a dozen published games for all the major console platforms, including as the lead programmer of the cult British game, *Hogs of War*. He has programmed, designed, and project-managed titles for such publishers as Gremlin, Infogrames, Atari, Disney, and Konami and he truly knows the meaning of the word "crunch."

Somehow he also found time to study a Ph.D. in the psychology of learning and has a passion for most things that connect video games and learning (including a whole lot of research involving zombies: www.zombiedivision.co.uk). This passion also extends to teaching game development, which he practices in his position as Senior Lecturer in Game Development at Sheffield Hallam University in the UK.

Jacob is the husband of a primary school teacher who is talented enough to inspire children without the need for a PlayStation controller, and the father of two beautiful daughters who are the product of their mother's inspiration.



Nana Nielsen

Nana Nielsen grew up in Denmark under the watchful eyes of two computer programmers: her parents. Being force-fed Tolkien books and text adventures on the Commodore 64, she developed a keen interest in both stories and games, and how the two intermingle. Since then, she has earned a degree in Interactive Media and worked in the games industry for more than a decade and published a dozen titles in different genres, including the platformer *Crash Bandicoot: Twinsanity*, the RPG *Sudeki*, the adventure series *Broken Sword*, and the sports title *Virtua Tennis*. She is currently working on the popular episodes of *Doctor Who*—*The Adventure Games*.



Martin Rijks

Martin Rijks wrote his first lines of code on paper, at the age of eight, using a book from the library that he probably returned too late. Not owning a computer himself until years later, little Martin had to wait for birthday parties at his uncle's to actually be able to test his programs on a TI99/4A. When he had finally bought his own, he wasted the best years of his youth dashing boulders, shooting mutant camels, raiding stars, or navigating several alternate realities carrying potions while swinging pointy weapons at critters.

Martin discovered Game Maker in 2001, and ever since has kept prodding it to see what it would do. Having played an important role in building and maintaining the lively Game Maker Community, you can still occasionally find him there telling people that They Are Doing It Wrong. For fear of not wanting to go home after working hours, Martin was fortunate enough to find a daytime job that has nothing to do with game development. These days hardly ever gaming, he still likes to challenge people for a multi-player match of *Duke Nukem 3D*, but he is unable to find anyone who is still willing to play it.

Having become a father while missing another deadline for the book, Martin is already planning to give his newborn son Dimar the same sermons he got from his own parents on the virtues of Playing Outside and Getting Some Fresh Air. This time, he hopes they will work.

About the Artists

Kevin Crossley

Kevin Crossley began his long career as an artist at the tender age of three, when he discovered that rather than eating poster paint, it was actually useful for painting with. It was a while before he worked out that the teachers shouted less when you kept the brush on the paper, (rather than, for example, the trouser leg of Mr. Robinson!) Nevertheless, he displayed a precocious talent for one so young, and by the age of seven, he was composing vast battles between armies of Daleks and Hulked-up dinosaurs on his bedroom walls (as he still hadn't mastered the art of staying on the paper.)

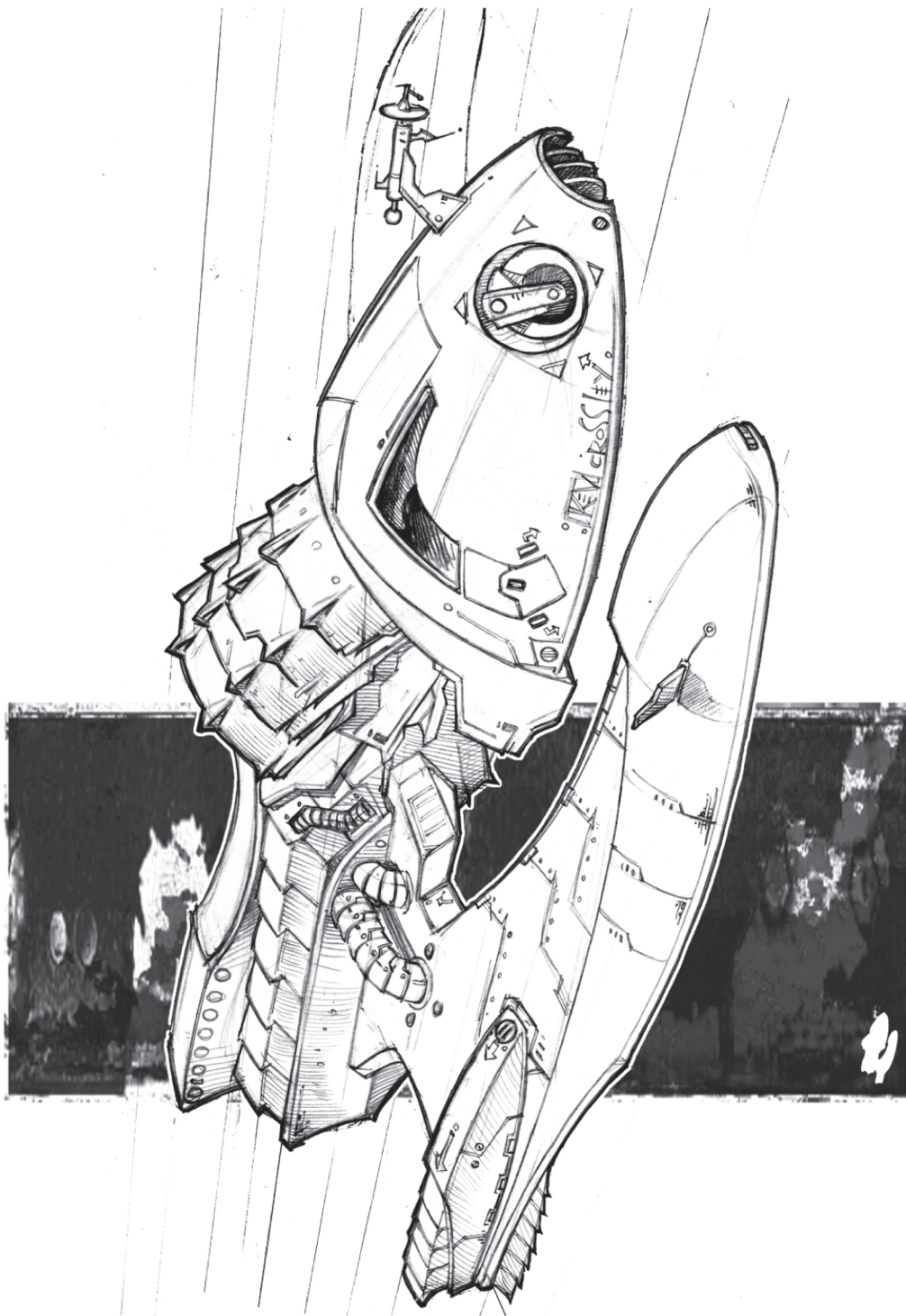
Such a promising start augured well for the future, and after a distracting three years at University studying typography and how not to design stationary, he stumbled into a job as a video game artist. This was a role he enjoyed for 15 years before becoming a freelance illustrator, contributing to numerous RPG books and comics such as *2000AD* and *KISS4K*. His book *Fantasy Clip Art* was published in 2006, and he writes regularly for various art magazines, including *ImagineFX*.

His grueling work schedule is made bearable by the un-swaying support of his wonderful wife, Fiona, and thanks to the example set by his two-year-old son Aidan, Kev's brushes still stray from the paper....

Griffin Warner

Griffin is a technical artist specializing in animation and he was responsible for creating the character animations used for the *Shadows on Deck* example in this book. He is a former student of Jacob's, and graduated from Sheffield Hallam University in 2010 with a first-class degree in Games Software Development B.Sc. (Hons). At the time of writing, he is looking forward to a successful career in game development.





About the Technical Reviewer

Sean Davies

Sean was just 21 years old when his life was changed forever. Indoctrinated into a Top Secret government training program, Sean was transformed from a mild-mannered, floppy-haired guitarist into a cold-blooded coding machine. Rebuilt from the ground up (more than 70% of his blood replaced by machine code), he can now exist on diet of pure caffeine (with occasional pizza supplements) and has been forced to take on a treacherous double life. By day, he appears to be a perfectly normal early-30s male—chatty, outgoing, approachable. By night, he stalks the halls of Sheffield game development houses viciously optimising any code unfortunate enough to get in his way with the power of his bionic stare.

In the ten years since this change, he has worked on games for almost all of the major console platforms and for several major franchises including *Men in Black*, *Superman*, *Outrun*, *Virtua Tennis*, *Sega Superstars*, and *Doctor Who*.

Sean Davies does not sleep—he codes....



Acknowledgments

Writing a book like this one takes an enormous amount of time and commitment from all the people directly involved in its development, but these are not the only people who deserve recognition for their contribution to making this book a reality.



A huge debt of gratitude is owed to Ian Stewart for allowing us to reintroduce a lost icon of gaming history to a new generation of game developers. The original *Zool* resources and artwork have proved a fabulous addition to the offering of this book, while providing a fitting record of a retro classic. Further thanks go to Paul Hiley, Adrian Carless, George Allan, and Patrick Phelan for their assistance in reassembling the pieces after so much time.

Nonetheless, even intergalactic ninjas wouldn't get very far without the support of their nearest and dearest and we cannot express enough appreciation for the understanding and patience of our families while creating this book. Consequently, we extend our heartfelt thanks to Jenny, Elsie, and Jess Habgood, Pete Ellacott, Fiona and Aidan Crossley, Lucienne Klinkenberg, and Dimar Rijks. A particularly special welcome is extended to Jess and Dimar, who made their entrance into the world during the writing of this book.

Further thanks go to those who have tested and provided feedback on the book's content in some form, including Mark Overmars, Barry Lowndes, and Alex Aris. Our gratitude is also extended to Sandy Duncan of YoYo Games for providing permission to include Game Maker along with the CD.

Additional thanks goes to the members of the English Amiga Board forums for their support in obtaining *Zool* resources, and the Game Maker Community forums for their support and feedback on the plan for the book.

Finally, we would like to thank everyone at Apress for their unwavering support in our blind pursuit of another labor of love.

Introduction

Game Maker has become an extremely popular tool for beginners wanting to take their first steps into game development. This is hardly surprising, as its intuitive drag-and-drop programming system makes it very easy to start creating your own simple games. It's also possible to make very professional-looking games using Game Maker, and you've probably already come across some impressive offerings made with it on the YoYo Games web site. Nonetheless, it's not always easy for users to see how they can make that leap into creating more professional games for themselves—and that is exactly what this book is for.

We're focusing on creating platform games this time around, but the development principles you'll learn are applicable to other genres as well. We've organized this book into parts based around the example games. In **Part 1**, you'll be introduced to *Fishpod*: nature's first ever platform game character, which we'll use as a way of introducing some of the main issues involved in creating platform games.

In **Part 2**, we're going back in time to visit a ninja of the Nth dimension, as we revive the 90s classic that is *Zool*. You'll learn how to create a slick, commercial-quality platform game almost entirely using drag-and-drop programming, and without using any features from the Pro edition of Game Maker. Nonetheless, the principles you'll glean are just as applicable to GML and will form the foundation of the platform game "engine" used in the remainder of the book.



Zool: One of the games you'll learn how to make in this book

In **Part 3**, you'll follow the development of a new game called *Shadows on Deck* from its initial concept design through to the implementation of a vertical slice of game play. In the process, you'll learn how to create compelling storylines and interesting characters to inhabit your games and keep your players absorbed while they play. You'll discover how the role of a concept artist feeds into the design of a game and gain access to the original animations and graphics that we've used to create the resources for the game.



Shadows on Deck: The final game you'll learn how to make in this book

To implement the game play, you'll see how you can convert the *Zool* engine into GML and enhance it in the process. We'll gradually transform the colorful playground of *Zool* into a sinister world inhabited by skeletal pirates and filled with fatal traps. We'll conclude by adding the puzzles and dialogue that give this platform-adventure its character.

Part 4 is something a little different, providing a handy resource for all those essential Game Maker features that are needed time and time again in order to create all sorts of different games. If you want to know how to add cheat codes, a countdown clock, or smoke trails to your game, then it's all here for you to find. Numerous different game features are covered in this step-by-step "how-to" guide for Game Maker.

We really hope that you enjoy this book and that it will be as well-received as its predecessor. If you read *The Game Maker's Apprentice*, then we hope that it kindled a passion for game development that will be sustained in this sequel. If you're an old-hat to Game Maker, but new to our books, then we hope that *The Game Maker's Companion* will help you to realize your own gaming visions. Either way, we're confident this book will continue to convey our passion for creating games, and impart some of our combined years of experience along the way.



PART 1



Fishpod

“There’s no point trying to run before you can walk,” as the old saying goes. This must have been particularly true for the first creature to crawl out of the primeval oceans onto dry land. It’s also true if you’ve never created a platform game before, so this prehistoric mudskipper will make the perfect companion as you take your first steps in platform game development.



Greetings, Game Maker

So here we are, about to embark on another journey into the world of game development. You may have joined us last time in *The Game Maker's Apprentice: Game Development for Beginners* (Apress, 2006), or perhaps you taught yourself the basics of Game Maker under your own steam. Either way, we invite you to dust off your trusty keyboard and loosen up your mouse-arm as you join us in *The Game Maker's Companion*.

The path ahead is an exciting one and we have a host of new challenges in store to enhance your skills as a game developer. Nonetheless, it would be foolish to undertake such a journey without making suitable preparations first. Each of you will bring your own unique skills to the journey ahead, but you won't get very far without some level of background knowledge. This chapter will equip you with that knowledge, so please make sure you are familiar with it before continuing. The majority of this information was covered in our first book, so this chapter simply summarizes the important facts as a reminder. It won't take long to cover the essentials, so let's make a start. This chapter will also serve as a handy reference if you need to check back on something later on in the book.

Resources

Video games are made up of different kinds of digital resources such as animations, sounds, music, and backgrounds. Game Maker lists all of its resources down the left-hand side of the main window (see Figure 1-1). These are grouped together into folders according to the different kinds of resources that Game Maker supports. You don't need to know every detail of every kind of resource, but you should be generally aware of what each type of resource is for.

- *Sprites*: Sprite resources are the digital images that you use to represent foreground objects in your games. Game Maker supports loading sprite images from `.bmp`, `.jpg`, and `.gif` file formats and now in Game Maker 8, `.png` and `.gmspr` as well. You can load animated images using the `.gif` and `.gmspr` formats, or by treating `.png` files as sequential strips of images by using `_stripXX` at the end of the file name (where `XX` = the number of frames in the image).
- *Sounds*: Sound resources include both sound effects and music for your games. Game Maker supports `.wav`, `.mid` (MIDI), and `.mp3` formats, but `.mp3` music can take up a lot of space and often contributes to the large size of finished games.
- *Backgrounds*: Background resources are digital images that you use to represent the background scene of your game. Backgrounds can only contain single images and Game Maker can load these images from `.bmp`, `.jpg`, `.gif`, and `.png` formats.

- *Paths*: Path resources contain a series of points that define a route for *object* resources to follow in the game. These can be either closed looping paths or open paths with a start and finish point.
- *Scripts*: Script resources contain programming instructions written in Game Maker Language (GML). GML provides a more advanced way of programming in Game Maker.
- *Fonts*: Font resources provide a means of displaying text in your game using the fonts installed on your machine. Game Maker grabs images of each character in your chosen font so that the player doesn't need to have the same font installed on their machine.
- *Timelines*: Timeline resources provide a way of triggering many different *actions* at specific points of time in your game (see Objects for more on actions).
- *Objects*: Object resources are the most important of all the resource types in Game Maker as they are used to represent all of the active components of your game. Objects can respond to *events* in the game by following a series of *actions* that you add to the event. In this way, you can program the desired behavior for all the different components of your game.
- *Rooms*: Room resources provide spaces for staging all the visible aspects of your game (levels, menus, cut scenes, and so forth) and contain all sorts of settings relating to backgrounds, views, and the game window. It also provides an editor for placing *instances* of objects into your rooms to determine their starting positions.

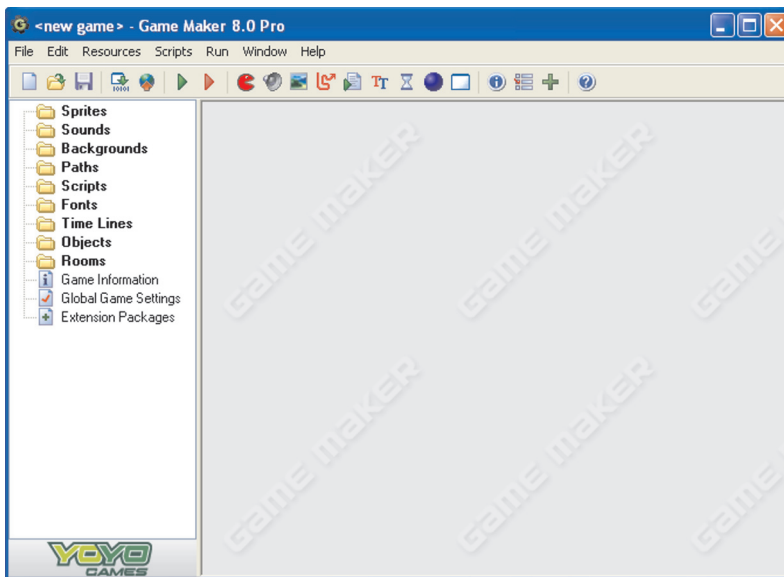


Figure 1-1. The main Game Maker interface with the resource list on the left

Instances and Objects

Objects are the programmable elements of your game and their behavior is directly determined by the *events* and *actions* that you choose to give them. However, there is an important distinction to be made between the *object resources*, which define the general behavior of objects, and the individual *object instances*, which occupy your game world. Once you have defined the behavior of an object resource, then you can place any number of instances of that object in your game. All these instances will behave in the same way (because they follow the same events and actions), but each has its own variables (position, speed, direction, and so forth) that are unique to that instance. If you like, you can think of object resources as being like jelly molds, and instances as the jellies you make with them. You only need one mold to make any number of jellies that have the same basic structure (see Figure 1–2).

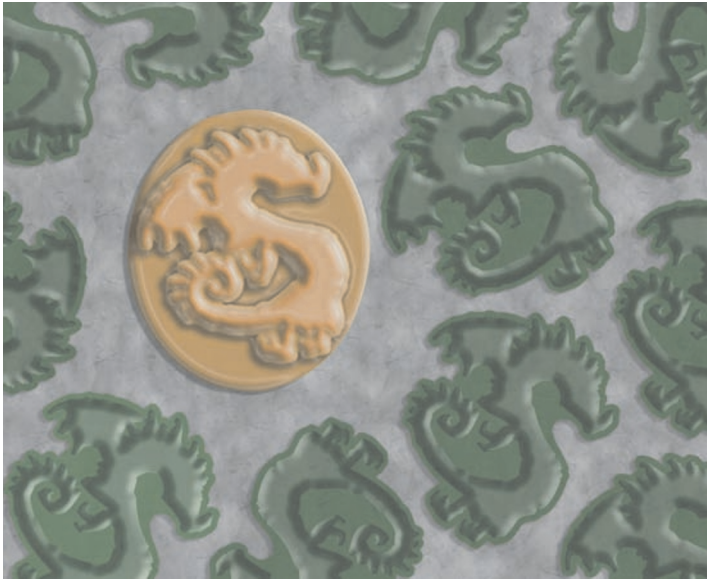


Figure 1–2. Object resources are like jelly molds and you can use them to create any number of object instances

Variables

A variable is a general programming term for something that can store information. In Game Maker, variables can either store a number (for example, 3, -12, 151.33) or some text (for example, *demons don't like dragons*). You use variables in Game Maker to store all the unique information about individual instances, such as their position on the screen, or their speed. So if you create a new *power* variable in the **Create** event of a dragon object, then every dragon object will have this variable, but each instance can have its own different value for *power*. It's also worth noting that you must assign variables a value (using a **Set Variable** action, for example) before you can use them (with a **Test Variable** action); otherwise, Game Maker will produce an error.