

# Introduction to the Second Edition

It has been four years since the release of the first edition of the book you now hold in your hands. It is interesting to reflect on what has changed in the industry during the intervening time, or, more specifically, what has not changed. In many ways we have seen a continuation of the trends that were well underway when this book was first written. Games continue to get bigger and prettier but not necessarily any more fun. Licenses have become more prevalent than ever, whether in the form of a movie tie-in or just having a quasi-famous personality attached to a project. The line between a computer game and a console game has become more and more blurred, with the largest games typically coming out on both, some under the same name but in different forms, but most providing almost exactly the same experience. In general, boldly original titles have become fewer and farther between.

A lot has happened to me since the first edition, and where appropriate I have woven that experience into this revision. The game that I was working on during the writing of the first edition, a western called *Gunslinger*, died out from under me. Though it had a number of problems, in the end it fell prey to the industry's more and more risk-averse nature. Following that, I managed to do quite a bit of work on *Drakan: The Ancients' Gates* and then developed *The Suffering* from conception through to localization. New examples from the practice of game development on *The Suffering* are integrated throughout this edition's chapters. Also, in addition to the *Atomic Sam* document that appeared in the original book, the complete design document for *The Suffering* has been included as an appendix. I sincerely hope this design document will be of particular interest to readers since it was used for a title that actually shipped.

Since the first book came out, two games have achieved greater popular success than anyone could have predicted. Those games are *The Sims* (which was analyzed in the first edition) and *Grand Theft Auto III* (which is analyzed in this new edition). In the intervening time, all of the game designers who were interviewed in the original edition completed new works in the industry, with five out of the six shipping new games, while Chris Crawford released two books. For this edition, I was fortunate enough to talk once again with most of these designers to update their interviews to reflect their most recent accomplishments (with the notable exception of Sid Meier, who as of this writing is busily trying to ship the new version of *Pirates!*). Also, the second edition gave me the opportunity to do an in-depth interview with a game designer I quoted extensively in the first book, Doug Church. Church is one of the most forward-looking designers working today, and I hope reading his thoughts prove inspirational for any designer.

As well as adding more examples from the games of the last four years, for the second edition I wanted to improve on what the book did well the first time, while filling in

a few of the gaps. Multi-player games have become significantly more prevalent since the first edition and were woefully underrepresented in the book before; now multi-player gaming is the subject of an entire chapter. Though the storytelling and artificial intelligence chapters are among the most expanded in the book, all of the chapters have been revised and updated significantly. Even the bibliography and glossary have been reworked and expanded.

When working on the second edition of *Game Design: Theory & Practice*, I revisited a lot of the feedback I received from the first edition, and did my best to address some of the concerns that were brought up. Nevertheless, I can say the views contained herein are still distinctly my own and represent my personal views on game development. Often my thoughts fall in line with the commonly held wisdom in the industry, but other times you will find I disagree with what everyone else seems to be doing. Who is right? No one is right, per se. In the creation of art there are no easy absolutes. As a game designer you need to balance going with the prevailing wisdom with what you feel in your heart. If you always make decisions based on popular opinion or on the flavor of the moment, you will always make average, predictable games. As a game designer, you should take what I say in this book, reflect on it, and decide where you stand and how you want to proceed on your own projects. It is my sincere hope that your views of game design end up substantially different from mine, so that when you make a game and I make a game we do not end up with exactly the same player experience. Variety, after all, is not only the spice of life, it is life.

One of the most frequent comments I heard about the first edition of the book was that it seemed dated. I would argue that it was not dated, merely that it attempted to look at game development over the entire history of the medium, not just the three years preceding the book's publication. The book contained examples and discussion of current games proportionate to classic games. Indeed, if I had focused more on what was current in the industry when I wrote it, the book might have seemed relevant on its release, but within a few years truly would have been horribly dated. If one looks at the first edition today, four years after it came out, one will find it is nearly just as relevant today as it was then. Thus, in making a new edition, I strove less to bring the book "up to date" and more to expand on what it was already doing. Yes, I've included references to newer games, since many great new games have come out since the book was first published, but I've kept just as many discussions of the classics from the last three decades. Anyone who has worked with me knows that, when in the heat of game development, I am as likely to pull inspiration from a game made in 1983 as a game made in 2003. I would argue that to be a great game designer, you need to understand the past just as well as the present. As a game designer, if you cannot see the value and lessons to be learned from a classic game made in 1983, then you have a long way to go before you truly understand our medium.

In truth, I have always seen this book as something of a history lesson for game developers and enthusiasts alike. In addition to the game analysis chapters, this especially comes through in the interviews, which I hope readers enjoy as much for what they tell us about game history as they do for their specific insights into game development. If a reader sees a reference in this book to a game that they are unfamiliar with, it is my hope that they might seek out that title in order to play it. Almost all the games I

refer to in this book are titles that I consider to be worth anyone's time to play. That said, a big problem for game historians and developers alike is that actually playing a game from twenty or even ten years ago can be quite difficult. If you are an aspiring filmmaker, tracking down almost all the cinema classics stretching back a hundred years is fairly easy. Not so with computer and video games. Emulators have done a lot to help this, but many games that are quite well known and respected are all but unplayable for most people because the systems they worked on no longer run, because the games themselves are out of print, or both. I believe that our ability to grow as an industry is directly proportional to our ability to understand our past: if we cannot understand it because we cannot play it, our evolution may well be stunted.

Throughout this book I discuss what I believe a game designer should think about when developing a game. I have found that one way to improve your game design methodology is to write a book about it. Though I might not recommend this technique to everyone (after all, the bookstores can only bear so many different volumes on the subject), I can testify that it can be quite helpful to take your nose off the grindstone every once in a while and think about games and their development a step or two removed from the day-to-day process of making it happen. I should warn that one unfortunate side effect to writing a book is having your coworkers point out to you whenever you are failing to follow one of the techniques you advocated in print. And therein lies the fundamental problem: regardless of how much you think about game design or try to do everything the best way possible, at the end of the day modern computer games are still incredibly hard to create. I certainly don't pretend to have all the answers, but my hope is that this book will make things a little bit easier, not just for me, but for you as well.

Richard Rouse III  
Seattle, Washington

# Introduction

My earliest recollection of playing a computer game was when I stumbled upon a half-height *Space Invaders* at a tiny Mexican restaurant in my hometown. I was perhaps six, and *Space Invaders* was certainly the most marvelous thing I had ever seen, at least next to LegoLand. I had heard of arcade games, but this was the first one I could actually play. *Space Invaders*, I knew, was better than television, because I could control the little ship at the bottom of the screen using the joystick and shoot the aliens myself instead of watching someone else do it. I was in love. The irony of this story is that, at the time, I failed to comprehend that I had to stick quarters into the game to make it work. The game was running in “attract” mode as arcade games do, and my young mind thought I was controlling the game with the joystick when I was actually not controlling anything. But the idea was still mind-blowing.

This book is about developing original computer games that will hopefully have the same mind-blowing effect on players that *Space Invaders* had on my young brain. This book deals with that development process from the point of view of the game designer. Many books have been written about the programming of computer games, but I can remember my frustration in being unable to find a book such as this one when I was an aspiring game designer. In some ways, I have written this book for myself, for the person I was a decade ago. I hope that other people interested in designing games will find this book informative. In my humble opinion, it is the game designer who has the most interesting role in the creation of a computer game. It is the game’s design that dictates the form and shape of the game’s gameplay, and this is the factor that differentiates our artistic medium from all others.

## What Is Gameplay?

I hear you asking, “But what is gameplay?” Many people think they know what gameplay is, and indeed there are many different reasonable definitions for it. But I have one definition that covers every use of the term you will find in this book. The gameplay is the component of computer games that is found in no other art form: interactivity. A game’s gameplay is the degree and nature of the interactivity that the game includes, i.e., how players are able to interact with the game-world and how that game-world reacts to the choices players make. In an action game such as *Centipede*, the gameplay is moving the shooter ship around the lower quadrant of the screen and shooting the enemies that attack relentlessly. In *SimCity*, the gameplay is laying out a city and observing the citizens that start to inhabit it. In *Doom*, the gameplay is running around a 3D world at high speed and shooting its extremely hostile inhabitants, gathering some keys along the way. In *San Francisco Rush*, the gameplay is steering a car down implausible tracks while jockeying for position with other racers. In *StarCraft*, the gameplay is maneuvering units around a map, finding resources and exploiting them, building up forces, and finally going head to head in combat with a similarly

equipped foe. And in *Civilization*, the gameplay is exploring the world, building a society from the ground up, discovering new technologies, and interacting with the other inhabitants of the world.

Though some might disagree with me, the gameplay does not include how the game-world is represented graphically or what game engine is used to render that world. Nor does it include the setting or story line of that game-world. These aesthetic and content considerations are elements computer games may share with other media; they are certainly not what differentiates games from those other media. Gameplay, remember, is what makes our art form unique.

## What Is Game Design?

What, then, is game design? Having defined what exactly I mean when I refer to gameplay, the notion of game design is quite easily explained: the game design is what determines the form of the gameplay. The game design determines what choices players will be able to make in the game-world and what ramifications those choices will have on the rest of the game. The game design determines what win or loss criteria the game may include, how the user will be able to control the game, and what information the game will communicate to him, and it establishes how hard the game will be. In short, the game design determines every detail of how the gameplay will function.

## Who Is a Game Designer?

By this point it should be obvious what a game designer does: he determines what the nature of the gameplay is by creating the game's design. The terms "game designer" and "game design" have been used in such a wide variety of contexts for so long that their meanings have become diluted and hard to pin down. Some seem to refer to game design as being synonymous with game development. These people refer to anyone working on a computer game, whether artist, programmer, or producer, as a game designer. I prefer a more specific definition, as I have outlined above: the game designer is the person who designs the game, who thereby establishes the shape and nature of the gameplay.

It is important to note some tasks in which the game designer may be involved. The game designer may do some concept sketches or create some of the art assets that are used in the game, but he does not have to do so. A game designer may write the script containing all of the dialog spoken by the characters in the game, but he does not have to do so. A game designer may contribute to the programming of the game or even be the lead programmer, but he does not have to do so. The game designer may design some or all of the game-world itself, building the levels of the game (if the project in question has levels to be built), but he does not have to do so. The game designer might be taking care of the project from a management and production standpoint, keeping a careful watch on the members of the team to see that they are all performing their tasks effectively and efficiently, but he does not have to do so. All someone needs to do in order to justifiably be called the game's designer is to establish the form of the game's

gameplay. Indeed, many game designers perform a wide variety of tasks on a project, but their central concern should always be the game design and the gameplay.

## What Is in This Book?

This book contains a breadth of information about game design, covering as many aspects as possible. Of course, no single book can be the definitive work on a particular art form. What this book certainly is not is a book about programming computer games. There are a wealth of books available to teach the reader how to program, and as I discuss later in this book, knowing how to program can be a great asset to game design. However, it is not a necessary component of designing a game; many fine designers do not know how to program at all.

The chapters in this book are divided into three categories. First are the thirteen core chapters, which discuss various aspects of the development of a computer game, from establishing the game's focus, to documenting the game's design, to establishing the game's mode of storytelling, to playtesting the near-final product. These chapters discuss the theory behind game design, and what a designer should strive for in order to create the best game possible. The chapters also include discussions of the reality of game development, using examples from my own experience, to delve into the actual practice of game design.

There are six analysis chapters included in this book, covering six excellent games in six different genres. One of the most important skills a game designer must have is the ability to analyze games that he enjoys in order to understand what those games do well. By understanding these other games, the designer may then attempt to replicate those same qualities in his own projects. That is not to suggest that good game designers merely copy the work of other game designers. Understanding the reasons why other games succeed will bring the designer a more complete understanding of game design as a whole. Every game designer should take the games that he finds most compelling and try to examine what makes them tick. The examples I include in this book, *Centipede*, *Tetris*, *Loom*, *Myth: The Fallen Lords*, *The Sims*, and *Grand Theft Auto III*, are all very unique games. And though a given project you are working on may not be similar to any of these games, a lot can be learned from analyzing games of any sort. First-person shooter designers have had great success in revitalizing their genre by looking at adventure games. Certainly, role-playing game designers have recently learned a lot from arcade game designers. *Grand Theft Auto III* improved over its predecessors by cribbing from racing games. Melding in techniques from other genres is the best way to advance the genre you are working on and to create something truly original.

This book also includes a group of interviews with seven of the most well-respected game designers of the industry's short history who have designed some of the best games ever released. These are lengthy interviews that go deeper than the short press kit style interviews one finds on the Internet or in most magazines. In each interview the subject discusses the best titles of his career and why he believes they turned out as well as they did. The designers also talk at length about their own techniques for developing games. Throughout my own career in game development, I

have found interviews with other computer game designers to be exceedingly helpful in learning how to perfect my craft. There is much information to be gleaned from these chapters, ideas that can help any game designer, regardless of how experienced he may be.

At the end of the book you will find a glossary. Though it is far from a complete listing of game design terminology, it does cover many of the more esoteric terms I use in the book, such as a personal favorite of mine, “surrogate.” Every game designer has a set of jargon he uses to refer to various aspects of his craft, and this jargon is seldom the same from one designer to the next. If nothing else, the glossary should help you to understand my own jargon. For instance, it will tell you the difference between gameplay and game mechanics. Furthermore, readers who may find the content of this book to assume too much knowledge may find the glossary helpful in sorting out what an RTS game is and what the two different meanings for FPS are. Often, discussions of game design can degrade into questions of semantics, with no two sides ever meaning exactly the same thing when they refer to a game’s “engine.” I hope that the glossary will help readers to avoid that problem with this book.

## ***Who This Book Is For***

This book is for anyone who wants to understand the computer game development process better from a strictly game design standpoint. As I stated earlier, there are plenty of books available to teach you how to program, or how to use Photoshop and 3D Studio MAX. This book will do neither of these things. Instead it focuses on the more elusive topic of game design and how you can ensure that your title has the best gameplay possible. Though solid programming and art are both central to a game’s success, no amount of flashy graphics or cutting-edge coding will make up for lackluster game design. In the end, it is the gameplay that will make or break a project.

I have written this book in such a way as to encompass projects of different scopes and sizes. It does not matter if the game you are working on is destined for commercial release, if you hope to someday release it as shareware, or if you are only making a game for you and your friends to play; this book should be helpful to a game designer working in any of those circumstances. Furthermore, it does not matter if you are working on the game with a large team, with only a few accomplices, or going completely solo. In the book I often make reference to the “staff” of your project. When I refer to “your programming staff” I may be referring to a team of ten seasoned coders commanding massive salaries and pushing the boundaries of real-time 3D technology, or I may be referring to just you, coding up every last aspect of the game yourself. When I refer to “your playtesting staff” I may be referring to an experienced and thoroughly professional testing staff of fifteen who will pride themselves on giving your game a thorough going-over, or I may be referring to your cousins Bob and Judith who, like you, enjoy games and would love to play what you have made. Good games certainly do not always come from the biggest teams. Even today, when multimillion-dollar budgets are the norm, the best games still often result from the vision and determination of a lone individual, and he need not always surround himself with a massive team to see that vision through to completion.

Many places in this book make reference to you leading the design on the project on which you are working. Of course, not every designer can be in the lead position on every project, and even if you are the lead, you will often find yourself without the absolute final say on what takes place in the game. In this regard, this book is written from a somewhat idealistic point of view. But regardless of how much authority you actually have over the direction of the project, the important point is to always know what you would do with the project if you could do whatever you wanted. Then you should campaign for this direction with the other people on the team. If you are persuasive enough and if you are, in fact, correct in your instincts, you have a good chance of convincing them to do it your way. Projects are often led not by the people with the most seniority or who have the right title on their business card; projects are led by the people who “show up” to the task, who care about their projects and are committed to them, and who are willing to put in the time and effort to make the game the best it can be.

## *Theory and Practice*

Every medium has a unique voice with which it can speak, and it is the responsibility of the user of a medium to find that voice. Computer games have a voice that I firmly believe to be as strong as that available in any other media. Computer games are a relatively young form when compared with the likes of the printed word, music, the visual arts, or the theater, and I think this currently works against the likelihood of computer games truly finding their most powerful voice. This book is an attempt to help readers find that voice in their own projects. This can come in both the more theoretical form of questioning why it is that players play games, but also in the entirely more practical form of how to most effectively work with playtesters. To have any chance of producing a great game, the game designer must understand both the theoretical aspects and the practical necessities of game design.