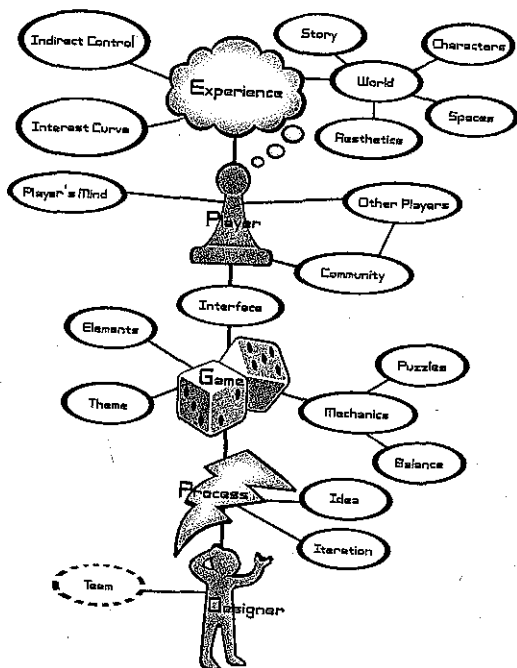


CHAPTER TWENTY-THREE

The Designer Usually Works with a Team



The Secret of Successful Teamwork

To create a modern videogame, a team of tremendous diversity is required. You need a team of people with a wide variety of artistic, technical, design, and business skills. They generally have very different backgrounds, and value very different things. But if your game is going to be a great one, they will all have to pull together and set aside differences and disagreements to make the game as great as it can be.

And there is a simple secret shared by all teams that have ever successfully collaborated to make something great. It is so simple, that when you hear it, you will likely think I'm not being serious. But this is the most serious thing I will say in this book.

The secret to successful teamwork is love.

No, really.

Now, by this, I don't mean that if the team hold hands, and sings "Kumbaya," that you are going to make a great game. I don't even mean that you have to like the other people on the team, although it wouldn't hurt.

What I mean is that you have to love the game you are making. For if everyone on the team has a deep and true love for the game they are making together, and for the audience they are making it for, all differences and disagreements will be set aside in service of bringing the game into existence and making it be as wonderful as it can possibly be.

Developers lucky enough to have been on a team that truly loves the game they are making will know just what I mean. Everyone on the team feels like children anticipating Christmas when they think about the game getting finished, and they think about that constantly.

Similarly, developers who have been on a team that had some kind of "love deficit" will also know what I mean. There are three main kinds of problems when it comes to team love for their game:

- **Love Problem #1: Team members incapable of loving any game.** Though it is difficult to understand, some people get into the games business even though they have no particular love for games or the people who play them. When someone like this is on your team, it is like carrying deadweight. They often contribute little of use, and constantly waste time arguing with team members who actually love the work. Unfortunately, the team members in charge of management or budget are most likely to have this affliction. Regardless, there is only one cure for a team member with this problem: get them off the team.
- **Love Problem #2: Team members in love with a different game than the one they are making.** This problem comes in many shapes and sizes: A level designer who only loves first-person shooters, forced to work on a role-playing game; an engineer who only loves games with cutting-edge graphics, forced to work on a simple Web-based game; and an artist who loves the work of H. R. Giger, forced to develop a new game featuring the Care Bears. When you find members of your team have this problem, the key is to work with them, to see if there is something

about the current game that they can fall in love with — or perhaps they have some idea for a new feature or element that will take the current game somewhere new and different. On the Pirates game I mentioned in an earlier chapter, we ran into a love problem early on. The animators on the team were eagerly looking forward to animating exciting pirate characters for the game. But as the design proceeded, it became clear that this would be a game about ships — the only people in it would be far away, and so tiny as to be incapable of any meaningful action or emotion. The animators tried to fight this for a while, but gradually realized it was a losing battle, and they clearly began losing their love for the game discussing it in tones of quiet detachment. Several of us on the team saw this as a major problem — we needed the animators to put their heart and souls into making beautiful effects animations, but they seemed so disappointed they wouldn't be able to animate characters, that didn't seem possible. Then, in one meeting, everything changed. One of the animators had a big sheaf of papers. "Look, I've been thinking about this game, and at first I was really bummed that we cut all the characters, but then I started thinking, the stars of this show are the ships — what could I do to make them cool?" He then proceeded to show pages and pages of sketches of how the ships would explode into pieces, how their masts would crack and break and crash into the sea, how their sails would rip and tear and flap when hit with cannonballs — it was truly inspiring to everyone. Immediately, the animators were excitedly competing to see who could come up with the coolest effects. This shift in perspective turned a project they hated into one they loved, and it made a huge difference in game quality.

- **Love Problem #3: Team members in love with different visions of the same game.** This is the most common, and the most challenging, love problem. In this situation, a team is full of people passionate about building a game, but everyone has very different ideas about what the game will be like. The key to avoiding this problem is to get everyone on the same page about what the design is as soon as possible. There will be arguments, and disagreements, but if everyone hears them out, and respectfully considers the ideas that others present, the team can work toward that all important thing — a shared vision of something that all the team members love. But it can only happen with thorough communication and respect. The moment you sense that someone in a meeting doesn't buy into an idea (even if they verbally claim they agree with it), you must stop everything, find out why, and try to find a way to get them on board. If you don't, they may secretly disagree with the direction, and lose their love for the game. And when that happens, the valuable contributions they would have made are lost. No decision should be final until the team agrees that it is final.

If You Can't Love the Game, Love the Audience

Getting others to love the game is part of your responsibility as a designer. But what about that terrible situation when, horror of horrors, you realize that you yourself

do not love the game you are working on? Again, this is not something you can ignore, or hope will remedy itself. Unless you find a way to love your game, the game you create will be mediocre at best, because the insincerity of your contribution will show through. So when your love for your game lapses, you must find a way to restore it. But how?

One way, as mentioned earlier, is to search long and hard, for something in the game that you do love — perhaps it is a moment, or a clever mechanic, or a slick interface. If you can find just one thing that you are excited about, and can be proud of, it can sometimes be enough to make the whole project worthwhile for you — enough to make you love the game and work hard to make the game succeed.

But perhaps you can't find that one thing to love, perhaps because you are not the target audience for the game. In that case, don't think of it as a game for you — think of it as what it really is, a game for the intended audience. Think of a time when you went through a great deal of preparation to give someone you love a special gift. Think how excited you were about seeing the expression on their face when they opened it up and saw it. The anticipation of this moment made you put so much thoughtful energy into the gift choice, the wrapping, and the presentation of it. You carefully designed that moment, because you loved that person, and you wanted to see that moment when they were so happy. And what was it that made them happy? Just the gift? Surely not. What made them so happy was that you loved them so much that you created that special moment just for them. The love you put into that moment shone through, and into their hearts. If you can take that kind of love and put it into the game you are creating for your audience, the love will shine through the game and into the hearts of your audience. The game will feel special to them, as they realize that someone really cared how they would feel when they played the game, and knowing that someone cares about you is a very special feeling. A designer cannot fake this — you must really feel it. As the great magician Henry Thurston once said:

Long experience has taught me that the crux of my fortunes is whether I can radiate good will toward my audience. There is only one way to do it and that is to feel it. You can fool the eyes and minds of the audience, but you cannot fool their hearts.

If even this does not work for you, if you find that not only do you not love your game, but you have no particular love for your audience, only one thing remains: To pretend. This sounds like an insincere thing to do. Didn't we just say that love cannot be faked? But something strange happens when we pretend to love things — sometimes real love starts to emerge. Have you ever been part of a group that has to do some dreary task together? Perhaps a day of spring cleaning. Everyone is dreading it and moping about it. Then one person says, half-jokingly, "Come on, everybody, this is gonna be awesome! We're going to have so much fun!" Everyone chuckles at the sarcasm, and, just for fun, starts approaching the activity with a pretend "this is going to be awesome" attitude. And just by pretending this, soon the activity does start to become fun — and ironically, everyone starts to love it. If you don't know

how to love something, just ask yourself what kinds of things someone who really loved the game would say and do, and start doing those things. You may be surprised at the transformation that starts to take place within yourself.

Lens #88: The Lens of Love

To use this lens, ask yourself these questions:

- Do I love my project? If not, how can I change that?
- Does everyone on the team love the project? If not, how can that be changed?

Again, I am completely sincere when I say that team love for the game is the most important factor determining whether the team will succeed. Love is not a luxury — it is a necessity if you are to have any hope at all of producing a great game.

Designing Together

If everyone on the team loves the project, that's great! But it gives you a new problem — everyone is going to have opinions about the design! For some designers, this is terrifying — the idea that other members of the team want to contribute design ideas threatens their status as designer, and puts them in a position where they have to argue with others about the “right” design for the game. These designers often choose to withdraw from the team, ignore these opinions, and produce a design completely independent from the rest of the team. The effect is predictable: All the beautiful ideas that each team member had for the game have been crushed, and the love they had for the game dries up and blows away. The designer becomes frustrated with the team because they seem unwilling and unable to realize his glorious vision, and the game, as you might expect, pleases no one.

A much more successful approach is to include the team whenever possible in the design process. If you can set your ego aside, you will quickly realize that most of the people on the team with design ideas don't want to hijack the game design — they just want their ideas to be heard, because they, too, want the game to be great! If you include everyone in the design process, taking every idea and suggestion seriously, you will:

- Have more ideas to choose from
- Weed out flawed ideas quickly
- Be forced to view the game from many perspectives
- Make everyone on the team feel like they own the design

When the whole team participates in the design, your game will be stronger, and everyone will embark on implementation with confidence that they understand the design. This is very important, because not all design decisions get made ahead of time. Hundreds of tiny decisions get made all the time — not by the designer, but by the programmers, artists, and executives working on the game. If all of these people have a solid, shared understanding of the game design, these little decisions will all reinforce the design of the game, and the project will have a unified robustness and solidity that it can't get any other way. It is not uncommon for many different people on a project to feel that their contribution was the most important part of the game — and not unhealthy, either! This just means that many different team members feel personal ownership and responsibility for that game. One great way to amplify this feeling is to avoid "over-fleshing" your designs. If you leave some ambiguity in the detailed design of your game, particularly for parts you aren't sure about, it forces the developers working on that section of the game to think about what that section of the game should be like, and to come up with ideas for how to implement those fine details. Since they are often closest to that part of the game, their instincts about detailed design are often quite good — and if their ideas are good ones, and go into the game, they will feel real pride of ownership of those parts of the game.

Does this mean you have to have everyone involved in the design all the time? Not everyone has the stamina to spend three hours debating the right way to lay out the inventory interface, so for detailed discussions, you will probably want to establish a core design team based on who on the team is both interested and productive at these kinds of sessions. But after this core team has come to consensus about how a design should work, you should inform the rest of the team about these decisions as soon as possible. A typical process looks something like:

1. **Initial Brainstorming:** Involves as much of the team as possible.
2. **Independent Design:** Core design team members think about ideas independently.
3. **Design Discussion:** Core design members bring their independent ideas together to discuss and try to come to consensus on ideas.
4. **Design Presentation:** The core design team presents their progress to the whole team, allowing time for comments and criticism. This often turns into brainstorming, kicking off the next round of the iterative cycle.

It takes both time and energy to involve the whole team in the design, but you will find that it makes the game stronger in the long run, provided your team is able to communicate.

Team Communication

Hundreds of books have been written about how to facilitate good team communication. I'm going to boil it down here to nine key issues that are particularly

pertinent for game design. You might think these things sound basic, and they are — but mastery of the basics is essential for excellence in any field, especially something as complicated as game design by a team. Without further ado, the nine keys to team communication are:

1. **Objectivity.** This one is listed first because it is the most likely to go wrong. In the passionate throes of design ecstasy, it is easy to become attached to an idea that struck you like white lightning from heaven. But if other team members don't like your idea, where are you then? Nowhere, if you are going to fight a war of opinions and gut feelings. The tool that will rescue you is Lens #12: The Lens of the Problem Statement. It can give you the objectivity you need. All team discussion must focus on how well design ideas solve the problems at hand. Personal preferences about these ideas don't matter — all that matters is whether the ideas solve the problem. Don't even talk about the idea as "my idea" or "Sue's idea" — speak objectively: "The spaceship idea." Not only will this separate the ideas from the individuals (giving them over to the team), but it will be clearer, as well. Another nice trick is to phrase alternatives as questions. For example, instead of saying "A is no good. I like B better," simply saying, "What if we did B instead of A?" lets the group collectively discuss the relative merits of B and A. It's a subtle difference, but much about mastering team communication is subtle. If you can develop good habits of objectivity as a designer, everyone will bring you design questions to answer without hesitation, because they know there is no danger of an awkward situation when you "pass judgment" on the design — they will just get honest, objective, useful feedback. Further, people will want to include you in every design session, because by bringing a tone of objectivity to the room, your presence can help defuse tense struggles between people taking a less objective attitude. And best of all, when a team design session has a tone of objectivity, every idea is taken seriously, which means that even shy team members will feel they can speak freely, and many ideas that might have hidden, trembling in the shadows, will confidently come to light.
2. **Clarity.** This one is simple. If communication is not clear, there is going to be confusion. When you explain something, check to see if people understand what you mean. Illustrate your ideas when possible. And if someone else says something that isn't clear, don't ever pretend you understand what they are saying. No matter how embarrassed you are, keep asking questions until you understand what they mean. Because if everyone on the design team isn't on the same page, how can there be any meaningful communication? But understanding each other is only half of clarity — the other half is getting concrete and specific. There is a big difference between saying to your producer "I'll design the combat system by Thursday" and "I'll e-mail you a 3-5 page description of the interface for the turn-based combat system by this Thursday at 5 p.m." The first throws wide the door for miscommunication, but the second gives important details about a specific deliverable, leaving little room for misunderstanding.

3. **Persistence.** WRITE THINGS DOWN! There, I said it! Verbal communication is momentary — easily misunderstood and forgotten. Things that are recorded can be checked later by everyone on the team. And you should use every persistent medium that might be useful to you — notebooks, e-mail, forums, mailing lists, flihares, wikis, printed documents, etc. Make sure someone in every design meeting is taking notes that can be shared with the team. When you do send an e-mail about a design topic, make sure to include everyone on the team. This avoids the danger of people being left out, or even just feeling left out.
4. **Comfort.** I know this one sounds a little silly. What does comfort have to do with communication? Simply this: When people are comfortable, they are less distracted and communicate more freely. Make sure your team has a place to communicate that is quiet, the right temperature, has enough chairs, and has a large writing surface; in short, a place that is physically comfortable. Also, you need to make sure team members aren't hungry, thirsty, or overtired. People who are physically uncomfortable will be terrible communicators. And physical comfort isn't enough — they must also be emotionally comfortable, which leads us to our next item.
5. **Respect.** We have discussed how the secret to being a good designer is to be a good listener. Well, the secret to good listening is to respect the person you are listening to. People who do not feel respected tend to speak little, and when they do speak, they often are not honest about their feelings, for fear they will be judged harshly. People who feel respected speak freely, openly, and honestly. Respecting people is easy, if you can remember to do it. Simply treat them, at every moment, how you would like to be treated. Don't cut them off, or roll your eyes, even if you think what they are saying is foolish. Be polite and patient at all times. Find nice things to say, even if you have to stretch a little. Keep in mind that others are more like you than unlike you — look for things you have in common, for it is easiest to respect people like ourselves. When all else fails, repeat this mantra to yourself: "What if I'm wrong?" If you somehow insult or offend someone, do not rush to defend what you have said. Rush instead to apologize, and do so sincerely. For if you can manage to respect your teammates at all times, they cannot help but respect you. And when everyone feels respected, they will communicate at their best.
6. **Trust.** Respect is impossible without trust — if I can't trust what you say and do, how can I know whether you respect me? Trust is not something that works on faith alone — relationships of trust gradually build up over time. For this reason, quality of communication matters much less than quantity of communication. People who see each other day in and day out, constantly talking, constantly solving problems together, gradually learn how much they can trust each other, and when. A group of people who barely know each other and only meet once a month have no idea who can be trusted with what. This is one area where digital communication isn't good enough — there is something in the nuance of face-to-face communication that allows us to make subconscious decisions.

about how and when to trust people. The easiest way to figure out who trusts who on a team is to observe who eats lunch together. Most animals are very selective about who they eat with, and humans are no exception. If the artists eat separately from the programmers, there is a good chance the team has pipeline problems. If the Xbox team eats separately from the Playstation team, there are often porting problems. Give your team every opportunity to be together, and to communicate together, even if it is not about things to do with your project, for the more high-bandwidth communication (about anything!) that your team can have, the more they will learn how to trust each other — this is the reason so few game studios have individual offices, preferring instead to seat teams together in open offices where they can't help but have constant face-to-face communication with one another all day long.

7. **Honesty.** Just as comfort depends on respect, and respect depends on trust, trust depends on honesty. If you have somehow developed a reputation for dishonesty in some area, even if it has nothing to do with game design or development, others will be afraid to be honest with you, which will inhibit team communication. Game development can sometimes get very political, and you will surely have to stretch the truth about some things from time to time — but your team must always feel certain they are getting the truth from you, or team communication will be strained.
8. **Privacy.** Being honest isn't always easy, because sometimes the truth can be painful. And even though we all hope to stay objective in our design work, there are times when personal pride and ego are necessarily tangled up in our work. Talking about these things honestly in a public forum can be difficult or impossible. People will tell you their true feelings in a one-on-one conversation much more easily than in public. Take the time to speak privately with each member of the design team when you can — they will often present ideas, and discuss problems they simply did not feel comfortable discussing publicly. These one-on-one conversations also go a long way to help build trust, as well, creating a virtuous circle: more trust leads to more honest communication, which leads to still more trust, and so on.
9. **Unity.** During the design process, there will be many conflicting opinions and arguments about what is right for the game. This is healthy and natural. Ultimately, though, the team must arrive at a decision everyone agrees upon. Keep in mind that it takes two people to have a disagreement. If one member of the team is stubborn on a particular point, you must treat them with the respect they deserve and work with them until a meaningful compromise can be found. Asking them to explain why this point is so important to them can often make the rest of the team understand why the point is important. When this fails, an excellent question to ask is "What would it take to bring you in?" You may not be able to settle this difference in opinion immediately, but the one thing you cannot do is ignore it. Just as a single cylinder not firing in a car engine cuts performance in half, and ultimately ruins the engine, one team member who

does not buy into the design slows the efforts of everyone on the team, and can, in the end, tear the team apart. The final goal of communication is unity.

Lens #89: The Lens of the Team

To make sure your team is operating like a well-oiled machine, ask yourself these questions:

- Is this the right team for this project? Why?
- Is the team communicating objectively?
- Is the team communicating clearly?
- Is the team comfortable with each other?
- Is there an air of trust and respect among the team?
- Is the team ultimately able to unify around decisions?

Game design and development are hard. Unless you are multitalented and your project is tiny, you can't do it alone. People are more important than ideas, because, in the words of Pixar's Ed Catmull, "If you give a good idea to a mediocre group, they'll screw it up. If you give a mediocre idea to a good group, they'll fix it."

You might think that all this team talk has nothing to do with design — that if other people on the team don't do their jobs, it has nothing to do with you as a designer. And that may be true, but it has everything to do with the game that gets created. Since everyone who touches a game exerts some influence on its design, you need everyone on the team to pull together, if the glorious vision you share is ever to come to light.

Now, with all this team communication going on, someone is going to write some documents — and that is the subject of our next chapter.