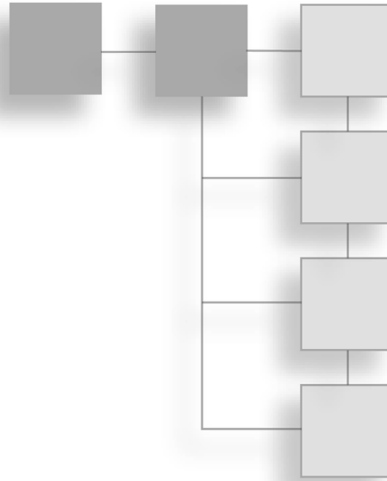


CHAPTER 1

THE BASICS OF GAME DESIGN



Building games is one of the most challenging and rewarding experiences I can think of. Taking pure imagination and making it come alive is absolutely addictive—a creative process so immersive and consuming that you’ll start craving it when you haven’t done it for too long. Some people think the fun is in playing the game, but, for a few special people, creating and building the environments in which other people play causes mere gameplaying to pale in comparison. If you’re reading this book, you are probably one of those special people who have that compulsion to create, and, with your creation, entertain.

Designing your game is the first step on your journey toward bringing your dream to life. Remember, many designers have come before you and failed to deliver. The game design world is like an iceberg: Only a small number of successes have peaked above the frigid water to shine in the sun. These successes are what happens when a great design meets a great team. The rest lurk in an underwater graveyard, rotting slowly in the company of a million other badly designed failures.

To avoid this watery fate, you’ll have to be smart, imaginative, tenacious, and driven. You’ll need to take a look at those successes and pick them apart like a scavenger bird, ripping out their guts to learn how they managed to get on top of the heap. You can learn from the failures as well, stripping them of their once-bright promises and glinting hype to peer at their ugly, ill-conceived gameplay so you can say to yourself, “I will not follow this path!”

2 Chapter 1 ■ The Basics of Game Design

In this chapter, you'll learn the following:

- The basic knowledge you need to begin designing a game
- How to empower the player
- System design
- Different forms of challenges and how to build them
- Pacing and flow
- The beginning, middle, and ending of a game
- Some tips on how to make your games better

The “Fun”amentals

Making games can be a humongous power trip. Having the ability to create what can amount to rat mazes for humans can lead some designers to grow egos the size of a large continent. They lose sight of the core fundamental, which is that games are about one thing: *entertaining people*. This is the first and most important thing to think about when you're making any kind of game, whether it's a teensy mod or a huge, 250-hour RPG. In making a game, you become an entertainer, not a puppet master bent on world domination. As such, your primary concern should be the happiness of your audience and not satisfying your unfulfilled need to punish those who annoy you. You have to make your game fun.

Fun

Fun is the first thing people think about when they hear the word “game.” Fun is a simple word, easy to spell, and everyone agrees on what it means. However, the things that people consider fun are as individual as fingerprints. Some people might like hang-gliding, some enjoy going to the mall, some enjoy watching sports, and some enjoy data-entry jobs. Although two people might agree that something is fun, if you get a group of 10 people together, you'll start having problems.

Games are supposed to be fun. People expect them to be sources of entertainment and delight, a source of diversion to distract them from a less-than-perfect existence. The game industry employs thousands of testers and spends millions of dollars a year in market research, trying to determine what people think is fun. So far, no one has really narrowed it down enough to create a magical “fun” formula that guarantees success time after time.

As a future level designer, you'll want to make your levels fun. Although you might not be able to please everybody, there are some ways to hedge your bets.

Know Your Audience

Unless you’re making mods that only you are going to play, you’ll be making your game for other people. These people will have definite opinions as to what is and isn’t fun, and they’ll completely pass you over if you don’t consider those opinions when making your game. Knowing your audience can be an easy task if you’re making a game that isn’t exceptionally innovative, such as a first-person shooter (FPS) or a real-time strategy game (RTS). The further you get from the accepted genres, the harder it will be to find your audience. There are exceptions, of course. Sometimes companies create a genre out of whole cloth, much like Maxis did with their wildly successful game *The Sims*.

To *know* your audience, you have to *find* them. Again, it can be pretty simple to find your audience if you’re making a game that belongs to an established genre, especially if that genre has an online multiplayer component to it. You can frequent Internet message boards and chat rooms dedicated to games similar to the type of game you want to design to see the opinions of people who play the games like the one you want to create.

Another good place to find people talking about what they like and dislike about games are game review sites and magazines, like Gamespy.com and *Computer Gaming World* magazine. One site that I’d recommend is Gamerankings.com. It’s a portal site that gathers links to all kinds of game reviews. You’ll be able to find as many opinions on what’s good and what’s bad as you can handle.

Once you find your audience, pay attention to what they like and what they don’t like. This will give you tremendous insight into what to do and what *not* to do when designing your game.

A word of warning: As you start looking for opinions on message boards and chats, remember Sturgeon’s law: 99% of everything is crap. For many, the only reason to write anything about a game, positive or negative, is because they have very strong feelings about it. They might not be looking at the game in the most balanced way. A lot of game reviewers can also let their feelings get away from them. Remember, these people are trying to describe why a game is or isn’t “fun,” and “fun” is a slippery thing to define. Always keep your own counsel, and when you read something that seems highly emotional, try to get what you can from it and move on to the next opinion. Remember, you’re trying to make a game that *many* people will enjoy, not just one or two.

4 Chapter 1 ■ The Basics of Game Design

Know Your Genre

Just as it's important to know who your audience is, you need to know the games that your own game will be competing with. Not just so you don't unconsciously copy another game developer's work, but to learn what players *expect* from your genre.

Tip

What players expect from your game is perhaps the deciding factor in whether it will be a success or a failure. If you meet the players' expectations, or even exceed them (in a positive way, of course), your game will be a hit. If you fail to meet the players' expectations, well... Welcome to Nowheresville, baby. Population: You.

Expectations are usually generated well before players pick up your game. They'll be influenced by the scanty information you provide on your Web site, the possibly false information generated in online or magazine previews, any marketing you may do, the box your game comes in, and even the name of your game. And, most annoyingly, they'll be influenced by pure conjecture generated by word of mouth. The more your audience's expectations get out of hand, the more disappointed they'll be when they find out that your game *doesn't* actually allow them to match Captain Kirk against Darth Vader in a duel to the death.

It's important to know your genre, and what that genre has given its fans so far. Consider first-person shooters (FPS) games on the PC. Currently, every single FPS uses the W, A, S, and D keys for major movement control. The W key moves you forward, the S key moves you backward, and the A and D keys strafe, keeping you facing forward while moving side-to-side like a crab. Players now expect that key configuration when they sit down in front of any new FPS, and woe to the plucky game company that tries to do it "a better way."

When you're making your game, you need to find all these standardizations that have become associated with your chosen genre. It's not just control configurations, either. A boss at the end of each level is a cliché that a lot of players expect. In an RTS, starting a level near needed resources is expected.

You also need to know the taboos. Jumping puzzles aren't very popular in first-person shooters. Random disasters aren't appreciated in *any* game. Each genre and even each console and the PC have their own "thou shalt nots" associated with them. PC gamers, for instance, hate save points, and like to save anywhere. Console players don't mind as much. The white and black buttons on the XBox controller are hard to get at. By studying games and reading reviews, you can get a good idea of what drives players crazy and what they like.

You should also know your genre well enough to know what sorts of things it could do better. Although some of the mechanics may be set in stone, others might be more pliable. If you can find and improve the things that need improving, or change the things that won't alienate the player, you're on the way to making a great game.

Know Yourself

This may sound a little philosophical, but in order to make a fun game, you truly need to know *yourself*. Or at least you need to know what you think is fun about games. You'll never truly know your audience enough to predict what every single one of them will think is fun. However, you do know what *you* think is fun. When playing a game, whether yours or someone else's, try to notice when you are having a good time. If you can pause, do so and ask yourself what you just did that caused that big smile on your face.

The next step is figuring out why what happened was so fun. Is it because of the way your character moves? Is it because of the rewards you're getting? The victories you're achieving? The cool dialogue? The other players in the game? The intriguing puzzles? The challenge of it all?

You need to find that root, identify it, and really look at it hard. Then, you need to figure out how to implement it in your own game. If you can do this, you're ahead of the pack. Many people can't tell why they're having fun, and if you quiz them about it, they'll give you fairly vague answers that can change each time you ask them. Knowing yourself, and being able to objectively identify the core reason why you feel that a game feature is or isn't fun, is essential to making fun games for other people.

Empowering the Player

Tim Schafer, the designer behind such games as *Grim Fandango* and *Full Throttle*, once noted that all games are about wish fulfillment. When you play a game, you're putting yourself into a fictional scenario that you wish you could experience in real life, at least in general terms. You can be a mighty general in chess, a tough, sarcastic biker in *Full Throttle*, or a powerful dwarven paladin in Blizzard's *World of Warcraft* game.

This is a good point. When you design a game, you want to immerse the player in a role that he thinks is fun and cool. As they say about writing good fiction: “Take me to a place I've never been, make me something I could never be, and let me do things I could never do.”

However, I like to boil this down a little more than that. I think that the root of fun in most games has to do with power. When a player feels empowered, achieves some level of competence that was formerly beyond him, that's when he starts having fun.

6 Chapter 1 ■ The Basics of Game Design

Empowering the player is pretty easy to do in modern video games. In fact, it's hard not to give the player *too much* power! You can give him super-strength, armies of crack soldiers to command, or even power over life and death itself. He can survive deadly ninja attacks, falls from great heights, or scathing verbal assaults from salty pirates. Game developers can create any conceivable world and make the player its god.

For some players, being a god is the pinnacle of fun. For others, just being a tad more competent than they are in real life brings the most enjoyment. Once again, this goes back to knowing your player and your genre. Knowing how to properly balance your game so that the player has as much of a challenge as he wants, without making it too easy or too hard, is one of the many balancing acts you'll have to face.

A Small Lesson on the Nature of Power

Power, by definition, is the ability or official capacity to exercise control. By understanding the nature of power, and which types of power appeal to which types of players, you can begin to fine-tune your game design technique.

There are three types of power: creative, destructive, and manipulative.

Creative Power

Having creative power allows you to bring something into existence that wasn't there previously, usually by combining separate, already existing objects or concepts. You can create a chair, a meal, or a relationship. Building games, like *Sim City* and *RollerCoaster Tycoon*, focus a lot on this type of power.

Many hobbies and professions revolve around this type of power, from model building and painting to manufacturing cars and game design. Creative power brings with it a sense of accomplishment that is extremely rewarding. Games that focus on creative power are generally considered “toy” types of games because they are more about play than about competition.

Most creative games have two aspects to them, a building aspect and a reward aspect. The building aspect usually concerns itself with giving the player a toolset that allows the player to create whatever he wants given his building materials. An example would be a building game based on Lincoln Logs. The player can use the toolset to create buildings and constructs out of an endless supply of virtual Lincoln Logs: giant log skyscrapers, log statues of famous rock stars, log museums, etc. The reward aspect of the game would issue challenges to the player, such as “build a log International House of Pancakes that seats 100 log citizens with as few logs as possible” and rewards him when the challenge is completed.

There is a large, vocal audience out there that loves creating. However, this type of gameplay is usually complex and time consuming, and can turn off players who want instant gratification.

Destructive Power

Destructive power is the ability to uncreate or radically alter the state of something until it no longer resembles its original form. You can destroy just about anything: civilizations, rhinos, or ideas. Games like *Serious Sam*, *Space Invaders*, and other shooter-style games primarily focus on destruction.

Games that center on destruction are the most satisfying in an immediate sense, and thus are the quickest to empower. Destruction, at least in western culture, is also associated with winning, as this quote from the movie *Apocalypse Now* suggests:

Robert Duvall, *Apocalypse Now* (1979): *You smell that? Do you smell that? Napalm, son. Nothing else in the world smells like that. I love the smell of napalm in the morning. You know, one time we had a hill bombed, for twelve hours. When it was all over I walked up. We didn't find one of 'em, not one stinkin' body. The smell, you know that gasoline smell, the whole hill. Smelled like... victory. Someday this war's gonna end...*

Erasing all existence of the things and people that annoy and anger in order to “win” drives the nihilistic pleasure that players find in games of destruction and anarchy. Probably not something we want to remind ourselves of very often, but it's a part of human nature that readily becomes apparent when given large weapons and a sense of unaccountability.

Games that focus on this type of power also have the quickest gameplay, with the shallowest learning curve. Players can boot up the game, learn the mechanics, and accomplish something very quickly, making this type of game the friendliest for people with a limited amount of time or short attention spans.

Manipulative Power

Manipulative power allows you to control other things. (It could be argued that this is the only real type of power, but for this discussion, the three types make more sense.) Manipulative power is present in all games. A player can control armies in *Command & Conquer*, control how Lara Croft moves in *Tomb Raider*, or control the falling blocks in *Tetris*.

Manipulative power is the most subtle power, and its correct use rewards the player by making him feel clever and proud of that cleverness. Giving the player the power to

8 Chapter 1 ■ The Basics of Game Design

manipulate also allows the player to immerse himself in your game, as your game characters become his extensions into your game world.

Games centering on manipulative power usually require the most thought, and can be incredibly complex. Depending on their complexity, they can be short or long experiences: a game of *Tetris* can be short, but a game of *Civilization* can take quite a long time.

The Flow of Power

To be complete, we should also consider the flow of power. In any contest, there comes a point where you have power over your opponent, or your opponent has power over you. If I jump from a step stool, I have the ability to land safely. I've triumphed over the adverse effects of falling. If I jump from a cliff, it's more likely that the adverse effects of falling will overcome me. In the game *City of Heroes* (shown in Figure 1.1), I can jump from any height without killing my character (although he does take quite a bit of damage).

All this is part of the complex web of interrelationships between different power systems in a game. The ability to create, destroy, and manipulate often appear in the same game. Each ability can interact with the others, creating interlocking systems.



Figure 1.1 In Cryptic Studio's *City of Heroes*, players triumph over even the irresistible force of gravity!

System Design

When game systems interact with one another, they create other systems of gameplay, sometimes unintentionally. We call this process *emergent gameplay*, as new systems emerge from the ways the old ones combine with each other.

Using this to your advantage is one of the hardest jobs of the game designer. It requires a lot of thought and knowledge to balance systems so that they work well together without creating powerful loopholes or discrepancies.

For level designers, a lot of the game systems will be in place by the time you get to your toolset. Infliction of damage should be balanced with the characters' resistance to damage. The physics of the game, and how they affect the player, should be there as well. However, when you're starting from scratch, you have to think about these things carefully.

Let's say that you decide to make a game where the player can move crates around by pushing them. You want these crates to react to gravity, so they fall when there's no floor under them. Sticking with this idea, you decide that falling crates can cause damage to players and their foes. Suddenly, you've created a network of interlocking systems that allows a player to push crates from heights on top of unsuspecting foes and kill them.

Now, every time you place a crate within the game, you have to consider whether a player can push it to a place where it can be used to overcome a challenge that you intended to be much more difficult than pushing a box over a ledge. You also have to figure out what to do if the player drops the crate on a friend, or on top of some non-player character (NPC) that the player needed to talk to in order to get to the next level.

These sorts of interconnecting systems bring the flavor of real life to a game. However, it's very hard to predict when the player can use these systems to avoid the gameplay you've set out for them.

Gameplay

Gameplay is a catchall word for whatever the player does with your game that's fun. In *Unreal Tournament*, the gameplay is running around and wasting as many competitors as possible. The gameplay in *Warcraft 3* is controlling troops and defeating enemies. There are as many flavors of gameplay as there are games. When you're creating your game, you'll have to identify what your gameplay is and make it as fun as possible.

Challenges

The challenge is usually the central hub of the gameplay. The objective, and the barriers that prevent the player from achieving that objective, are what determine the challenge. By identifying your challenge, you can boil it down until it's pure.

Here are some of the standard challenges:

- **Time Challenge:** The player is allowed only a certain amount of time to complete a task. This is one of the oldest challenges, and in modern games it's usually combined with some other challenge. A simple example is a race that must be run within a certain time. *WarioWare* uses time challenges in every single mini-game it presents the player.
- **Dexterity Challenge:** The player must accomplish some sort of feat that requires dexterity. In modern games, a dexterity challenge might be shooting a target with a pistol. It doesn't need to be about physical dexterity, though. It could be a mental challenge, where the player has to make quick decisions in order to overcome the obstacles he faces.
- **Endurance Challenge:** Sort of the opposite of a timed challenge. Instead of having a limited amount of time to complete a task, an endurance challenge tests how far the player can go before he falters. Older arcade games like *Defender* and *Pac-Man* were endurance challenges.
- **Memory/Knowledge Challenge:** This type of challenge requires the player to know certain facts in order to win. Game shows like *Jeopardy* present this kind of challenge. In video games, usually it means teaching the player some fact, like "baboons really like barbecue chicken pizza," and then making him recall that fact later on in the game, like using a piece of pizza to lure a baboon guard away from the door to the treasure trove of the Baboon God. Other examples include making the player memorize certain button patterns on the controller to execute combination attacks, remember his way through mazes and difficult terrain, or remember which types of keys work in certain types of locks.
- **Cleverness/Logic Challenge:** Somewhat like the knowledge challenge, the cleverness challenge requires the player to figure out a puzzle without having the answer beforehand. An example would be trying to figure out what combination of buttons to press to open a door. Games like *Tomb Raider* and the *Indiana Jones* series include cleverness puzzles.
- **Resource Control Challenge:** Many games use resource control as the challenge. The player is given a certain amount of a resource. He must use that resource to overcome an objective before it runs out. Strategy games like checkers, chess, and *Warcraft* (shown in Figure 1.2) have finite resources that the player must use to win the game.



Figure 1.2 Blizzard Studio's *Warcraft 3* uses resource challenges to make a compelling real-time strategy game.

Designing Challenges

In most games, challenges are combined to make more complex gameplay. A timing challenge can be combined with a dexterity challenge to create a racing game like *Gran Turismo*. A resource control challenge can be combined with a knowledge challenge to create a game like *Scrabble*. Challenges can be combined in just about any configuration.

As always, when you're designing a challenge, let the game's genre be your guide. What kinds of challenges do you find in similar games? If you're a beginner, re-creating the puzzles and challenges that you've seen in other games can be a great learning experience. If you're an advanced designer, you need to know where the bar is set and what you have to surpass in order to be competitive and not derivative.

Be inventive with your challenges. By deconstructing challenges down to their basic parts and then reassembling them in new and different ways, you can make something unique. However, as mentioned before, you shouldn't stray too far from your genre's roots. Putting a text adventure-style puzzle into a deathmatch arena will confuse the player and make it hard for him to immerse himself fully into your game.

12 Chapter 1 ■ The Basics of Game Design

Also keep in mind the setting of your game. If you're creating a jungle-based game, the challenges need to fit the jungle theme. Although this can lead to clichés, with all your puzzles seemingly created by the Professor from *Gilligan's Island*, you always want to make the player forget he's playing a game and let him immerse himself as fully as he can.

Another aspect you'll have to keep in mind is the interface of the game. If you're building an FPS and suddenly you want to add RTS-style challenges, such as the player controlling groups of other units to achieve a goal, you'll probably have to include a lot of interface elements that may break the original gameplay you wanted.

For instance, if you want the player to be able to control other units, you'll have to put in functionality for selecting friendly units. You'll need controls that let the player do everything with those units that they could do in a normal RTS, because people have come to expect that level of functionality in an RTS game. The player must be able to tell those units to go to a certain place, stop where they are, attack, stop attacking, or not attack at all. You'll also need controls for selecting enemies, so the player can tell the controlled units who to attack. FPS games usually don't have a good range of sight (the game character, being on his feet, usually can't see very far away), so you may want to allow the player to disassociate himself from his character and see where he wants to send his units. Suddenly, with all of these mechanics, your player can now see into areas that were supposed to be a surprise. Your programmers are cursing your name and have added an extra year to the development time. Your design becomes less and less like the thing you originally built.

Keeping the player's challenges within the scope of your original interface will help you avoid these problems.

The placement of challenges is also very important. You want the player to not only see the challenge, but also understand it enough to know the first step in solving it. You definitely don't want your challenge to kill the player before he knows it's even there.

An example of a bad challenge would be a camouflaged pit trap that the player can't see until he falls into it. This leads to a game where the player learns by dying, where he knows the dangers he faces only after they've killed him. Another example of a bad challenge would be a door-opening puzzle where the player must press a button completely out of sight of the door it opens. If the player can't establish a mental link between the puzzle and the solution, he's likely to become frustrated and quit playing your game.

Challenges should always be beatable. This seems obvious, but at times you can inadvertently create challenges that *aren't* beatable. If the player must possess a certain object in order to overcome a challenge, and the object becomes inaccessible, the challenge becomes impossible.

For example, let's say that the player controls the rough and ready adventurer Dirk Badsneeze. In level 2, you've placed a key in a fairly out-of-the-way place, on top of a counter. The player breezes right past the key in level 2 and proceeds to level 3, where he's confronted by a locked door. He needs the key from level 2, which is now inaccessible. Before you can say "gesundheit," Badsneeze is stuck and the player has to restart your game.

Another problem is when, in the process of playing your own game repeatedly, you create a challenge that requires an expertise beyond the level of the average player. This is pretty common because designers and testers, having played the game so many times before, start seeing it as too easy and ratchet up the difficulty of the puzzles. However, when a first-time player tries his hand at the puzzle, it's too difficult and he quits. This effect can also spread to your game's story. The designer may start forgetting that the player doesn't know as much about the game world as he does, so he starts leaving out small bits of information that the player needs in order to make sense of it all.

Challenges, more than anything else in your game, need to be perfected. You'll never create a great challenge on the first try. It takes a lot of tweaking, or iteration, to get it right. Once you've created a challenge, run through it again and again to make sure everything goes off without a hitch. Then, give it to other people and watch how they deal with it. Always try to make your challenges as solid, understandable, and fun as possible.

Once you've created all your challenges, you need to figure out where to place them. In some games, the challenge is constant. *Tetris*, *Space Invaders*, and most other arcade games have constant challenges. More recent games, especially those that revolve around a story or a variety of different goals, have their challenges spaced apart. The player goes from one challenge to the next, resting in-between. This pattern of challenge/rest/challenge determines the pacing and flow of your game.

Pacing and Flow

The designer usually looks at the game as a whole and doesn't think too much about the individual challenges the player will face on a minute-by-minute basis. However, as a level designer, you'll be concerned about what the player will face every moment he spends in your level.

Game pacing uses challenges and breaks to establish a rhythm and tempo. A well-paced game has a flow to it; tension and relaxation follow one another to draw the player through the game.

14 Chapter 1 ■ The Basics of Game Design

This technique is widely understood in the realm of filmmaking, and it's perfectly applicable in game design as well. Action leads to excitement, which isn't sustainable for a very long time. You need to give the player a chance to cool off. A lull after each challenge allows the player to take a break, marshal his resources, and go into the fray once again.

In some games, this rhythm is easy to establish. In older arcade games like *Defender* and *Pac-Man*, the player gets about 5 minutes of gameplay to complete a level, and then a short rest as the next level is loaded. This little break was just as important to making these games compelling as the actual gameplay. In a lot of modern games, levels take a bit longer than 5 minutes, so rest breaks have to be included within each level. In single-player first-person shooters, this break is usually accomplished by not respawning enemies in a certain location. After the player has finished defeating all the enemies in one location, he can remain there and feel somewhat safe as he unwinds briefly. Then he can move to the next encounter. In platform games, the end point of a difficult puzzle or challenge is usually a safe zone where the player can rest for a minute.

Some games, though, use different means to create a rhythm. Most real-time strategy games don't have a rest component until the end of each level, which can sometimes take over an hour! The pacing is in the rhythm of managing each battle and then maintaining the growth of your forces and moving them into position for the next fray. In games like this, having a pause function can help the player generate his own rhythm. However, pausing can also disrupt the player's immersion in the game and make it hard to get back into it.

First Impressions: In the Beginning

The most important part of any game is the first 10 minutes. Unless you make them so compelling that the player can't put down the controls, he'll find it very easy to quit and never touch your game again. After all, he has very little emotional investment and a plethora of other games to choose from.

A good first impression is a pretty tough thing to accomplish. For one thing, whenever a player starts a new game, he enters a new universe. He only has a vague idea of how to do anything, from walking to putting on his clothes to opening a door. This makes the player feel dumb. Remember how the player wants to feel empowered? In the first 10 minutes of a game, even if the player has the power of a god, he'll feel like the biggest idiot on the planet. No one likes that feeling, so you need to make this breaking-in period as quick as possible.

There are a couple of things in your favor. It's likely that your player has played other games, so he'll know which way to hold the controller. If your game uses controls similar

to other games in its genre, the player will feel almost comfortable right from the start. If your game is a sequel to or a mod of a game the player has played before, he probably won't feel dumb in the least. This is why sequels are so popular. Players don't have to invest any time in learning a new interface.

The player is also likely to know a little about the setting of your game. If it's sold in a retail store like Electronics Boutique, the player was probably attracted to the packaging. Or maybe he's read some reviews or a brief blurb on a download site, or a friend has recommended the game to him. Whatever makes the player purchase or download your game will also help him get through that deadly first 10 minutes.

However, all this is hardly sufficient if the player still feels stupid or frustrated. As the game designer, you have to teach the player about his new environment, while entertaining him enough to keep him playing.

Most games start with an introductory movie. Although the player doesn't get to interact with this movie, it introduces him to the gameworld and his character's place in it. Generally, this introductory movie is the best-looking thing in the game. The developers know that they need to capture the player, so they throw the whole works at him: flashy explosions, beautiful vistas, scantily clad women, etc. They want the player to think that the rest of the game will be this sweeping and epic.

The introductory movie explains the setting of the game and shows off some of the gameplay that the player will experience, whether it's racing cars, battling aliens, playing a sport, or placing blocks on top of one another. It briefly lays out the story of the game, giving the player enough knowledge that he doesn't feel completely confused when he finally gets to play.

After the introductory movie is over and the player gets to start playing, the designer's next step is to start feeding information to the player. You want him to feel confident about playing this game with something resembling competence.

Note

At this point, you may be thinking, "What about the game manual? Why doesn't the player just read the manual that comes with the game, so I don't have to waste time on expensive CGI movies or training levels?"

Unfortunately, today's players generally don't read manuals. For one thing, they aren't very entertaining. When a player picks up a game, he wants to be entertained as soon as possible, so he skips the boring stuff and jumps right into the gameplay. Also, manuals these days are expected to fit inside a DVD box, which limits them to 15 or 16 pages. Even if you do want to provide an entertaining manual, you'll have a bit of a problem fitting a funny bon mot between the necessary pictures of the controller and the credits pages. (You *do* want your name in the manual, don't you?)

16 Chapter 1 ■ The Basics of Game Design

In most games, the solution is a thinly veiled training session. This usually takes the form of a few very basic encounters where the player is taken from simple lesson to simple lesson. Teach him a skill, let him play with it for a bit, and then take him to the next lesson. Teach him what various UI elements do (such as status bars), how to use the UI, and how to use the different menus and what the options on each menu do. Then teach him how to interact with the game, such as combat strategies or the core gameplay itself.

This is actually a powerful, efficient way to teach. However, since you're in that dreaded first 10 minutes, your lessons also have to be *fun*. In many story-based games, these lessons are taught by a non-player character, someone who speaks clearly and is amusing. That way the player is entertained while he practices using the A button to open doors.

During this training, especially in story-based games, you may want to shove a bunch of exposition down the player's throat. "As you know, the Elder Muskrats besieged the town of Dryer in 1528, which was when your grandfather, who was Dryer's greatest smith, created the Tongs of Eldwere when he found getting hot dogs out of boiling water to be too painful..." *Bad idea*. Try to avoid doing too much exposition at once.

If you can make all of this training fun and transparent, those first 10 minutes will fly by. The player will think, "I am *awesome* at this game!" He'll feel like he's discovered some sort of hidden talent he never knew he had until he started playing *your* game.

In the Middle

The middle of your game will contain the bulk of your gameplay. This is where you put all the levels, plot twists, cool items, and everything else you want to stuff into your game.

The most important two concepts for the middle of your game are consistency and growth. Your game world, whether it's a boxing game or a kart racer, has to have an internal consistency. If the player can pick up one vase, he expects to be able to pick up all vases from that point on. Even if the physical laws of your game are nothing like real life—let's say that cats are now frictionless and dogs are superconductors—once the player immerses himself in your world, he'll be very disappointed if he suddenly finds a cat he can't use as a hockey puck.

Consistency is very hard to maintain when you decide to balance your game. For instance, if you're finding it too simple to get through certain parts of your game, you might be tempted to adjust how much damage the enemy's weapons inflict, or to make them harder to hurt.

Even if the change is subtle, players will notice immediately. They might not count how many laser blasts it takes to destroy a Snarg Interceptor, but they'll feel a sense of disempowerment when it suddenly takes four instead of three.

Growth is the second most important concept in most games. As the player progresses through your game, he'll get better at playing it. In today's games, the standard method of keeping the player from getting bored is to ramp up the level of challenge incrementally, keeping him on his toes as he faces greater and greater adversity.

In many games, growth in the player's abilities is aided artificially as the player's character grows more powerful. In many FPS games, the player gets access to more and more powerful weapons and armor as the game goes on. RPGs do this as well, and they also allow character growth, making the character progressively stronger and more resistant to damage. RTS games give access to new units, fighting games unlock new moves, and racing games allow the player to drive better cars.

Throughout a game, the level of difficulty should be incremental. A game that has sudden surges in difficulty can frustrate the player, causing him to quit. A game that suddenly becomes too easy will bore the player, who might move on to something more entertaining. To ensure a gradual increase in difficulty, you need to test the game over and over to make sure it's consistent. Then have other people test it, just to make sure you haven't become blind to your game's weaknesses, or to unforeseen ways of finishing it. Believe me, there will be ways of playing your game that you never imagined, often bypassing much of the content you've slaved over.

The Finale

The end of a game is both the easiest and hardest part to design. It's easy because you no longer have to train the player. He already knows how to play your game as well as can be expected. You don't need to pull your punches anymore, or worry about how you're going to top that last challenge. It's kind of a freeing experience, not having to worry about babying the player along anymore.

It's also hard, because the end needs to be satisfying. The player needs to feel that he has overcome the best you could throw at him. (Which isn't true... but he has to *think* that.) The finale needs to wrap up any plot hooks that might be laying about, all the bad guys who retreated earlier in the game need to show up, and you need to give out any quest items that the player needs before he fights the big bad boss.

Climax and Denouement

There are two parts to an ending: the climax and the denouement (pronounced *day-new-ma*). The climax is the height of excitement, where the plot comes together and

18 Chapter 1 ■ The Basics of Game Design

resolves itself in a way that's entertaining to the audience. The denouement is the part *after* the climax, where any hanging plot points or clarifications are given. In *Star Wars*, the climax is when Luke Skywalker blows up the Death Star; the denouement is the award ceremony afterward. It's always important to have a denouement in story-based games, so the player doesn't feel like there's something more he should do. The player needs that sense of closure. He needs that final sense of reward, where the game, through its characters, congratulates the player on a job well done, and assures him that they appreciate his efforts. The denouement is when you remind the player how much his character has grown in both power and wisdom, and how that growth has affected his place in the world.

Using the *Star Wars* example, the awards ceremony shows us Luke Skywalker as a changed man. In the beginning, he was a friendless, whiny farm boy whose self-centered world view was focused purely on how tough he had it being a moisture farmer on a nowhere planet. By the end, we see him standing tall, accepting awards with some dignity along with Han Solo and Princess Leia, both of whom had looked at him like he was a backwater idiot earlier in the movie, and now look at him as an equal and a friend. Showing us this growth in the movie helps remind us of all that has happened, and brings to our minds "what comes next?"

The Climax

In the climax, or the finale of the game, you want the player to use all the tricks he's learned. Challenges have to be wide-ranging, allowing the player to demonstrate all the skills he's learned while progressing through your game. You should allow him to use the best weapons, the fastest vehicles, or the coolest martial arts moves. Let him be the master he's trained the whole game to be.

The end challenge has probably been fairly predictable for some time by the climax. The player will have a general idea of whom or what he'll be facing and be raring to go. For this reason, it's usually best to make your climax straightforward, but tough. Don't try to throw in any unexpected gameplay or tricks at this time that you haven't already used. The finale should be the culmination of everything that has come before, not a step in a new direction.

Finishing a good finale should leave the player sweaty and joyful. He has achieved victory after a hard-fought battle. An easy success here would be disappointing to the player.

Unfortunately, this is usually too often the case. In a lot of games, the player has accumulated so much experience and equipment that it becomes hard to defeat him without cheating. There is also a lot of range in what each individual player's abilities are, so it becomes hard to predict exactly how a player might approach your finale.

Once again, playing this battle over and over, in every single way you possibly can, is the way to find those weaknesses and overcome them. The end of your game is often the determining point on how well a player remembers your game, or how well a reviewer scores it. The finale of your game is nearly as important as the beginning in this respect. Make sure your climax is the best it can possibly be.

The Denouement

The end movie is usually created to illustrate the denouement. In order to show the player how much his character has changed, and how the universe has been affected by his deeds within your game, you'll probably plan to include something just as grand or grander than the initial, beginning movie.

Unfortunately, things will probably have conspired against you.

Games change in their making. As your game comes to life, some aspects will have become obviously undoable, and some really nice, new plot events or features will get implemented along the way. This makes the end of the game highly unstable until such time that you've cemented down all of the parts previous to it.

This means that your ending movie will probably be the last thing you want to create, since it needs to incorporate the whole of all that has come before it. Because of this, a lot of pressure in terms of time, money, and resources will build up. You'll be running out of time, running out of money, and running out of people by the end of the project.

It would be easy to just let the whole thing go. To make a short, sorry little movie that barely does justice to your game.

Don't do that! The players who have worked so hard to get through your game should be rewarded. Give them the best ending you can. If your game isn't story-based, like a racing game, save something really nice to give the player after he beats that final opponent. Be creative with it. Just make sure that the end of your game receives all the attention and love that it deserves.

Making Your Game Better

Like any skill, designing games is something you get better at with practice. You'll develop a bag of design tricks that you can draw from: how to make a challenge tougher, how to better pace your game, where a story works and where it doesn't. In this section, I want to share some of the stuff I've learned that has made me a better designer.

Respect the Player

The player is not your enemy. He's your paycheck. He's just as smart as you are, and he has similar goals and dreams. He deserves your respect and thanks.

Show that you respect the player's time by letting him quit your game at any time without losing his place. If possible, don't make him redo things he's already done. Give him as much choice as you can allow (*Deus Ex* is a great example—see Figure 1.3). Let him choose how his character looks, what sex his character is, and how his character deals with the various challenges you present him. Always make your game's rules and laws consistent. The player must know the consequences of failure, and it must be dealt with swiftly so he can start playing again as soon as possible.

There are a thousand more ways to show your respect for the player. Try to do so any way you can.

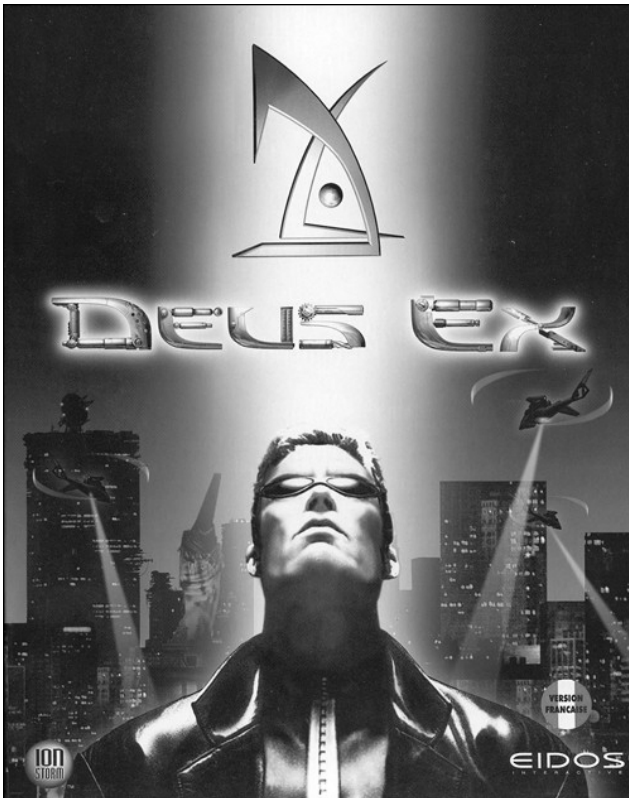


Figure 1.3 Ion Storm's *Deus Ex* shows great respect for the player. The single-player campaign allows for several different play types, which can be combined to customize how the player overcomes the game's many challenges.

Hubris Is the Game-Killer

Game designers often suffer from hubris, or excessive pride and presumption. This is generally the result of the designer forgetting that he is an entertainer, and thus providing a service for others. Instead, he begins to feel that he is an artist, who is allowing others to play in a world of his own design.

You can tell you have hubris if you suffer from the following symptoms:

1. A general dislike or hatred for your audience. You feel as if your players are intellectually stunted children and annoying, unworthy opponents.
2. You easily become upset when you see someone playing your game in a way that you didn't mean it to be played.
3. You begin to take delight in thinking of ways to punish players for doing stupid or objectionable things within your game, like using cheat codes or acting in anti-social ways toward friendly non-player characters.
4. You begin to take bad reviews or insulting internet posts as personal attacks on your character.

Hubris can make you forget to try to make a fun game, and will cause you to be blind to obvious faults within your creations. It will give you easy excuses to salve your ego as you start thinking of anyone who doesn't like your game as a jealous low-brow troglodyte maliciously trying to tear your game down.

Basically, hubris will make you a jerk and a bad game designer.

As I said at the beginning of this chapter, creating games is all about entertaining others. Never forget that. Try to stay above the fray and concentrate on making the best experience you can. *Avoid hubris.*

If You Aren't Having Fun, the Game Won't Be Fun

You can't make a worthwhile game unless you're having fun doing it. If you dread working on your game, it probably won't be very fun. Even if you do manage to finish it, what's the point?

This isn't to say you have to be giggling with glee through every step in the process, or that you shouldn't take the game seriously. But people tend to be more creative when there's laughter involved. Advertising tycoon David Ogilvy once said, "The best ideas come as jokes. Make your thinking as funny as possible." He also said, "If it doesn't sell, it isn't creative." Fun helps spark creativity, and creativity helps sell your game.

It's Just a Game

Finally, the last thing to remember is that you shouldn't sacrifice relationships with your family or friends because of a game. It's easy to get lost in the work and spend long hours on it, ignoring the people you care about in the process. Creating games is *not* a higher calling. Your game won't save any lives. It won't bring about a social upheaval that brings peace to the world. You're entertaining people. That's important, but not as important as your own life. Don't squander your relationships so somebody you'll never meet can have a few hours of diversion.

What You Have Learned

This chapter has taught you the following concepts:

- Games are for entertaining people.
- To come up with clear ideas about what does and doesn't work in games, you should study three things: your audience, other games in your chosen genre, and your own gaming preferences.
- Games should empower their players.
- A game's challenges make up its gameplay.
- Breaking down challenges into their most basic forms allows you to use a mix-and-match method to create new, innovative challenges.
- Give the player rest breaks between challenges. This helps to create a good pace, which keeps him playing your game.
- The first 10 minutes of your game are the most important.
- Consistency and growth are the two main concepts that keep your game interesting and playable.
- At the end of your game, give the player a good sense of closure. Also, reward the player who finishes your game.
- Respect your player.
- Don't fall into an "Us vs. Them" mindset.
- Give your game as much polish as possible, and have fun doing it, but don't forget your family and friends in the process.

Review Questions

(Answers are located in Appendix C)

1. As a game maker, your primary duty is to do what?
2. What are the three types of power?
3. Where can you go to find opinions and thoughts about games so you can find out what works and what doesn't?
4. Name three types of challenges:
5. What is hubris?

On Your Own

1. Go to Gamerankings.com and look for your favorite game. Read the best review, the worst review, and a preview.
 - a. Did the preview have any information about the game that was just plain wrong? If so, what was it? Also, how do you think the players felt when they found out this facet of the game wasn't in the final version?
 - b. What did the worst review and the best review disagree on? Did you feel one of them was wrong? Why?
2. Pick a game that you've finished and write a mini-review of it. (This shouldn't be the same game that you looked at in the previous exercise.) This mini-review should cover the following:
 - a. The game's greatest strengths.
 - b. Its greatest weaknesses.
 - c. The most compelling moment of that game, the part that you remember best.
 - d. The user interface, including which keys or buttons performed which actions, what the screen looked like when you were playing, and what the menu system looked like. Include a paragraph on what they could have done to make it better.
3. Write down five different ideas for games. They can be silly or serious. Using what you know about how games are developed, identify the game that would be the easiest to create, and the one that would be hardest to create.

