

Chapter 3

Gameplay

Now we're going to take a closer look at some of the things we mentioned in the last chapter. In particular, we'll be examining gameplay—what it means, how to achieve it, and how to enhance it.

Strictly speaking, game theory is a branch of economics in which systems governed by rules are mathematically analyzed to determine the payoffs of various end points. The strategies required to reach specific endpoints are collectively termed *gameplay*.

This is not quite as limited a definition as it sounds. Morality can be considered as an economic question—the costs and payoffs of an altruistic strategy, for example, being quite different if you are within 500 feet of the top of Everest as opposed to an English village. Getting to an important interview after you overslept could involve gameplay. In the movie, *A Beautiful Mind*, Russell Crowe (playing John Nash, a pioneer of game theory) applies gameplay to the choice of which of three women he should try to chat up.

However, for all its many applications, gameplay is only one element in the composition of modern games. In *GTA Vice City*, it isn't some careful mathematical assessment of payoffs that makes you choose which kind of car to drive or which radio station to listen to. You do it because it's fun.

Hence there is nothing special about gameplay. A software product doesn't have to involve gameplay to be entertaining. Famously, *SimCity* has been described by its creator as more

KEY TOPICS

- Gameplay means interesting choices
- The use of strategies
- Gameplay is just a subset of interactivity
- Different kinds of interactivity

of a toy than a game. And the foremost aim of simulation war games is to recreate the reality on which they're modeled. If you have a pure Conquistador sim, it may not be a good well-balanced game at all (it may not be possible for the Aztec player to win) but it could still be fun to play.

Only precedent makes us expect gameplay in entertainment software. No one demands the same thing of a movie or a book. Plenty of things are enjoyable that aren't games. Who's going to say, "Hey, that Hamlet is a great drama but where's the gameplay?"

So then, rule number one is still that your product should be fun. That's where the entertainment part comes in. To be a "good" software product implies it will also be interactive. That's what will make it deliver something that you could not deliver in any other medium. But gameplay? That's a matter of choice.

Later in this chapter we'll discuss some other kinds of entertainment software that are interactive (certainly) and fun (hopefully) but aren't necessarily games at all. This chapter focuses on gameplay because it can be objectively analyzed. That's all. You can *learn* how to design a good game system the same way you can learn to lay bricks or mix cocktails. But be aware that there is a lot more than game design to creating great entertainment software. The bigger part of it is an art form.

What Is Gameplay?

Imagine you are playing a role-playing game. In this game, you play a group of adventurers—say a knight, a priest, a dwarf, and a thief. During an encounter, you typically want your fighters—that's the dwarf and the knight—at the front of the group while your thief snipes from one side with arrows. Your priest, who is vulnerable, stays at the back to cast spells.

Now, suppose the priest has two kinds of spells, each of which cost him the same number of magic points. One spell injures the enemy (we'll call those "E-Bolts"), and the other heals injuries to your own group (we'll call those "Band-Aids"). Which should he cast during a fight?

First off, suppose the E-Bolts do as much damage as a sword blow and that the Band-Aids heal the same amount. When facing opponents who are equally matched with

your own fighters, E-Bolts and Band-Aids are equally useful. Against a single foe using several weapons (an insect creature with a sword in each of six hands, for example) you are obviously better off using E-Bolts. When facing a horde of opponents with individually weak attacks, such as a pack of rats, you are probably going to use Band-Aids. The point is that in each case you can easily decide which spell is better.

However, suppose instead that Band-Aids still only affect a single target, but E-Bolts are area-effect spells that damage all the enemies in a given radius. And suppose also that E-Bolts don't do quite as much damage as before, but the target's armor no longer makes a difference. *Now which is the best to use?*

There's no easy answer. It depends on lots of things. That makes it an interesting choice. And that's what gameplay is all about.

Implementing Gameplay

Sid Meier said, "A game is a series of interesting choices." To be worthwhile, gameplay choices must be non-trivial. Each strategy that the player considers using must have an upside and a downside. If there is only an upside, the AI should take care of it automatically. If there's only a downside, no one will ever use that strategy so why bother including it in the game?

Remember that we are only talking about game theory here. You might put in something that's just intended to be fun. A ray gun that plays a little tune whenever you shoot something, for example. But even so, if that ray gun doesn't have some gameplay value—if it doesn't have *some* upside, even if that upside is only a wide variety of catchy tunes—then pretty soon players will stop using it, which means all that development effort was spent just so players would say, "That's cool—now hand me the BFG."

A decision has gameplay value when it has an upside and a downside *and* when the overall payoff depends on other factors. Usually this will be something along the lines of "What if the other player uses neutron torpedoes?" or "What if I'm too close to the sun when the warp drive kicks in?" So you choose the option that seems best for you in the circumstances. It might be that you can see a better tactic but you simply don't have the manual dexterity to carry it off. That's still a game choice—albeit one that is likely to have hardened strategy gamers grinding their teeth.

Also it's important to note that the game must be a *series* of interesting choices. Each decision affects the next. The value of using the nail gun as opposed to the mining laser depends on whether your previous decision was to run to the center of the room or skulk in the corner. That depends on whether you set the cargo bay doors to blow or not, and so on. Hence a game allows strategy. Indeed, a well-designed game cannot be won without strategy. And strategy manifests itself as a series of interesting choices. We've come full circle.

All this requires the game to display complexity. This doesn't mean, however, that the rules themselves must be complex. In the last chapter we looked at the concept of emergence—complexity arising from simple rules. A neat example of emergence that we like is the termite's nest. Termites build towers of dried mud with cooling vanes inside that keep the nest at the precise temperature needed to incubate the eggs. But no termite has a blueprint of this wonderful structure in its head; it only has a few simple rules to follow about where to put each bit of mud.

Emergence results from rules (behaviors) interacting with other rules or with the environment. Resist the temptation to design in too many rules. The best games follow the less-is-more principal. In the rest of this chapter, we'll discuss how very often the simplest rule sets are the ones that lead to the most interesting gameplay.

The Dominant Strategy Problem

How often have you flicked through gaming magazines and seen articles promising “10 killer tactics” or claiming to disclose the ultimate character, weapon, or maneuver in your favorite game?

If those articles are on the level (as they too often are), what they are doing is taking advantage of flaws in the game design. A well-designed game shouldn't contain an option that is never worth using. So it certainly shouldn't have a best maneuver, or character, or weapon—otherwise, what was the point of the other maneuvers, characters and weapons?

To borrow some terminology from game theory, an option that is never worth using under any circumstances is a *dominated* strategy. An option that is so good it's never worth doing anything else is a *dominant* strategy.

Dominance means some of the options that you designed into your game—and maybe even all but one of the options—are useless. If they're useless, it means that pretty quickly

players will drop them and won't use them again. Not only is that bad game design, it was a waste of development time as well. The only people happy to see dominance in a game are the magazine columnists—because it makes thinking of strategy tips a lot easier.

Summing up, then: A dominated option is worthless. You wasted your time putting it in your game. A dominant option is worse. It means that *all* the other options are worthless.

Near Dominance

Looking for dominance warns you if options will rarely be used. It's worth taking some time to look for *near-dominance* as well. A near-dominated option is one that is only useful in very narrow circumstances. Conversely, a near-dominant option is one that players will use most of the time.

Let's consider an example of a near-dominated option: You put a special weapon in your game, a stun gun that is only worth using against one foe, the raptors. Since the stun gun has no effect on other foes, you won't see it used except on the raptor level. This is obvious, but still it's valuable information. It means you can ask yourself these questions:

1. Did I want the stun gun to be a more common feature?
 - ▶ Should I think of other applications?
 - ▶ Is the one-off use a positive feature?
What should I do to make the most of that?
2. If the stun gun is going to be used only once during the game, how much development time is it worth spending on it?
 - ▶ Should we spend a lot of effort on this feature?
 - ▶ Is it worth putting in some special effects just for the raptor encounter?
3. What's the best way of building the stun gun into the interface, as compared with weapons that are more commonly used?

A game with lasting appeal must remain interesting even when players know the tricks. Near-dominance isn't as disastrous as dominance. A careless player could overlook the near-dominated option, and opponents can exploit that. But near dominance

does show which options will get used most, and logically those are the ones on which you'll want to lavish the best graphics.

Avoiding Trivial Choices

Good gameplay is achieved when the player faces problems that have non-trivial solutions. But how do you avoid designing in trivial choices?

For this example we'll be talking about units in a wargame, but they could just as easily be different maneuvers, weapons, spells, or whatever. The principal is the same. Say that the three units are warriors, barbarians, and archers. Figure 3.1 shows one possible way that the combat relationship between them might work.

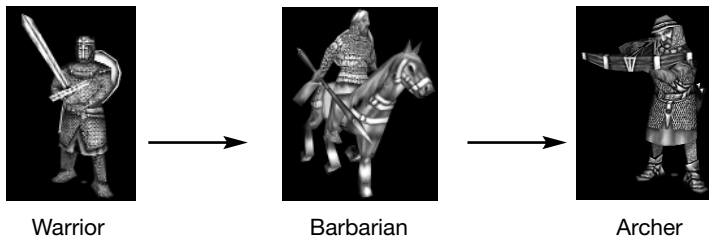


Figure 3.1 Transitive combat relationship.

This is a transitive relationship. Warriors are best, then barbarians, then archers. Now consider the intransitive version in Figure 3.2.

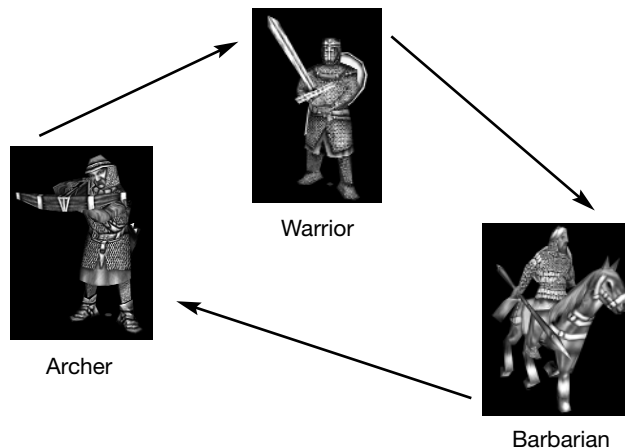


Figure 3.2 Intransitive combat relationship.

Warriors still beat barbarians, and barbarians beat archers, but now archers can beat warriors. It's a little bit more interesting. In fact, it ought to remind you of something familiar. That's right, it's Paper-Scissors-Stone. The example in this case comes from Dave and Barry Murray's 1984 game *The Ancient Art of War*. In that game, the warriors were stronger than barbarians at close quarters, but they were slow. The barbarians, on the other hand, were fast enough to close with the archers without taking much damage from their arrows.

There's more to be learned from Paper-Scissors-Stone, and we'll analyze it rigorously in Chapter 5, "Game Balance." For now, though, it's worth emphasizing that a non-transitive relation like Paper Scissors-Stone is only the *minimum* condition to get interesting choices. It ensures a dynamic equilibrium, which keeps the game from stagnating, but it's really just first base.

To illustrate this, suppose you're putting a Paper-Scissors-Stone kind of set-up into your game and you decide to hardwire it. That is, A beats B, B beats C, C beats A—in each case according to some look-up table that applies in all circumstances. If you did it that way, you would need to decide the values for the look-up table—how *much* better is each unit than the next one round the chain?

Figure 3.3 shows one option, in which a single warrior can beat any number of barbarians.

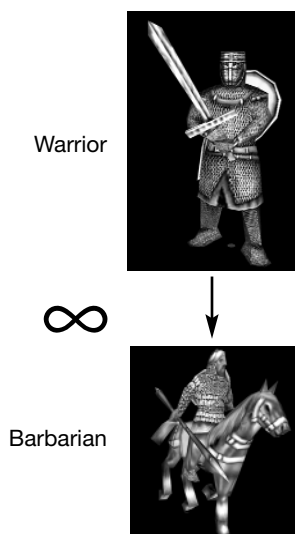


Figure 3.3 Absolute superiority.

And Figure 3.4 shows the opposite extreme. Now, the warrior is just a tiny bit better than the barbarian, so 99 warriors are an even match for 100 barbarians:

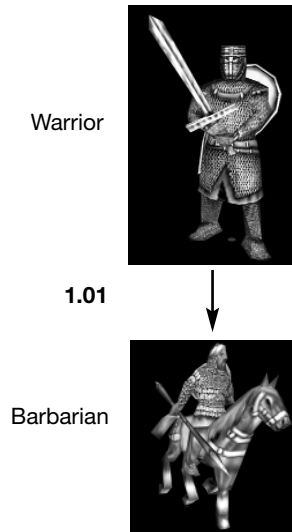


Figure 3.4 Marginal superiority.

Which of these is better? Don't waste time thinking about it; it's a trick question! *Neither* of them leads to interesting gameplay. In the first case (assuming all units have the same cost) armies will have to consist of an equal mix of As, Bs, and Cs; all other combinations will lose. In the second case, a player loses very little by fielding the wrong unit and so the unit interaction becomes virtually redundant.

So you can't hardwire in a single rule and hope for interesting gameplay. It just won't happen. What's needed is a dynamic relationship between different strategies so that in some cases A is much better than B, and in others there's little to choose between them. And sometimes maybe the worm can turn, and B can beat A.

Paper-Scissors-Stone is a first step in avoiding trivial gameplay. It means there isn't a single optimum strategy that wins every time. However, there *should* be an optimum strategy for each situation; "situations" in a well-designed game being complex because they include not only terrain, altitude, weather, time—the obvious things—but also whatever your opponents are doing. Case Study 3.1 provides an example of gameplay.

Case Study 3.1 Environment Plus Rules Equals Gameplay

At the Battle of Hastings in 1066 AD, King Harold of England had the dual problems of an exhausted army (having just force-marched the length of the country after another battle) and an enemy with a better mix of troop types. Duke William of Normandy had more and better archers, as well as cavalry in the form of his Norman knights. Harold, by contrast, was obliged to rely on infantry. The mainstay of his army, his housecarls with their massive axes, were formidable fighters but lacked the mobility for an offensive battle.

Knowing this, Harold drew up his troops on Senlac Hill, near Hastings, and awaited the Norman onslaught.

Under normal conditions, the Norman archers could have skirmished against the unsupported infantry and won eventual victory. Their volleys would pick off targets and, whenever the lumbering infantry in their heavy mail hauberks came too close, the archers could always run away.

That was the classic theory of early medieval warfare. On this occasion, Harold had found a way to refute the theory. Shooting uphill, the archers' volleys proved less effective. Harold's men were able to stand their ground, their kite shields interlocked to form a defensive wall around them. Even so, initially some of the arrows found their mark. Now and then one of the English soldiers would fall. The housecarls yearned to charge down the hill, but Harold knew that the enemy archers would counter that by retreating, and then would have the advantage. He ordered his men to stand and, as the day wore on, the Norman archers grew tired. Their arrows carried less force. They fell harmlessly among the housecarls, like twigs.

Harold was wise in warfare, but Duke William too had some tricks. Ordinarily, William would not have ordered his cavalry against the English shield wall. Horses will not charge onto a solid line of infantry (as long as the infantrymen know this and have the courage to stand firm). Harold's army did not lack for courage, William knew; they had the opposite problem. He sent his knights in wave after wave against the English line. Each time, the knights would wheel around at the last moment, or take a few lance-thrusts against the shields of the English footmen, and then ride off.

continues

continued

The English housecarls had suffered arrows raining down on them all day. They had little patience for a long wait and liked much better to be in the thick of battle. Now they had to watch Norman knights turn their backs and flee. Many broke from the line and charged.

Off the hill, the conventional theory of combined arms prevailed. Archers could inflict injury on infantry who lacked the mobility to close with them. If the infantry charged in an effort to catch the archers, enemy knights could ride in amongst them and wreak havoc.

If things had been different, could Harold have won? Who knows? The point is that he understood that there are ways to change the balance between different troop types. A good commander isn't the one with the best army; he is the one who knows best how to use it.

Ensuring Interesting Choices

Elite was one of the seminal games of the early '80s. The aim of the game was to accumulate wealth in the form of credits by trading between planets. When a player had earned 1000 credits, he could trade in his pulse laser and get a beam laser and 400 credits in exchange. The 400 credits kept the player in a good position to trade, and the beam laser was far superior to the pulse laser. Therefore, the player would regard the decision to upgrade as a no-brainer. The curious part of the choice was that you had to have 1000 credits before you could spend 600 on a beam laser. Had the beam laser been available as soon as you had 600 credits, an interesting choice would have been created: whether to upgrade right now (but have no credits left to trade with), or carry on trading armed with a substandard weapon.

In the context of a great game, it was a small enough flaw, but a highly illustrative one. A difficult choice had been taken out of the hands of the player.

Here are the kinds of choices that gameplay can involve:

- ▶ An option that should sometimes be taken, and sometimes not, depending on other factors
- ▶ An option whose timing is critical and depends on the context

- ▶ An option that makes little difference whether you take it or not
- ▶ An option that is always worth taking
- ▶ An option that is never worth taking at any point in the game

The first two are evidently the most interesting in a gameplay sense. The third kind (choices that don't make much difference) are perfectly valid but they are chrome and should be recognized as such. Options always worth taking should be handled by the AI. For instance, in most third-person action games (*Tomb Raider*, *Enter the Matrix*, and so on) the AI auto-targets the nearest enemy because, nine times out of ten, that's the one the player will need to shoot first. Lastly, options that are never worth taking will be fun once only (if that) and thus ought to be seriously considered before you waste developers' time on them.

A Toolbox of Interesting Choices

Interesting choices are those that require good judgement on the part of the player, which means that the correct choice must vary with the circumstances. Your aim as designer is to ensure that those circumstances will never stagnate to the point that there is only one right way to win.

There is no method for finding the best and most intriguing choices. That's where the creative process comes in. (We said at the outset that it's a black art.) Even so, you can follow a number of tips to ensure the gameplay choices aren't trivial.

Strategic Choices

Strategic choices are those that affect the course of the game over the medium or long term. (This is distinct from tactical choices, which apply *right now*.) Strategic choices will have a knock-on effect on the player's range of tactical choices later, so they are a prime means of enabling good gameplay.

StarCraft is a typically well-designed game from Blizzard. One of the choices in the game is whether you should first upgrade the range of your Marines' rifles, or upgrade the damage the rifles do, or research the StimPack which will allow the Marines to fire faster. This decision requires some thought. If you are expecting a heavily armored foe such as a Protoss Zealot, you will probably want the extra damage—whereas lots of fast-moving opponents like Zerglings call for different tactics. Other factors that will

influence your choice are how many Marines you have, what the surrounding terrain is like, whether you have bunkers built, and whether you anticipate an offensive or defensive campaign.

Warzone 2100 includes a beautiful example of strategic decision making. The player needs to build manufacturing plants to spawn his war machines. Optionally, he can build these manufacturing plants in the current level—in which case they deliver units quickly but only for use in that level—or he can build them back at his main base, which means that new units arrive more slowly but the plant can be used to provide units in later levels too.

Sometimes one option will be called for, sometimes another. A player who selects options according to a set plan will not do as well as one who adapts to circumstances. The lesson is that, to create good gameplay, different choices should lead to different *kinds* of payoff. This way, you reduce the risk that choices will be trivial, and increase the scope for good judgment.

Supporting Investments

Often a game has not only a primary objective but also secondary aims that have to be attended to before you can reach the final goal. For example, the primary goal might be to destroy the enemy, but to do that you might need to build farms to produce food to spawn peasants to trade to make money to recruit soldiers and so on.

Some expenditure directly achieves your eventual goal. In a wargame, an example might be buying a mercenary. Other types of expenditure indirectly work towards your goal, and these are called *supporting investments*.

Primary supporting investments operate at one remove. For instance, improving weapons technology to make your mercenaries tougher, or building an extra barracks to house mercenaries.

Other supporting investments work more indirectly. For instance, building a smithy so that weapons can be upgraded, or researching construction techniques so that barracks can be built more quickly.

Note that researching construction techniques helps at both the second and third remove: It helps you build barracks to attract mercenaries, and it helps you build smithies to research technology to improve mercenaries. This kind of complexity is excellent, because it makes the decision to research less simple and more interesting.

By including decisions that operate at different levels, you create the need for players to think strategically. The payoff for choosing different investments will alter according to what other players do.

Versatility

A useful rule-of-thumb for anticipating gameplay is to ask what is the best and worst thing about each of the player's options. For instance:

- This maneuver does the most damage, but it's the slowest.
- This maneuver is the fastest, but it leaves me defenseless.
- This maneuver gives the best defense, but it does little damage.

And then there's a unique kind of choice:

- This maneuver is never the best or the worst, but it's the most versatile.

So, a useful question to ask yourself when designing a weapon or strategy for your game is "When, if ever, is this the best option for the player?" Most choices that you put into the game should be the best in some way. And one of these can be the choice that works only moderately well, but in many different ways: the jack-of-all-trades option.

The more unpredictable the game environment, the bigger the payoff for having versatility of choice. Beginners in particular will benefit from versatile options in a game, as it means there's something they can do while working their way up the learning curve. But versatile options are handy for expert players too. When fighting an expert opponent, you must expect the unexpected, and choosing the versatile maneuver or unit may buy time to put together a more considered response.

One obvious kind of versatility is speed. The fast moving character or unit can quickly go where it's needed. So, normally, you won't want the fastest units to also be the best in other ways.

Also, the value of a fast-moving unit depends on the game environment. On the battlefields of the 14th century, a knight was deemed to be worth 100 foot soldiers. That wasn't because knights were each individually as tough as 100 men, but rather because in a terrain of hedgerows, ditches, ploughed fields, and heathland, the knight had more chance of being at the right place at the right time.

There are many other ways to make an option versatile. If a beam weapon can be used to mine asteroids as well as to destroy incoming nuclear missiles, then that versatility can make up for a disadvantage elsewhere. Of course, if there is no compensating disadvantage, there's no interesting choice. Be careful not to make the versatile choice dominant over all others. Also, be aware that the versatility of a choice may not be obvious even to you as designer. In the last chapter, we saw how the designer of the fantasy game *Arena* hadn't originally anticipated the way players might use the fireball spells. Case Study 3.2 provides an example of unexpected versatility.

Case Study 3.2 Unexpected Versatility

In 1997, Dave Morris was working with Jamie Thomson on *Abraxas*, an on-line CRPG for a major UK publisher. One of the things we wanted was for the game world to have its own laws for the players to discover and exploit. As Jamie put it, "We might suddenly find that a player in Ohio has invented the *Abraxas* equivalent of steam engine."

One of the factors we were considering was combustion. The temperature scale ranged from -30 (equivalent to absolute zero) up to +1000, with 0 being comfortable room temperature. Materials had an ignition temperature, a burning temperature (within the flame) and a radiative temperature (adjacent to the flame). So oil would ignite at +6, the temperature of its flame was +4, and the flame modified the ambient temperature by +1. (The range of that effect depended on the size of the flame.)

In one playtesting session, players were transporting barrels of phlogiston, a magical substance that burns with a cold flame. They got caught in a blizzard and the temperature dropped to -8, which was definitely life threatening. Now, the properties of phlogiston were that it ignited at +5, burned at -5 and had a radiative temperature of -1. How could they use it to save themselves? A phlogiston fire

couldn't warm them because it would merely reduce the ambient temperature by a further -1, taking it down to -9.

Then somebody realized that the temperature within the flame was constant irrespective of the environment and because phlogiston burned at a negative temperature, it wouldn't ignite them or their belongings. So they started a fire and survived the night by sitting inside the flames at a cold but tolerable -5. They had found a versatile solution that we as designers had never expected.

You can measure versatility by looking at the *switching costs* in the game. This is how much it costs a player to change his mind about the strategy he's using. An example in an espionage game might be if you recruit a spy and later realize you need an assassin instead. The switching cost is however much you wasted on the wrong character, assuming for the sake of argument that the spy is not usable elsewhere. So, say that both cost \$1 million. When deciding which to buy, at first you'd think, "If I buy the spy and I need the assassin, I'll end up paying \$2 million. If I choose right, it costs me just the \$1 million. On the other hand, suppose I buy both now—I only need one, so I'll definitely have wasted \$1 million."

Now suppose there is another character, the ninja, who can function as either spy or assassin. How much should the ninja cost? It depends how unpredictable the game is. In this example, if the game were completely predictable, the player would know in advance which character to recruit and so versatility is of no value—the ninja should cost \$1 million just like the others. In a completely unpredictable environment, the average cost would be \$1.5 million (\$1 million if I choose right, \$2 million if I choose wrong), which is what a good gambler would pay you for the ninja. Since the truth will lie between those extremes, the versatile unit should cost more than \$1 million but less than \$1.5 million.

Versatility is more prized in an uncertain environment. No multiplayer game is completely predictable, since you can never know what the other player(s) will do. Even in a relatively predictable game, some levels are more uncertain than others. All of which makes the choice between specialization or versatility an interesting one because it all depends on the circumstances.

Compensating Factors

Consider a strategy game that includes an aerial unit capable of crossing any terrain. The aerial unit is the best at getting places because no other unit can cross all terrain types—ships can't go on land, mountaineers can't cross oceans, camel riders are slowed by forest, and so on. But this unit can go anywhere. How can we as designers balance that?

- ▶ We could make it slow.
- ▶ We could make it weak (for example, easily destroyed).
- ▶ We could give it low surveillance range (rather unlikely).
- ▶ We could make it expensive to buy.

The last of those isn't so good, though, because it doesn't oblige the player to use the unit in an interesting way. He either buys the unit or he doesn't, and once he's made that choice he's committed. The choice whether to buy might be strategically interesting, but it won't lead to clever tactics.

These are all compensating factors. They ensure that any game choice has something in its favor and something weighing against it.

Compensating factors only work when it is clear to the player what they are. Maybe he will decide that the powerful unit is worth having despite the fact that it's so expensive because he intends to make a big push and grab the oil fields. The player's plan might fail, but the choice is his and it is an informed choice. Case Study 3.3 provides an example of how to balance compensating factors.

Case Study 3.3 Balancing Compensating Factors

Shapeshifter is a proposed action-adventure game that allows the player to take the form of various creatures. Each creature has unique abilities. To ensure a degree of strategic choice, Martin, the designer, builds in a cost in magic points each time the player changes shape.

Martin is describing one of the creatures at a development team meeting. “The werewolf is by far the strongest character in the game when the moon is full, and he’s invulnerable to anything but silver bullets, but he’s only as tough as a normal man at other times.”

Sandra, one of the level designers, is skeptical. “If we have a full moon level, which I assume we will, then any player who has the points to transform into a werewolf will always do it. And on other levels you can see how long till full moon, so you can save up your points. It’s a no-brainer. Is that good gameplay?”

“It’s not quite that simple,” says Martin. “First of all, didn’t we say we can have weather effects? If the moon goes behind a cloud, or if another player casts a fog spell, that weakens the moonlight and the werewolf isn’t so useful.”

Sandra isn’t entirely convinced. “Well, the weather effects are random, so that makes changing to wolf form just a gamble rather than a gameplay decision. I can see the point if other players are able to create fog, though.”

“What about the silver bullets?” points out Kay, the software planner. “Either another player will have one—in which case the werewolf is dead—or not, in which case the werewolf wins hands down.”

Martin considers this. “Obviously we don’t want it all decided from the start of the level. So there will have to be a silver bullet somewhere on the full moon level, and finding it has to be difficult.”

“It still comes down to another gamble,” points out Sandra. “Do I spend a lot of points changing to a werewolf, or go looking for the silver bullet and hope I find it before the werewolf player finds me? It’s just luck. I’m not sure these are interesting choices.”

continued

“You know what?” said Kay. “It’s this simple. The silver bullet is going to be very effective against the werewolf, which means it needs to be less useful in other ways. It might be that it does less damage to other characters, or it might be more expensive. But simply making it rare or hard to find is a bad way to balance it, because that takes the choice whether to use it out of the players’ hands.”

From a gameplay perspective, compensating factors are worthless if the player only finds out about them *after* making the decision that they apply to—for instance, if the player has to choose a commando squad before seeing the mission they are to undertake. Maybe the arctic survival expert’s many strengths are counterbalanced by not having the disguise and fast-talk skills of the CIA man. But all that counts for nothing if it turns out the mission they’ve been picked for is a hike through tropical jungle to blow up a bridge.

Impermanence

All decisions that pay off lead to some kind of advantage. Some advantages are long lasting: You got to the crate first, so you got the bonus. Others are impermanent: I built my base near the tiberium, sure, but you can still grab it off me.

Impermanence gives the designer another way to confront the player with difficult choices. Would I rather have pretty good armor for the rest of the game, or be completely invulnerable for thirty seconds? It depends, of course, on the circumstances.

Advantages (and disadvantages) can be impermanent in a number of ways:

- ▶ They can be destroyed by chance or by enemy action.
- ▶ They can be stolen or converted.
- ▶ They can apply to something which you don’t always have (your heavy cavalry, your jet bike, your wand of flame).
- ▶ They can have a certain number of uses (six bullets in a magazine).
- ▶ They can last for a certain time (the spell wears off at midnight).

Impermanence is of course another kind of compensating factor. But it is one that occurs so often in games, and has such special consequences, that it is worth considering as a separate category of choice.

Shadow Costs

In every game, players are continually being presented with costs and trade-offs. A cost doesn't have to mean money or victory points; it can be simply the things you had to succeed at before you could get to the options you're facing next. What is the real cost of a game choice—in terms of time, effort, attention and alternative resources required to get there? In *Warrior Kings*, players can hire mercenaries for gold. But the cost of a mercenary isn't only the gold you spent to hire him. First you had to build shops, and townsmen to trade in them, and a tavern to attract the mercenaries to your town in the first place. So you need to consider everything you had to go through to hire that mercenary. This is the shadow cost.

Note that shadow costs are related to supporting investments. Tracing a flowchart for all supporting investments will show you the kinds of strategic choices the player has to make. Summing all the factors along one particular branch of that flowchart tells you the shadow cost required to reach each node of it. Case Study 3.4 gives an example of shadow costs.

Case Study 3.4 Shadow Costs in Age of Empires

The primary resources of *Age of Empires* are wood and food. Food is pretty much an inexhaustible resource within the confines of the game, but wood—although easy to harvest—is finite.

Consider a charioteer. The apparent costs of the charioteer are 60 wood, 40 food, and 40 seconds to spawn. This unit's shadow cost, however, changes greatly over the course of the game. Early on, when the player's economy is underdeveloped, both food and wood are expensive. The forty seconds' spawning time is not so important, though. Later in the game, wood and food are plentiful and your main concern is how quickly you can pump out new units. In the end game, if there is no convenient supply of wood left, each charioteer you spawn is priceless.

continues

continued

The shadow cost of the charioteer involves other factors too, of course—like the stables you had to build, and the upgrades you’ve paid to make him a useful unit. But it is the influence of the fluctuating availability of resources that has most dramatic impact throughout the game.

A shadow cost is the underlying cost to the player of a decision. You can use the variability of shadow costs to add subtlety to your game. Vary the environment and you vary the shadow costs. This is both a challenge for the level designer and an opportunity to give the gameplay more depth that expert players will appreciate.

Synergies

Synergies are the interaction between different elements of the player’s strategic arsenal. They take four forms, as shown in Table 3.1.

Table 3.1 Synergies

Good fit = positive feedback

Economies of Scale: The more units of one type you have, the better each unit is. If your magicians draw strength from each other, then the value of a new magician is greater if you already have many others.

Economies of Scope: Of obvious use representing the advantage of combined arms (for example, infantry units should be supported by tanks). Also covers the use of complementary gadgets (such as mining lasers with mineral scoops, or a gladiator fighting with net and trident).

Poor fit = negative feedback

Diseconomies of Scale: The first unit is the most useful—after a while it’s not worth getting any more. Some investments are designed to be increasingly costly—although this sometimes disguises a drop in the real shadow cost.

Diseconomies of Scope: Mixed troops must move at the pace of the slowest, and must be given different orders. Sometimes a focused approach is better than hedging your bets and being able to do nothing well.

Ideally, all four of these synergies should be in play at once because when they interact, they make decisions of timing more critical. When carefully chosen they also add verisimilitude and make the game more immersive.

Altering the balance between different synergies critically affects the gameplay. A game with many economies of scale and scope is a game that gives positive feedback and encourages the player who is already winning. Chess is a game of positive feedback; it's very carefully balanced but even a small mistake is magnified throughout the game. Games like this tend to tip suddenly as a small advantage becomes crushing. Positive feedback in a game with chance elements also encourages gambling—having 20 units is much better than having ten, so if offered a double or quits gamble you'll take it. Monopoly displays this kind of gameplay.

On the other hand, if there's an emphasis on diseconomies then you have negative feedback. Here, advantages are increasingly hard won, and those who are losing will rarely fall far behind. These synergies create a “catch-up” factor. Such a game will tend to last a long time. It's a good simulation of certain types of warfare, such as a land war in Asia, or trench combat. How much fun it is will depend on what else you have put in to keep the player's interest.

A Final Word About Gameplay

As we said in the last chapter, by way of introduction to this whole topic, a good game is one you win by doing something your opponent did not expect and making it work. This is not simply a case of allowing for interesting choices. As the designer, you also need to make sure those choices will interact. It's not so satisfying to win a game just by out-optimizing the other guy on resource production, or whatever. I want his choices to have made a difference to mine. If the aim is just to complete the level with more points than another player scored when he tried it, then your choices and your opponent's choices aren't interacting at all. What you have then is a competition rather than a game.

Worst of all is a game where you simply have to know the right thing to do, and choices don't enter the picture at all. Then it isn't even a competition, it's a foregone conclusion.

We have to stress again that all this only applies to gameplay, and gameplay isn't everything. Many software entertainment products aren't games at all. There's no harm in that. *Grim Fandango* is a mystery story requiring you to solve puzzles; it is no more a game than a book of riddles is a game, but nobody cares because it's fun.

Interactivity

So much for gameplay; now for interactivity. Interactivity isn't gameplay. It's much more important.

Most of the time when you're playing a game, what you're doing is just that—play. You're exploring, running, jumping, or thumb-twitching furiously to unleash a barrage of bullets—which can be lots of fun, but a game that *only* has that going for it won't be fun for long. Gameplay enriches the experience because it allows you to interact and take the experience somewhere. It's the difference between idly kicking a football about in the back yard and setting up a pitch with goals and having a match.

But gameplay is only *one way* of interacting. Take a game such as *The Getaway*. If it consisted of only a detailed city for you to drive around in, that kind of game would soon pall. The first thing you want to do is be able to interact with your environment. Bump another vehicle off the road, smash the doors off a police car, run somebody down. It's violent fun—but it's still fun. Kids play the same way, building bricks and knocking them down. So you need to have enough opportunities to interact so that the setting becomes more than a backdrop; it becomes an environment.

A game like that will be better still if you can, for instance, knock over a fire hydrant and leave it spouting water. Then that pool of water makes the road slippery so that a passing car skids onto the sidewalk. A pedestrian has to dive out of the way. He hauls the driver out of the car and punches him. Now what if the driver identifies you, the player, as the source of the trouble? He points, the pedestrian looks your way, and then both characters start yelling at you. That way the interaction has created a story with tension and action with emotions running high.

When you are playing *Creatures* or *Black & White* or *The Sims*, most of what you are doing hardly involves gameplay at all. Those titles, like many other entertainment software products, aren't truly games at all. You don't mind—you probably don't even notice—because you are having fun. But if these products weren't *interactive*, you would certainly notice that. Then you might start wondering if your time wouldn't be better spent reading a book, watching TV, or throwing a frisbee around in the park. A non-interactive computer game wouldn't win out against other fun activities.

Interactivity is what computers do best. Thus interactivity, much more than gameplay, is the heart and soul of entertainment software.

Kinds of Interactivity

You can interact with a game or narrative in many different ways. Even so, computer game designers have pretty much restricted themselves to interaction with the *facts*. Their concern is whether I open the first door and get the lady or the second door and get the tiger.

Maybe this is because designers, like Mr. Gradgrind in Dickens' *Hard Times*, believe that "facts alone are wanted in life". But in focusing only on the details of action and plot, they are overlooking other possibilities.

Consider the ways that I, as a player, could potentially interact with a game:

- ▶ Affecting the game world itself, either by changing the front end settings or (in simulations and god games) as part of the game itself
- ▶ Directly controlling the actions of a character or group of characters
- ▶ Influencing a character's actions at one remove (for example, by giving him clues or weapons, like Zeus aiding a favored hero)
- ▶ Influencing a character at two removes (for example, by leading him somewhere or pointing something out for him to look at, like a Muse providing inspiration)
- ▶ Deciding who to follow, rather than what happens—an invisible observer flitting between narrative strands
- ▶ Selecting what is interesting to me personally and making the game give more time to those elements—as any child will, when told a story at bedtime

How many of those kinds of interactivity do you commonly see implemented? The first, certainly, but only insofar as I can make the game easier or harder, or change the speed. Why shouldn't I also be able to say to a computer opponent in *StarCraft*, as I can to a human player, "Hey, let's build up a big army before we start fighting!" or, "Don't attack me because I just want to have fun building a city."?

The second kind of interactivity in that list—direct control of one or more characters—accounts for pretty much everything else. You find almost no games with the other kinds listed. And even the direct control option is used in a very restricted way. I don't know of any recent games, now that we're in the era of realtime, that allow you to switch control between two characters who are not on the same side. Why not? What inviolable rule says you can only play one side in a game?

This is a failure of imagination. Computers give us the means to create new kinds of games—and other toys and entertainment genres that aren't games at all. And yet most computer games remain, still, mutant versions of other media: films, or novels, or boardgames. And, since those media have not made much use of interactivity, there has been no prior example to guide the game designers of today. Case Study 3.5 discusses one of many possible alternative models for interactivity.

Case Study 3.5 A Different Kind of Interactivity

A couple of years back, Dave was discussing various design proposals with the managing director of a large development house. One of the ideas Dave was most keen to pitch was *Dalek City*, a tie-in to a BBC science fiction series of the 1960s.

“It's set on Skaro, the Dalek planet,” he began. “The Daleks have been mutated by nuclear war and can only move about in mechanized travel machines. At the start of the game they pick up power by induction, so they can't leave the city.”

“Yep, got it.” said the managing director. “So you have to get upgrades to let them travel outside the city.”

“Well, it's certainly possible for the Daleks to get those upgrades. But you don't play the Daleks in this game.”

“What, then?” He was getting impatient. Concepts that take more than ten seconds to explain are hostile territory to managing directors.

“There are all kinds of threats to the Daleks. Various mutants live in the jungle around the city. Natural disasters like meteor showers can occur. There's another race, the Thals, who are their ancient enemies.”

“You play the Thals, then.”

“Not really. That’s the point. You don’t play anybody. You can step in and help the Thals if you want. Or you could spawn lots of mutant monsters to overrun the Dalek city.”

The managing director evidently thought that here was something familiar. “That must cost you resources?”

“No, your resources are unrestricted, up to whatever the game engine can handle. You could just send in so many monsters, raiders, and natural disasters that the Daleks would be wiped out right at the start. The point is, say you do that a few times. Then you try something different: sending in just one monster to begin with. The Daleks kill it and take it to their labs. They start researching it. Pretty soon they don’t have any trouble dealing with that kind of monster—and what they’ve learned will help them in other ways, too.”

“I see; it’s one of these Artificial Life things,” the managing director said. (It was more of a growl, really.)

“Kind of. The Daleks are prime candidates for A-Life because their psychology is so simple. They’re paranoid, inquisitive, power-hungry and they hate everything. And their society is like a type of insect hive.”

The managing director wasn’t even growling now, just giving Dave a worried frown.

Dave decided to press on: “The aim of the game, you see, is whatever you want it to be. You can just observe the Daleks going about their duties, like your own little formicarium. Or you can trash their city and watch the little buggers get stomped. Or you can test them with various threats and see how they learn and develop. It’s the cruel-to-be-kind method. Eventually you might find you’ve nurtured them to the point where they can take anything you throw at them. Played that way, the ultimate aim of the game is to make the Daleks into an opponent you can’t beat.”

continues

continued

The managing director said nothing for a long while. Dave almost thought he might be chewing it over. But then he shook his head. “Players don’t like games without a clearly defined objective.”

Inwardly, Dave sighed. Drawing another sheaf of papers out of his file, he said, “Okay. Well, in that case I’ve got this really good storyline for a first-person shooter....”

New kinds of interactivity promise a world of possibilities that we have hardly begun to explore. To fully realize those possibilities, though, we have to be prepared to let go some of the control that we have come to expect, both as designers and as players.

“Why?” Versus “What?”

Interactivity doesn’t have to be about what happens. It’s natural for programmers who are also designers to emphasize the logical details. That’s the way they see the world. But it is often more interesting to think about *why* something happens, or how it caused other things to happen.

“If you are a process-intensive designer...then the characters in your universe can have free will within the confines of your laws of physics. To accomplish this, however, you must abandon the self-indulgence of direct control and instead rely on indirect control. That is, instead of specifying the data of the plotline, you must specify the processes of the dramatic conflict. Instead of defining who does what to whom, you must define how people can do various things to each other.”

—Chris Crawford, writing in *Interactive Entertainment Design*, April 1995

Games guru, Chris Crawford, talks about process (the laws of the game world) as opposed to data (the details of what happens in the game world). It is the blinkered, data-intensive approach of most designers that has mired entertainment software in the simplest forms of interactivity. To illustrate this, think back to some of the early discussions of interactivity using digital TV. Lots of people proposed a system (which they seemed to regard as a new kind of narrative) in which the viewer could decide what the characters did. Essentially, they were talking about that unlamented genre the interactive movie.

Would that work? Imagine an interactive version of the television series *ER*. You get to choose. Does Noah Wylie obey the hospital rules, does he give the experimental drugs to the sick kid, or does he just go outside and play basketball? Is that a rewarding way to experience interactive drama?

No. It assumes that the point of all drama is simply to find out what happens next. If it's *good* drama there ought to be a lot more to it than that. As we saw in Chapter 1, "First Concept," Aristotle listed five elements of drama: plot, style, setting, theme, and characterization. It's the last two that aren't well served by the data-intensive designer. Going back to the hypothetical example, is the point of the player's choice to guess what Noah Wylie's character would really do? Then the interactivity reduces merely to an empathy test. Is it to make him do things he wouldn't normally? Then the interactivity is just a nerdy way to test the storyline to destruction. In neither case is characterization explored in any detail, and the only theme we are apt to uncover is that it's kind of weird when people behave inconsistently.

Now consider a different way we could have played our interactive *ER*. This time we don't get to choose what the characters do or say. Now all we can do is choose which character to follow around the hospital. Dr. Carter quarrels with Dr. Pratt over a diagnosis—both storm off. We follow Carter, who gives his version of events to a third character. Later we switch back to Pratt, who by now has spoken to various other characters and is pursuing another agenda. It is only later, if ever, that we may stumble upon the resolution or aftermath of the quarrel that we witnessed at the start.

This example in itself isn't a sufficient condition for interesting interactive drama, but it illustrates a couple of important points. The first is that all drama (whether interactive or not) derives momentum from the viewer's relationship with the characters. A rule often quoted in writing classes is that a character has to change and develop—which, in turn, leads to the simplistic conclusion that all stories are about redemption. In fact, characters don't have to change to be interesting. But our understanding of the character has to change. We need to see things that will build on our first impressions and, ideally, that will surprise us and force us to reassess those impressions. That's because *the viewer is part of the system*—and if that's true of non-interactive media, you can bet it's even more so in the case of games.

The other point is that interactivity can enhance enjoyment of *all* the elements of drama, not only the hard facts of the plot. In the previous example, the protagonists would behave true to character. Various themes might be revealed, whether the writers intended them or not. Our motives for interacting would be emotional as well as logical, and it would remain an immersive experience into the bargain.

The aim of all entertainment is to make the audience care. Interactivity is a new and powerful technique in the writer's repertoire for achieving that goal.