Chapter 3

General Game Design: Strategy Games

THE EXPERTS

- Sid Meier, Firaxis
- Bill Roper, Blizzard North
- Brian Reynolds, Big Huge Games
- Bruce C. Shelley, Ensemble Studios
- Peter Molyneux, Lionhead Studios
- Alex Garden, Relic Entertainment
- Louis Castle, Electronic Arts/ Westwood Studios
- Chris Sawyer, Freelance
- Rick Goodman, Stainless Steel Studios
- Phil Steinmeyer, PopTop Software
- Ed Del Castillo, Liquid Entertainment

Do you like to use some brains along with (or instead of) brawn when gaming? This chapter is for you—how to create breathtaking strategy games. And do we have a roundtable of celebrities for you!

Sid Meier, Firaxis

There's a very good reason why Sid Meier is one of the most accomplished and respected game designers in the business. He pioneered the industry with a number of unprecedented instant classics, such as the very first combat flight simulator, *F-15 Strike Eagle*; then *Pirates*, *Railroad Tycoon*, and of course, a game often voted the number one game of all time, *Civilization*. Meier has contributed to a number of chapters in this book, but here he offers a few words on game inspiration.

"Find something you as a designer are excited about," begins Meier. "If not, it will likely show through your work." Meier also reminds designers that this is a project that they'll be working on for about two years, and designers have to ask themselves whether this is something they want to work on every day for that length of time. From a practical point of view, Meier says, "You probably don't want to get into a genre that's overly exhausted."

> For me, working on *SimGolf* is a fine example, and *Gettysburg* is another—something I've been fascinated with all my life, and it wasn't mainstream, but was a lot of fun to write—a fun game to put together.

To Meier, it all boils back down to passion:

> What do you get excited about, and what are you good at? Do an RPG, not an action shooter just because it's in style. Find something new and fresh—publishers want to be leading-edge, too, so they're usually receptive to new ideas. But remember, for every 20 guys who walk into the door, maybe two ideas are worth considering.

> There's a temptation to create "the ultimate game of all time," so keep your focus on a couple of cool features, make sure those are great, and leave some room for a second game.

Be sure to read Meier's advice on programming, artificial intelligence, proper game testing, breaking into the industry, and more in later chapters.

Bill Roper, Blizzard North

Blizzard North VP Bill Roper is an industry veteran with countless triple-A titles under his belt, including *WarCraft II: Tides of Darkness* (producer), *Diablo* (producer), *StarCraft* (producer), *Diablo II* (senior producer), and *StarCraft: Brood War* (executive producer). Most recently, he has participated on the oversight team for *WarCraft III: Reign of Chaos*, in which he was instrumental in shaping the direction of the game.

We chat in depth with Roper in Chapter 5, "General Game Design: Role-Playing Games (RPGs) and Persistent Online Worlds," on role-playing game development, but here we have his inspiring and enlightening words on strategy game design, and how *WarCraft III: Reign of Chaos* could serve as a benchmark to others. Roper was first asked to discuss what he personally learned from working on such epic projects as *WarCraft III*. Following is his thorough response.

Persistence of Vision

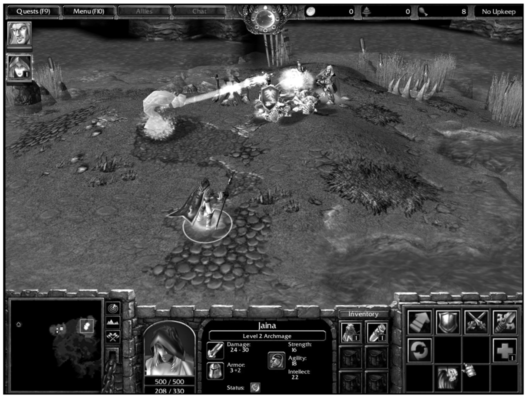
> The main goal of *WarCraft III* was to create a real-time strategy game that was heavily influenced by role-playing sensibilities. Although we went through innumerable changes, both grand and minute, we strove to keep the concept of the game in the forefront of discussions. This often proved challenging when we would have to try to balance the needs of a strategy game with the mechanics of a role-playing game. One of the key features of the game is the "Legendary Hero" units, and to emphasize their importance, we originally used an over-the-shoulder camera angle and required that regular units be "attached" to the Hero. This gave the game a very distinct look and feel while immediately driving home the point that role-playing elements were going to play a big part in the experience. Unfortunately, we discovered that this design prevented us from doing many things that we felt were necessary to create a strategy game. While the camera angle needed to be changed, we also didn't want to lose the immersion we had attained, so we also made sure that the environment had a much more organic feel as we pulled back the camera, making it easier to strategically control multiple units while at the same time eliminating the "game board" feel from which many strategy games suffer. Keeping our eye on the overall vision for the project allowed us to make major changes to even key elements of the game, with the knowledge that we would end up with a cohesive and fun game design.

Letting Go

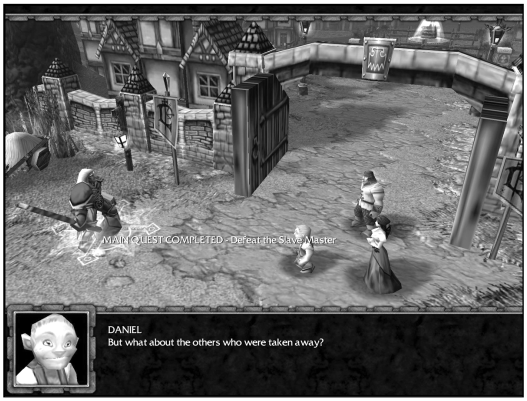
> During the creative process of designing a game, you come up with thousands of great ideas and not-so-great ideas. Sometimes you come up with ideas that seem like amazing groundbreaking concepts. Occasionally, these ideas just don't work. Whether for technical reasons, world continuity and design considerations, or simply because they're just not fun to play, these ideas need to be changed—or set

aside. Sometimes, this can be a simple matter when you try something out and realize that it isn't fun or simply doesn't look right. Other times, this can be an extremely difficult decision because you can see how it might work and are willing to try and find ways to keep the idea alive. For example, we had a major shift in the look and the associated design elements of *WarCraft III* after a year or so of development, when we moved away from the over-the-shoulder angle associated with Hero units. The interesting aspect of this decision was that although it probably caused some concern and confusion to our players, the development team actually found that they could now make the game they envisioned; the rest of our design concepts related to melding role-playing elements into a strategy game were basically unaffected.

Dumping an idea that you have conceived, fostered, and perhaps even implemented is a painful but necessary process. Not everything works out as planned, and it's difficult to admit when that gameplay mechanic you thought would revolutionize the industry turns out to be an adventure into tedium. When you're faced with the need to pull the plug on something in the game—no matter what the cause—you have to do so as quickly and painlessly as possible. Don't be afraid to scrap ideas because many times you'll replace them with something far better or simply find that [you don't need] anything there at all.



Jaina
Level 2 Archmage
Damage: 24 - 30
Armor: 3 +2
Strength: 16
Agility: 18
Intellect: 22
Status:
500 / 500
208 / 330



MAIN QUEST COMPLETED - Defeat the Slave Master

DANIEL
But what about the others who were taken away?

Bill Roper says that *WarCraft III* is "one of our greatest accomplishments to date. We started out hoping to create a fitting sequel to *WarCraft II*, and we ended up crafting a world of epic proportions. *WarCraft III* offers players an incredibly fun and dynamic experience."
(Used with permission by Blizzard Entertainment, Inc.)

Taking Time for Tools

> The design and implementation of the "World Editor" for *WarCraft III* was key to the successful realization of the role-playing game aspects found in the single-player campaign. The level designers worked hand-in-hand with the editor programmer to create a tool that was not only powerful but also easy enough for them to use day in and day out. While it's not uncommon to see tools created without much thought given to the end user, the growth of the World Editor was a cooperative venture between the people making it and the people using it. This was essential, not just because we intended to ship it as a part of the game, but also because we knew that we wanted to free up our game programmers as much as possible from having to do special-case programming for the campaign. By putting a heavy focus on designing a robust tool that the level designers could use to modify their work and then immediately review it in the game, we streamlined our development process [and gained] many more chances to iterate on those designs while reducing our required quality assurance testing time [see Chapter 17, "Proper Game Testing"]. It's said that a craftsman is only as good as his tools, so giving your craftsmen the best possible tools will help them create the best product they can.

Is there anything that Roper would've done differently? "We should have spent more time in designing the game before we put as many programmers and artists on the project as we did," he admits. Roper explains:

> By nature, game design is a very organic and iterative process, but in our excitement to start making the game, we honestly put too many people on the project at too early a stage. It's difficult to gauge the proper numbers to assign to a project in the first weeks and months, but the biggest lesson we learned was to put as many designers into the mix as possible. Also, keeping multiple high-end designers in addition to the level designers on the project until the end was something we didn't address until later in the development process. Fortunately, we caught this issue before it was critical, but it certainly set back our balance and campaign designs to some extent.

In addition to the *Legendary Heroes* and the role-playing game flavor, what else makes *WarCraft III* stand out from other real-time strategy games? "While there are many elements that go into any successful game, I honestly feel the character and emotional bond gamers get while playing *WarCraft III* makes it far different from any strategy game we have ever created."

> Movies have come a long way in the past 20 to 30 years in terms of what can be technologically created and brought to life onscreen. Games have leaped forward in the past 5 to 10 years in terms of what can be accomplished and represented with much of that same type of technology. Games provide potential immersion mainly because they're interactive, placing the player in control of the situation—something that movies simply can't do. But movies have traditionally provided something that games don't—an emotional connection between the entertained and the entertainment. This is an area where we as game developers can grow, and it was a major focus of the unit and level designers as well as the writers, cinematic artists, sound designers, and musicians on *WarCraft III*.
>
> In terms of gameplay, the units in *WarCraft III* play an even more important and unique role within the composition of an army than in any of our previous games. Even the most basic unit has some special twist that makes him different and a viable choice throughout the course of a game. The top-end units—

> the Legendary Heroes—are the epic leaders of those armies and can greatly influence the outcome of any battle in which they're employed. These special units really show how we used the sensibilities of role-playing games within a real-time strategy game as the units grow in experience, gain levels, and find and purchase items that they carry in an inventory. They even have their own names, making them a very personal element in the scope of any game. When one of these Heroes dies in battle, a message is sent throughout the game announcing his or her defeat, and the death animation is far more grand and dramatic than those of his less-epic compatriots. Although *WarCraft III* is first and foremost a real-time strategy game, these elements provide an interesting and unique gaming experience.

Roper says that in terms of making an emotional connection with the player, they spent a huge amount of time crafting a rich storyline that's expressed through the single-player campaign:

> The use of pre-rendered and in-game cinematic sequences, professional voice acting, and missions that are centered around a story (and not the other way around) create a compelling game that involves not only the skills of the players, but also their feelings. As the story unfolds, both the world and the characters within undergo changes. Reaching someone on a deeper level only makes for a more satisfying experience, and this is something that other forms of entertainment do extremely well. With *WarCraft III*, we wanted to involve the player on many levels while still tapping into the adrenal responses that have served games so well for so many years. If we can succeed in this endeavor, we can reach a wider variety of people than just the traditional core gamer.
>
> Anything and everything that touches your life is potential fodder for the creation of a game. Did you ever get stuck on the expressway and wish you could just jump the center divider, blast through oncoming traffic, and cinematically sail into the local 7-Eleven parking lot so you could grab a Slurpee while waiting for traffic to break? *Grand Theft Auto 3*. Have you ever sat through a long boring dinner, lazily arranging and rearranging the vegetables on your plate into new and interesting patterns—eating only the ones that line up three-in-a-row? *Bejeweled*. Have you ever listened to "Ride of the Valkyries" by Wagner while staring into a clear blue summer sky? The Wyvern Riders in *WarCraft III*.

What about the origins of *WarCraft III*?

> *WarCraft III* came from (a) wanting to expand the world we had already created and grown; and (b) not wanting to waste an exceptionally strong storyline that had been created for a canceled game. When we looked at where to go with the WarCraft universe after destroying the Dark Portal in the expansion set to *WarCraft II*, our creative director, Chris Metzen, set to crafting a story that would focus on what happened to the orcs who were trapped in Azeroth. Elements of the story and the background of where the orcs came from as a people were inspired by the culture and plight of the American Indians. The internment camps of World War II also played a part in shaping the background of how the orcs were marshaled and corralled after the war in Azeroth. The gladiatorial rings of ancient Rome and the way their combatants were both revered and feared as they lived out violent lives as valuable property also played a major part in shaping the character of Thrall, the main character of the orc campaign.
>
> Music, literature, television, animation, art, sports, music, food, love of debate, social interaction—these are all things that can spark ideas for a game designer. Also, playing other games is key—especially with other people on your team. The best solos in jazz are a fusion of the mind and skill of the soloist with a

> responsive and iterative rhythm section. They listen to what the soloist is doing and then play with it and off of it. In turn, the soloist responds with his own intuitive iterations or departures. When a second soloist joins the fray, his performance is augmented by what he has just heard only if he can build on and learn from that experience. As game developers, we shouldn't be afraid to play other companies' games and learn from what they did. We also need to play our games and learn from our own mistakes and failures. The rest of the equation is to listen to the people who buy our games and see what they want out of a play experience. All these things can inspire new chains of thought and can lead to new ideas in the design of game mechanics, world concept, and story.

In Chapter 21, "Breaking into the Industry," Bill Roper offers some advice on breaking into the industry.

Brian Reynolds, Big Huge Games

Throughout his illustrious career as a programmer and game designer, Brian Reynolds, now president of Big Huge Games, worked alongside Sid Meier at Microprose and Firaxis on such remarkable strategy games as *Colonization*, *Civilization II*, and *Alpha Centauri*. His latest project is *Rise of Nations*. Read more about this game, published by Microsoft Game Studios, at `www.bighugegames.com`. It seemed a simple task for Reynolds to list his best advice for strategy game designers and support these words with an example from past or current projects:

1. Get something running in the first month that you can actually play. (It doesn't matter if graphics aren't so great.) With Civilization II, we had the game to where you could play it all the way through over a year before it shipped (even though it was all-new code from the original Civilization). We knew that we needed to play and try out all the new things we wanted to add and that we'd need that much time to get the balance and AI tuned. For our current game at Big Huge, we had something playable by our first milestone, even though all we were actually required to "deliver" was a design document.

   Strategy games are extremely complex to design—although the individual components look deceptively simple, having a lot of "simple" moving parts makes for a very complex overall balancing task. It's easy to look at a strategy game and say, "I could make this better; I'd add this and this and this," but very hard to actually integrate lots of new parts into a game system without breaking the things that were already fun. To balance all the moving parts correctly, there's no substitute for actually playing your own game—the combinatorial explosion from all the moving parts makes it impossible to truly anticipate or tune results "on paper" in a design document. The sooner you get your game running, the sooner you can actually get to work on making the game fun and making it balanced. Both fun and balance tend to be taken for granted by novice designers. They think, "If I make a game about topic X and it has features A, B, and C and technology J, then it will be fun," but as it turns out, fun and balance both take a lot of hard work.

2. Each strategy should have both a unique strength to make it cool and a unique weakness to keep it from being too powerful. "Rock, paper, scissors" is the best model for game balance. Our current real-time game is based around a "rock, paper, scissors" model for game balance. That is, unit A is really strong against unit B, but weak against unit C; whereas B is weak against A and strong against C, etc. That way, no unit is so powerful that it's "unbalanced." Also, all units are strong against something, so they're cool to build in the correct context.

3. Play your game regularly. If you have multiplayer [capability], get that running early so that you can balance it. By the time our new game ships, we'll have had our multiplayer running for at least 18 months. We've built a special "multiplayer lab" with eight workstations side by side (it's modeled on the lab we saw at Ensemble Studios), and we run at least one multiplayer game a day there—often several. We have "novice days," "pro days," "intermediate days," "free-for-all days," and so forth—everyone in the company signs up and plays. Everybody in the company gets to see their own work interacting with the rest of the game, gets to see the "big picture," and has a chance to contribute to the design and balance. The progress on design and game balance was dramatic once we got the lab in, and we have plenty of time left.

Most of Brian Reynold's advice in this book is found in the chapters on programming (see Chapter 11, "Programming Theory") and artificial intelligence (Chapter 12, "Artificial Intelligence [AI]"), as well as on how best to break into the industry (Chapter 21).

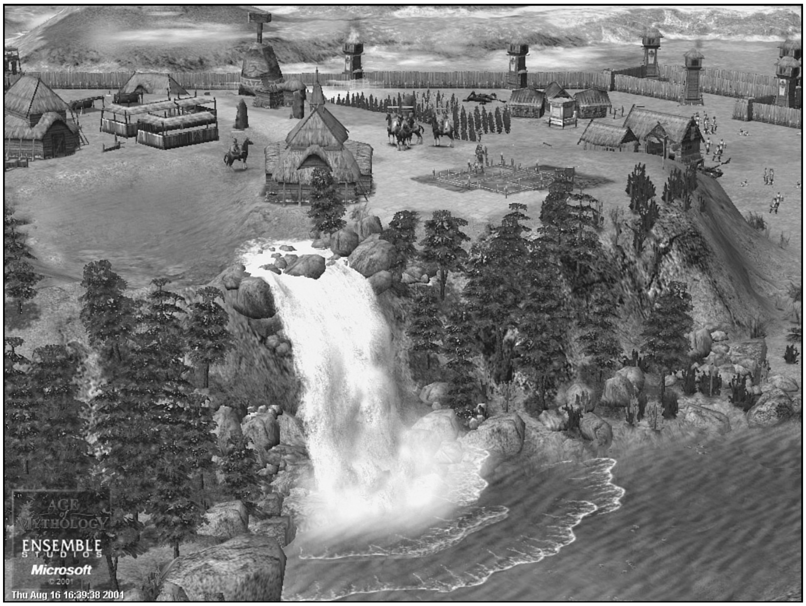
Bruce C. Shelley, Ensemble Studios

Bruce Shelley is a lead game designer and spokesman for Ensemble Studios, the development studio that has created award-winning and critically acclaimed real-time strategy games for Microsoft. In the past, Shelley has worked on computer games such as *Civilization* (Microprose), *Railroad Tycoon* (Microprose), and others. His latest project was the million-unit-seller, *Age of Mythology*. Shelley offers some key tips for those starting out in the industry:

> Be familiar with lots of games, but especially those most like the game you most want to design. Consider which parts of a game are working and which are not.
>
> Providing a player with interesting decisions is the rocket science of game design. If you agree, then consider whether a game you're playing or designing is providing interesting decisions. When you attempt to add a new piece to a game, ask yourself whether it will add interesting decisions for the player to deal with.
>
> When considering a new game, look at the competition first and make lists of features they do well, poorly, or not at all. The features they do well are the minimum requirements for your game. The features they do poorly or not at all are a list of opportunities where you can differentiate your game and offer innovation. Always seek to differentiate and innovate—don't clone around.

On providing direct examples, Shelley says this:

> When we set out to make the original *Age of Empires (AOE)*, we made our lists of features that were done well, poorly, or not at all in games like *Civilization*, *WarCraft*, and *Command & Conquer*. From these, we built our list of must-have features for *AOE*, such as hidden maps, economic buildup phase, empire building, town conquest, great multiplayer, scenario editor, and differentiated civilizations. Then we built the list of features that would differentiate *Age of Empires* because other games were not doing them well, such as historical theme, random maps, non-cheating AI, levels of difficulty, realistic graphics (not cartoonish), and multiple victory conditions. By meeting the competition where they were strong and providing clear differentiation and innovation in *AOE*, we created a game that was able to be quite successful in a very competitive genre (50+ RTS games in development in 1997). Competing games that fell by the side came up short mainly because they didn't sufficiently differentiate and innovate from the best games in the genre already available.



Check out this screen grab from Shelley's latest, *Age of Mythology*, published by Microsoft Game Studios in 2002. Talk about gorgeous graphics!

(Used with permission by Microsoft Corp. and Ensemble Studios, Inc.)

Shelley says it's critical for a game designer to keep these three pointers in mind when starting out:

1. Design for a broad global audience, not a narrowly focused, especially hardcore audience. The majority of game buyers are casual gamers, and too many games are beyond their skill or tastes.

2. Within a specific genre, make your game different in terms of topic, look, and feel. At the gameplay level, innovate new features and gameplay. Don't imitate successful games only to the point of being at least as good as they are on their strong points. Create a new experience with your game. Nobody buys imitations.
3. After you have a basic design document, prototype the game quickly and thereafter design by playing. Play every day, make adjustments, get opinions, recode for tomorrow, and play again. If you have enough good gamers giving comments and your instincts as a designer are good, you'll design a fun game.

Shelley supports this advice in light of his latest project, *Age of Mythology*:

1. Included within *Age of Mythology* are multiplayer games, a single-player campaign, random map Solitaire games, and a variety of game types such as death match. We believe that gamers of all skill levels and from all global PC markets will find a satisfying way to enjoy our game. We think this is reaching out to a broad global market, as all of our past games have done.
2. Adding the mythological elements and moving to 3D technology will make our game different from other RTS games, including the *Age of Empires* games. At the gameplay level, we think we've innovated with myth creatures, god powers, and more subtle changes. Overall, we've made *Age of Mythology* both different and innovative to provide a new experience. No one will consider it a clone of something else.
3. *Age of Mythology* has been playable since early 2001. Since then, we continued to design the game by playing every day, making changes, and recoding through thousands of iterations.

Shelley is also known for this piece of advice: "A game has to have a great first 15 minutes!"

> The concept of a great first 15 minutes was Sid Meier's. He mentioned to me that one of the reasons he thought *AOE* did well was because it had a great first 15 minutes. The point is that a game has to engage a new player within 15 minutes of that person sitting down to play. If not, the player is likely to give up and never try the game again. So the start of the game has to be engaging—get the player absorbed by presenting a lot of interesting decisions that pile on top of one another. Another one of Sid's phrases was the "inverted pyramid of decision-making." You want the player to deal with only a few decisions at first, which multiply quickly, fully absorbing the player. Games that do this pull the new player in.
>
> In the original *AOE* game, for example, the map is almost entirely hidden at the start. The new player has a few villagers to put to work and use to explore. Once he or she has built more villagers and begun exploring, there are more decisions to be made: what tasks to assign to newcomers, where to place gathering buildings, which new directions to explore, what strategic choke points to watch, etc.

It's been said that a game designer need not reinvent the wheel—that it's perfectly acceptable to improve on an existing formula. Does Shelley agree or disagree?

> The most valuable resource any game designer has is all the games that already exist from which we can get ideas. The risk is that we may do too much copying and not enough creating. Gamers want a new experience. It can be familiar or similar to other games they've played, but it must also be different and innovative to be successful. *Age of Mythology* is a big RTS similar to many others, including our own *Age of Empires* games. We've played and considered everything in the games that have been published. But we've also tried to create something that's different and innovative.

On creating an innovative yet accessible game, Shelley adds the following:

> Basically, cloning a successful game is a recipe for disaster. We believe a new game must be clearly different from existing products or people will ignore it. For the *Age of Empires* games, the big differentiation was choosing historic themes when the competition was doing mostly sci-fi and fantasy. As a bonus, it turned out that there was a huge interest in the casual market for a historical RTS, which no one else was trying to fill. Innovation is important also, perhaps mainly with the hardcore gamers. They want a new experience. If a new game doesn't add much that's new and fun, they won't bother to put in the time to learn it. Graphic look and feel can be an important differentiation. Tony Goodman, our CEO and early art director, championed a fairly realistic graphic look and bright colors. The competition at the time was going with a more cartoonish look and often very dark and gloomy colors. We think now that our look was an important differentiator and another important key to attracting casual gamers. We think too many developers largely ignore the casual market, to their loss. Most game sales reside in the casual market, yet developers often try to outdo themselves focusing on what only the hardcores seem to want. Make commercial art, not fine art. When someone says they're making the game they've always wanted to make, the question is, how many of them does that person intend to buy? It makes much more business sense to say that you're going to make the game that you think millions of average casual gamers are going to want to own. That's the kind of thinking that publishers want to hear.

Finally, Shelley was asked to be frank about what it's like working with a powerful publisher such as Microsoft—something many game designers are likely curious about! Shelley says the pros are that "big publishers get shelf space; have marketing muscle; have PR departments; and are usually well organized for testing, localization, and manual creation. They do virtually everything better than the alternatives (small publisher, self-publishing)." But he discusses some of the cons as well: "They usually control the purse strings and can dictate schedule dates and budgets. Your game is just one in their portfolio. They may use it for strategic purposes good for their company, but not necessarily for yours (loss leader, bundling deals)."

Bruce Shelley offers advice on many other topics in this book, including writing design documents (Chapter 6, "Creating Characters, Storyboarding, and Design Documents), coding realistic AI (Chapter 12), creating a good user interface (Chapter 14, "The All-Important User Interface [UI] and Game Control), and breaking into the industry (Chapter 21).

Peter Molyneux, Lionhead Studios

One of the computer game industry's most revered game designers is Peter Molyneux, the brilliant (and soft-spoken) managing director at Lionhead Studios. And talk about a track record! Molyneux is responsible for some of the most beloved PC games, including *Populous*, *Magic Carpet*, *Theme Park*, *Dungeon Keeper*, and the *Black & White* series. Molyneux is currently overseeing a number of new projects, including *Project Ego*, *Black & White 2*, *Black and White: Next Generation*, and *The Movies*.

Molyneux shares some advice he wishes he was told before he got involved in this industry:

> 1. Come up with a concept that's easy to explain in a sentence or so. This was highlighted to me by Bing Gordon at Electronic Arts back in 1993. I was heading up to a meeting to try to sell the concept of *Magic Carpet*. [As we rode up together in the elevator] I said that I was looking forward to making the presentation, and Bing replied that if I couldn't explain the concept on the way up then it probably wasn't a good concept. We were on the first floor, and the meeting was on the second floor.
>
>   Games that are easy to explain and that deal with subjects people can easily understand are always going to be more successful than more abstract ideas.
> 2. The most brilliant concept is useless unless you can think of the way that people will play the game. In the end, game design comes down to interface design—the key to making games playable is how you'll get people to interact with your concept and how simple the interface is.
> 3. Don't be frightened to be original and innovative. Remember that the greatest game concepts and interfaces are yet to be designed; just because everyone else has done things a certain way previously doesn't necessarily make it the right and only way to do something.

Put to the test—to support his advice with real-world examples—Molyneux passes brilliantly:

> The idea of "designing and building your own theme park" was very accessible and clear—in one sentence, you know all about the game and what you need to do to play it. Although this is the most perfect example, I can think of others. You "play God" in *Populous*, you play the bad guy in *Dungeon Keeper*, and you find out who you are in *Black & White*—all are examples of games that explain themselves.
>
> In *Black & White*, we wanted people to be able to do anything they wanted in the game world, so using the hand meant we didn't have to use a lot of explanations in the interface (although for this reason I think the tutorial should have been longer).

Molyneux once said in an interview that he would prefer to make games with a shorter development period. He clarifies this point:

> The length of time a project takes is becoming a real issue. More power means bigger teams, having to take care of more than just graphics and programming—AI, physics, interfaces, gameplay—and all of these elements need coding to be built from scratch. It makes the task of developing a game quite overwhelming. For a game to have the best graphics, sounds, and animation means that developing a game is becoming a Herculean effort. I hope that with better internal organization of our team, it will be possible to compress 3–4 years into 2–3 years.

Molyneux talks briefly about programming in Chapter 11 and breaking into the industry in Chapter 21.

Peter Molyneux says, "The greatest game concepts and interfaces are yet to be designed." Pictured here are a couple of sneak peeks at the upcoming *Black & White 2*.
(Used with permission by Lionhead Studios.)

Alex Garden, Relic Entertainment

Alex Garden's most recent venture, Relic Entertainment, is one of North America's premiere game-development studios. Relic's first title, *Homeworld*, has won more than 50 awards, including the prestigious "Game of the Year" award from CNN.com, MSNBC, and the world's largest computer game magazine, *PC Gamer*. Garden's latest game is *Impossible Creatures*, released in early 2003. This game allows players to make war-hungry creatures by combining animal traits.

Garden was cornered to surrender some of his best-kept game design theories. Interestingly, his answers are similar to Sid Meier's advice:

1. Design around themes that you're passionate about. Life is short, and making games is supposed to be fun. Make sure you're having fun making games, and chances are that people will have fun playing them.

2. When you're coming up with concepts, focus more on what makes your game fundamentally fun to play, not what the story is. Understand that as a game player you rarely consider the underlying mechanics of the experience you're having. More often, you focus on the story that you're playing through. When you read a book, you rarely think about the fundamental appeal of the book; you think about the setting, characters, etc. As a game designer, the paramount rule you must obey is to make sure that the core experience is appealing before you worry about the story. If story is your only concern, write a book.

3. When making decisions about your game, ask yourself two questions: "What do I think is the right thing to do?" and "What does the paying public think is the right thing to do?" A totally uncompromising commitment to your own personal vision creates art. But art alone doesn't sell. You must make fun art.

As examples, Garden provides the following:

> *Homeworld* had a totally freeform 3D camera. Some people loved it, but most people found it too hard to use. With *Impossible Creatures*, we've included the option to play with a totally freeform 3D camera, but by default the game is played from a fixed viewpoint, which dramatically simplifies the learning curve, thereby making *Impossible Creatures* accessible to a wider audience.

How can you create a new concept yet still include enough familiar concepts so the game isn't too obscure to the gamer?

> Our design rule at Relic is "One revolutionary step, multiple evolutionary steps." Basically, we try to do one fundamental thing that will set us apart from everyone else, then we try to do everything standard better than everyone else. In *Impossible Creatures*, the revolutionary leap we took was to add a totally user-created army-building system with real-world animals. The evolutionary steps are all based on the lessons we learned through our experience with *Homeworld* and *Homeworld: Cataclysm*.

Does anything about today's games irk Garden?

> I think that most games (and indeed, most game designers) today are afraid to take risks. When you take risks, bad things can happen (*Daikatana*), but truly magical, wonderful things can happen, too (*The Sims*).

Later in the book, Alex Garden talks about design documents and storyboards (Chapter 6) and breaking into the industry (Chapter 21). Read more about Relic Studios, Alex Garden, and *Impossible Creatures* at `www.relic.com`.

Louis Castle, Electronic Arts/Westwood Studios

Louis Castle is the general manager and co-founder of Westwood Studios, one of the most successful game development houses in the world. Westwood is best known for its *Command & Conquer* franchise, as well as *Blade Runner*, *Monopoly*, *The Lion King*, *Earth and Beyond*, and others. In early 2003, Westwood Studios become integrated with its parent company and publisher, Electronic Arts.

Castle provides some advice on creating successful computer games:

1. **Know your audience.** Spend the time to figure out who is likely to buy your product and research competitive or similar products. If none exist, spend time playing the best-selling and respected titles on your platform of choice. Be your best and worst customer.
2. **Make design a deliberate process.** Abstract the activities, emotional beats, and just about everything you can think of. Use the abstractions as filters and compare them to your favorite products. Design in any form is a deliberate action that requires planning, execution, and ongoing analysis.
3. **Listen to others.** The more exposure you can get and the sooner you can get it, the better your chances are to realize design flaws before you're committed to them. Be able to give up your best idea if it doesn't work in the context of your vision. Don't "make it fit."

Asked to comment on the key differences when creating a game for a console versus a PC, Castle has these comments:

> The main difference between consoles and a PC is that a console game is one in which you're "playing" the interface device. You need to treat the controller as the actual game and let the audiovisual systems be subordinate to the device. The console player builds skill in a game through manipulating the interface device. The PC player builds skill from understanding the game and environment. The interface needs to be transparent, and the best PC games don't require "skill" in manipulating the keyboard and mouse—but rather, knowledge of what to do and when. Of course, both of these points are important on both platforms, but the emphasis is different. Console games are about instant satisfaction and long-term skill development. PC games are about quick entertainment and accomplishment followed by in-depth understanding and mastery.

How can you acquire a "dev kit" to work on a console game?

> Console development kits are acquired through well-known publishers who support the platform. Developers can get systems directly through the manufacturer, but it usually requires a track record and volume of titles to be of interest. The very early systems typically cost tens of thousands of dollars and are in very limited supply. (They're considerably less now, although developers are not allowed to reveal exact prices. Formerly of THQ, Mike Rubinelli talks more in depth about dev kits in Chapter 21.)

Chapter 21 thoroughly discusses the various ways to break into the industry, but Castle offers some sound advice here:

> Take a low-paying job in any discipline you can. This minimizes the exposure for a company and gives you an opportunity to prove yourself. Learn to follow and execute; then worry about leading. Put your ego on hold and let your professionalism win you the creative jobs. I've always found the best way to get a job is to prove you can do it first, with the blessing of your boss and peers.

Chris Sawyer, Freelance

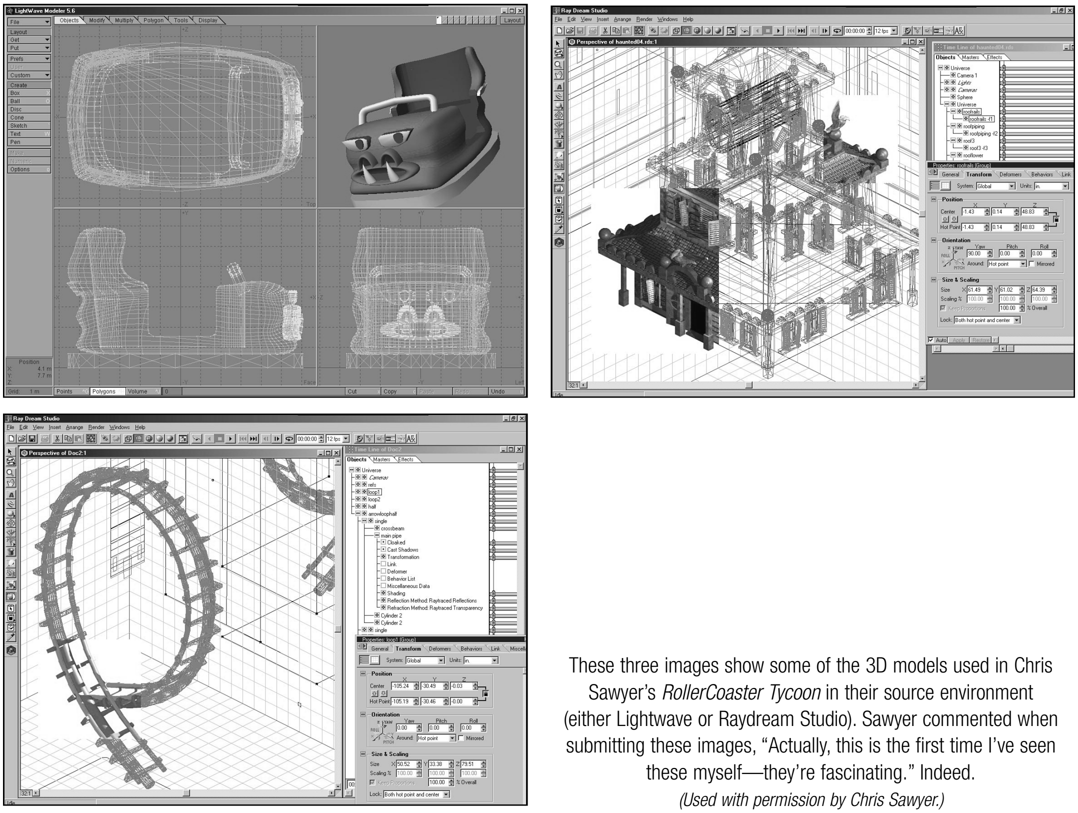
Although Chris Sawyer may not be a household name, this game developer, designer, and programmer created some of the most popular computer strategy/simulation games in the industry, including *Transport Tycoon*, *Transport Tycoon Deluxe*, the *RollerCoaster Tycoon* series, and its add-on packs *Corkscrew Follies* and *Loopy Landscapes*. Sawyer has also been in charge of many PC conversions of Amiga games, including *Frontier Elite 2*, *Xenomorph*, *Conqueror*, *Campaign*, *Virus*, *Goal*, and *Birds of Prey*.

Sawyer's contribution to the sage advice in this chapter:

1. Create something original and unique. Your game should look and play differently from other products to stand out.
2. Don't be put off if publishers initially discard your ideas and designs. When *RollerCoaster Tycoon* was in early development, I was told by everyone that it wouldn't succeed. The best way to convince people of a game's potential is to get the game to a playable state and actually let them play it.
3. Never forget that the ultimate aim of any computer game is to be fun. It's all very well coming up with a novel gameplay idea or an ever-more-immersive 3D world, but if it's not actually fun to play then it's worthless.

What was the most challenging aspect of creating *RollerCoaster Tycoon*?

> Perhaps the biggest challenge was to create a business model that was reasonably realistic but primarily "fun." I did this in *RollerCoaster Tycoon* by ignoring many of the financial and managerial aspects of running a theme park and concentrating more on the design and construction side.



System: Global Units: in.

Position

	X	Y	Z
Center	-1.43	0.14	48.83
Hot Point	-1.43	0.14	48.83

Orientation

Yaw	Pitch	Roll
90.00	0.00	0.00

Around: Hot point □ Mirrored

Size & Scaling

	X	Y	Z
Size	61.49	61.02	64.39
Scaling %	100.00	100.00	100.00

☑ Keep Proportions 100.00 % Overall

Lock: Both hot point and center

☑ Auto Apply Restore

System: Global Units: in.

Position

	X	Y	Z
Center	-105.24	-30.49	-0.03
Hot Point	-105.19	-30.46	-0.00

Orientation

Yaw	Pitch	Roll
0.00	0.00	0.00

Around: Hot point □ Mirrored

Size & Scaling

	X	Y	Z
Size	50.52	33.38	79.51
Scaling %	100.00	100.00	100.00

☑ Keep Proportions 100.00 % Overall

Lock: Both hot point and center

These three images show some of the 3D models used in Chris Sawyer's *RollerCoaster Tycoon* in their source environment (either Lightwave or Raydream Studio). Sawyer commented when submitting these images, "Actually, this is the first time I've seen these myself—they're fascinating." Indeed.
(Used with permission by Chris Sawyer.)

If Sawyer could go back and create the original *RollerCoaster Tycoon* all over again, would anything be done differently?

> Very little would change if I were re-creating the game to work with today's PCs. However, given more processing power, I'd reduce some of the compromises in the game, improve the graphical detail, and improve some of the AI.

Any pet peeves with today's computer games?

> Too many violent games, and not enough that focus on the more "positive" things we all enjoy, like building things and "nurturing" things.

Chris Sawyer returns in Chapter 12 to discuss artificial intelligence.

Rick Goodman, Stainless Steel Studios

While at Ensemble Studios, a company formed in 1995 with his brother Tony, Rick Goodman was the lead designer on the tremendously popular *Age of Empires* for Microsoft. He left Ensemble to form Stainless Steel Studios, where Goodman created the epic real-time strategy

(RTS) game *Empire Earth*, published by Sierra Studios in 2001. The game won "Best Strategy Game of 2001" by the readers at IGN (`http://pc.ign.com`). Goodman's latest game was *Empires: Dawn of the Modern World*, published by Activision.

Goodman offers us his "Ten Commandments of RTS Game Design," here in its entirety. Thou shalt pay attention:

1. The healthiest plants have the strongest roots. Goodman says to know your roots, whether comic books, sci-fi, board games, puzzles, toy soldiers, chess, *Dungeons & Dragons*, or anything else.

2. Paper never refuses ink. The key here is how to separate your good ideas from bad, otherwise known as "Hey, I have this great idea for a strategy game." Goodman says anyone can write a good design document, but there are four keys to success:

   - Play the game through in your head.
   - Model everything in a spreadsheet.
   - Prototype user interfaces.
   - Compose your "strike team" of play testers.

   In fact, Goodman used the Internet to recruit 12 savvy *Age of Empires* players and uses them all the time for guidance on *Empire Earth*, and to flush out bad ideas. Goodman reminds designers to welcome advice from others, with the adage "It's hard to see the picture when you're inside the frame."

3. A person who walks in another's tracks leaves no footprints. To innovate or to clone? Goodman reminds designers that we tend to criticize clones and then religiously follow up with new ones. He cites a phrase coined by Sid Meier, "innovative continuity," who came up with this term to not preclude him from doing anything he wanted. "Balance is key—add some innovative new things but make players comfortable in the game's surroundings so that they understand the rules and won't have to read the manual."

   Goodman says to "enhance the positive, eliminate the negative, and don't fiddle with the middle."

4. If I really wanted reality, would I have booted up the computer? Reinforce the fun factor over reality. Goodman refers to *Computer Gaming World* magazine polls that suggest that fantasy strategy games are often more popular than historical or reality-based ones. Even though *Age of Empires* is a historical strategy game, not everything is realistic: priest conversions, moving catapults, naval units attacking land units, and so forth.

5. If I had been present at creation, I would have given some useful hints. Goodman advises knowing your limits. Recognize your strengths and passions, and take advantage of them. For example, Goodman knew his knowledge of board games was very strong when he approached *Age of Empires*.

6. By the mile it's a trial, by the inch it's a cinch. Creating a game usually follows a top-down methodology, with game vision on top of the pyramid, game ideas and features in the middle, and low-level game mechanics on the bottom. Goodman explains, "There is a method to this process. The top is where the vision is generated ('How about a game where you're on a desert island?'); the middle is the game's features ('And this is what we'll do on this island'); and the bottom, the most crucial of the three, is how to implement these ideas in the game.

   According to Goodman, many game designers concentrate too much on the top two levels and not enough on how the play mechanics will work in the game. "The bottom level is the hardest; the implementation of the details is key," says Goodman. He also says most design docs seem to rely on the first two levels, when the emphasis should be on the implementation of the ideas into the computer and not just the ideas themselves.

7. Take note of the future; you're going to spend the rest of your life there. For the RTS genre to move forward, Goodman says, there has to be more to the game, and not just more of the same. He cites 3D action shooters as a genre that has evolved graphically, but not in gameplay (barring a few recent exceptions). With RTS games, try to foresee what will come down the pike and work toward bettering the genre.

8. Trees that are slow to grow bear the best fruit. "Time is your friend," says Goodman. "Use your project schedule well. *Age of Empires* was a 1,000-day project; 500 of those were spent internally at Ensemble Studios, the other half play-testing by Microsoft."

   "Beware—the only thing that can kill a good game is a great game. Be realistic at the beginning of your project and keep in mind that many of the great RTS games, such as *StarCraft* and *Age of Empires*, took longer than average to complete, but look at the outcome!"

9. Communicate your vision. One of the most important considerations while developing a game is communication. This is not only between a designer and his team, but between designer and publisher, the public, marketing, game critics, and testers.

10. Never put a glazed donut on a mouse pad. "…And other lessons learned the hard way," says Goodman. Some of his other lessons:

    - Schedules are not more important than quality.

- "Let's not tell them and maybe they won't find out" always backfires; it's no different than when you were a kid.

For more information on *Empire Earth* and the design team at Stainless Steel Studios, visit `www.stainlesssteelstudios.com`.

Phil Steinmeyer, PopTop Software

For the past few editions of this book, PopTop Software's president and lead programmer, Phil Steinmeyer, has shared savvy strategy design tips. Steinmeyer has a number of games under his belt, including *Railroad Tycoon 2* and the *Tropico* series.

When it comes to general game design, Steinmeyer says to focus on a small niche somewhere within the gaming universe for your first project:

> If you can, create something with a built-in audience that will find your product with little marketing or distribution clout on your end, and with no big-budget competition. Some excellent niches to start in are war games, detailed 4X space games, games about semi-obscure sports or hobbies, gambling games, and so forth.
>
> Too many beginning game developers overshoot on their first game and never get it done. First, games are rarely brilliant—better to get something completely done and published and shoot for the moon on your second game.

Where does *Tropico* fit in?

> *Tropico* was my fifth major game; I had a much bigger budget and a more experienced team than anybody would typically have starting out. My first game (released in 1994) was a low-budget war game called *Iron Cross*. I did all the programming, most of the art and sound, and it was my first game—a true garage effort. There was a built-in audience (World War II enthusiasts), and little big-budget competition. It didn't sell nearly as many copies as my later games like *Railroad Tycoon 2* and *Tropico*, but it did all right given its budget—I made a decent amount for my time invested, and more important, I got my start in the industry and moved on to bigger and better things.

But is it possible to make a game easy to get into and understand while trying something new at the same time?

> It's hard to define it—this is sort of a "gut feeling" thing. I think *Tropico* was pretty good with this. The concept was quite novel (playing a Caribbean dictator, à la Fidel Castro). There had been few if any games set in the Caribbean or Latin America before, and our humorous Latin/Caribbean feel was very fresh, I think. At the same time, we grounded our gameplay in previous successful titles, particularly the *SimCity* series, although that was only a rough gameplay guide. But we thought the millions of buyers of *SimCity* and *RollerCoaster Tycoon* would immediately be able to "get" *Tropico* just by looking at the screenshots or seeing a little snippet of gameplay.

For more from Steinmeyer, flip to Chapters 6 and 21.

Ed Del Castillo, Liquid Entertainment

As president and co-founder of Liquid Entertainment, Ed Del Castillo juggles his management responsibilities with active participation in all aspects of product development: especially design and art direction. Liquid's first release was the eagerly anticipated *Battle Realms*. Before co-founding Liquid, Ed Del Castillo was probably best known as producer in charge of Westwood Studios' *Command & Conquer* franchise, including *Red Alert* and its expansion packs, various ports to other platforms, and foreign language versions.

Del Castillo was asked to provide three pieces of advice to a new strategy game designer looking to be successful in this industry. He responds as follows:

- Go to film school to learn how to tell a story, develop characters, light a scene, set a mood, and evoke an emotion.
- Never stop reading, watching, and playing everything.
- Consider another line of work if having a family is one of your goals!

According to Del Castillo, the most challenging obstacle when creating a real-time strategy (RTS) game is getting people to innovate:

> The RTS genre has been in stagnation for so long with only micro-improvements that it's hard for people to create stuff that's truly better and not just different. Overcoming it is a matter of vision. You have to have a clear picture of what you're trying to achieve.

If Del Castillo could create *Command & Conquer (C&C)* all over again, would he do anything differently?

> That's tough because *C&C* taught me so much. I think that I would change two things:
>
> - A less-limiting universe. *C&C* is very limited, which is why you see things like *Red Alert* and *Emperor*. They're both attempts to use the *C&C* model in other worlds.
> - More controlled presentations prior to ship. A few people, who shall be nameless, went on to great success due to their good memories and timely use of video cameras during our presentations prior to our release.
>
> It had its place and time, and I'm having a hard time messing with that. The original *C&C* is my third favorite RTS after [Blizzard's] *StarCraft* and [Bungie Studios'] *Myth*.

Any pet peeves with today's strategy games?

> Yes, too much iteration, not enough innovation. We need fewer people just trying to squeeze all they can out of an engine and more people crafting emotional experiences.

As a programmer, Del Castillo offers these words for others:

> Think before you act. The days of the hacker are over. Think modular, flexible, expandable, well-organized, and well-documented code. Truly grasping these tenets will make you incredibly wanted by everyone.

On artificial intelligence (AI):

> I think the biggest mistake you can make, and not just in AI, is to believe your own marketing. There's no such thing as a "learning" AI yet. Everything is still incrementing and decrementing weighting variables on algorithms based on specific and predefined data. AI is hard because most games aren't in the oven long enough to bake up a proper one. I think the best AIs are written for people who understand how little time they have and work to distill the decision-making tree of a human down to the essentials for that given game. I'm afraid that the only tip I have is to study the player. Most of his decisions can be simple to understand and emulate.

Del Castillo gives some examples of his past or present work that reinforce what he's suggesting here on innovation and on adding an emotional layer to a game:

- ➤ Innovation. When we conceived this game, it came from our ideas. We all came together and designed an RTS with no restrictions, no legacy, no need to be in the same universe as a predecessor. We were unbound by the past, and that allowed us to more fully reach for an ideal. It's the way it should be done. Too many people are held back by what they've already created and the desire to get more money out of it in the form of an incrementally improved sequel.

  Battle Realms (BR) endeavored to innovate in two major ways. The first is combat. First-generation RTSs did a great job of showing the potential of this genre, but inadvertently they were very production-oriented. Building was more important than fighting. In *BR*, terrain matters. Height makes a difference to combat effectiveness. Real line of sight makes reconnaissance, sneaking, and ambushing a real element in the game. The unit dynamics focus on the combat dynamic and "incentivize" spending more of your attention commanding the battles rather than landlording the village.

  The second major thing is to create more of a living world. There are a number of prongs to this, but it basically revolves around distilling more elements of reality and turning them into fun elements for the game. We incorporated a Living Resource System, which allows the player much more freedom. Horses are gathered and harnessed as pack animals or war steeds; water is used to put out fires, grow rice, and quench the thirst of peasants. We've tried to connect everything in some way, like the real world.

- ➤ Inspiration. It's about what I love, not what makes money this week. The inspiration comes from my childhood. Way too much *D&D* and the like mixed with way too many Kung Fu movies. I've been a game master for paper RPGs since I was 14, so the world-creation part is definitely something I love.

- Living world and the Living Resource System. This affects things in three ways:
    - It makes things easier to understand. It feels right when the men run at each other, when the fire burns down a building or catches another building on fire or is put out by your peasants. Players of RTSs have learned a "vocabulary" of what they can and can't do and what things mean. We're allowing them to do more, and that means changing how things work a bit.
    - It connects everything. Water, as an example, can be used for many things—put out fires, domesticate horses, quench the thirst of peasants toiling to build a building or soldiers training, even make the rice grow faster. Connecting things in this way not only feels more "natural" or "right," but also allows the player to easily shift his strategy and adapt to the enemy.
    - It brings the world to life. We have units with more animations than ever before. Combats look alive as the units do different types of attacks. Birds fly from the trees when men move through the forest. Vultures gather on the battlefield. Soldiers juggle or clean their weapons when left alone too long, and it affects their abilities!
- Removal of artificial genre boundaries with the introduction of what other people would call "RPG elements," but what I call "character-investment elements." Genres are a thing of the past. Bringing in some of the growth and motivations from RPGs is very natural and allows for new cool possibilities in the game, to feel natural, and leave you asking, "Why couldn't I do this before?" It's all part of bringing the battlefield more to life. Allowing you to customize the characters is the first step to giving them individual characters and allowing you to fall in love with them. If you fall in love with them, you won't want them to die—thus creating the battlefield drama.

Ed Del Castillo offers some helpful advice on breaking into the industry in Chapter 21.