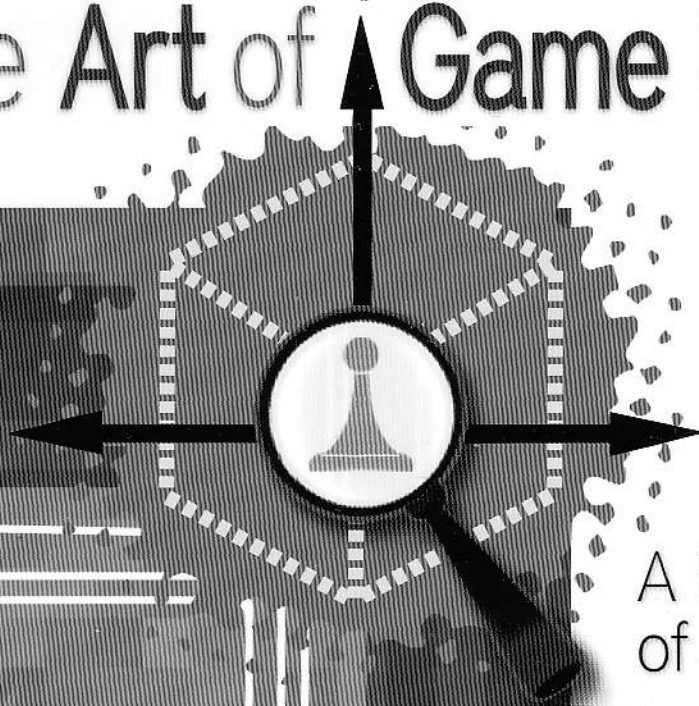




The Art of Game Design



A Book
of Lenses

Jesse Schell

MK[®]
MORGAN KAUFMANN

CHAPTER TWENTY-FOUR

The Team Sometimes Communicates Through *Documents*

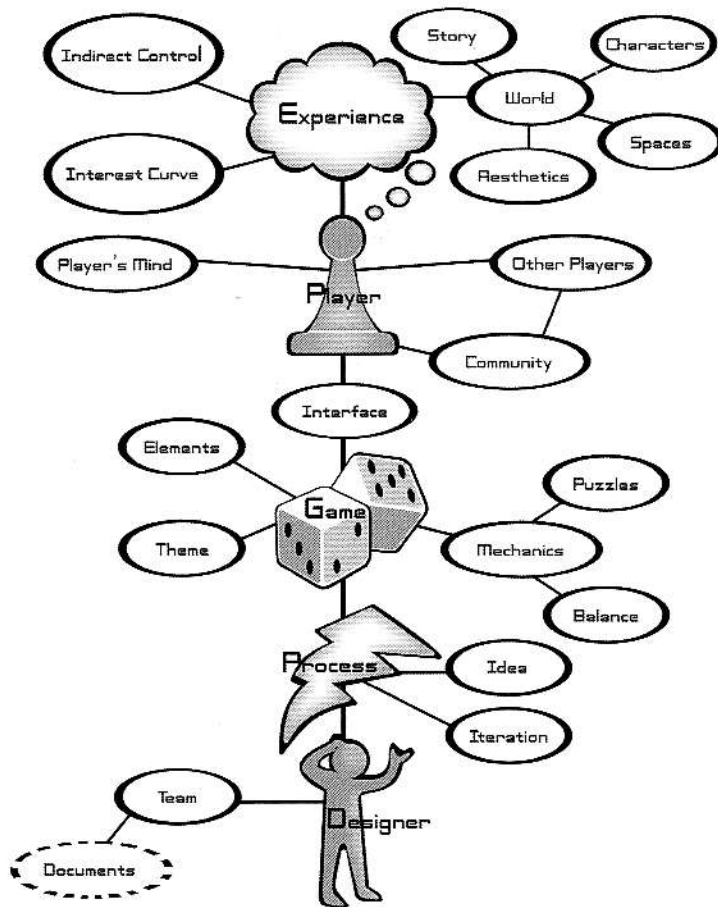


FIGURE
24.1

The Myth of the Game Design Document

Many novice game designers, and other dreamers, have an interesting vision of how the process of game design works. Not being acquainted with the Rule of the Loop, they believe that the process of game design involves a genius game designer sitting down alone at a keyboard and typing out a glorious and perfect Game Design Document. When this masterpiece is complete, all that needs to be done is to hand it to a competent team of programmers and artists and wait for them to turn this shining vision into a reality. “If only,” the frustrated would-be designer thinks, “I could find out the proper format for a Game Design Document, I could become a professional game designer too! I’m full of ideas — but without this magic template, there is no way for me to design games.”

It is very important for me to be clear about this next point, so I am going to use a very large font. Please listen closely:

The magic template does not exist!

It never has existed, and it never will exist. Does this mean that documents are not a part of game design? No — documents are a very important part of game design. But documents are different for every game, and different for every team. To understand the correct structure of the documents for your game, you must first understand their purpose.

The Purpose of Documents

Game documents have exactly two purposes: **memory** and **communication**.

Memory

Humans have terrible memories. A game design will be full of thousands of important decisions that define how the game works and why. There is a good chance you will not be able to remember them all. When these brilliant ideas are fresh in your mind, you will likely feel that they are impossible to forget. But two weeks, and two hundred design decisions later, it is very easy to forget even the most ingenious of solutions. If you get in the habit of recording your design decisions, it will save you the trouble of having to solve the same problems all over again.

Communication

Even if you are blessed with a perfect memory, though, decisions about the design of your game must be communicated to many other people on the team. Documents are a very effective way to do that. And this communication, as we discussed in

Chapter 23, will not be one-way. It will be a dialog, for as soon as a decision is put on paper, someone will find a problem with it, or come up with a way to make it better. Documents can get more minds on the design faster to more quickly find and fix weaknesses in the game design.

Types of Game Documents

Since the purpose of documents is for memory and communication, the types of documents you will need are defined by what needs to be remembered and what needs to be communicated. It is the rare game where one document serves all necessary purposes — usually it makes sense to create several different kinds of documents. There are six main groups that need to remember and communicate different things, and each generates its own special kind of documents.

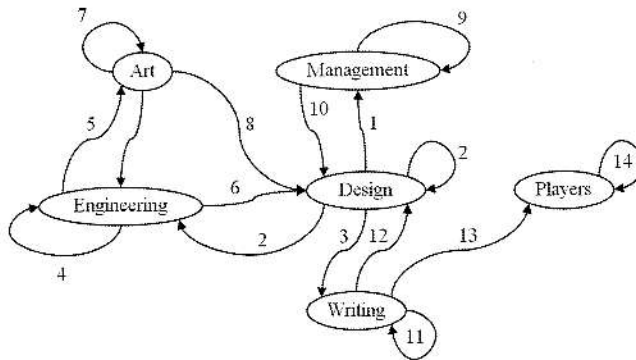


FIGURE
24.2

The figure above shows some possible paths of memory and communication on a game design team. Each arrow could be a document, or more than one document. Let's look at each of the six groups and what documents they might create.

Design

1. **Game Design Overview.** This high-level document might only be a few pages. It is often written primarily for management so that they can understand enough about what this game is, and who it is for, without getting into too much detail. The overview document can be useful for the whole team to get a sense of the big picture of the game.
2. **Detailed Design Document.** This document is the one that describes all the game mechanics and interfaces in great detail. This document usually serves two

purposes: so the designers remember all the little detailed ideas they came up with, and to help communicate those ideas to the engineers who have to code them and the artists who need to make them look nice. Since this document is seldom seen by “outsiders,” it is usually a terrible mess with just enough detail to spark discussion and keep important ideas from being forgotten. It is often the thickest of the documents, and is seldom kept up to date. Halfway through the project, it is often abandoned entirely — by that point, the game itself contains most of the important details, and the ones not in there are often exchanged through informal means, such as e-mails or short pages of notes.

3. **Story Overview.** Many games call for professional writers who will create dialog and narration for the game. These writers are often contracted, and often far away from the rest of the team. The game designers often find it necessary to create a short document that describes the important settings, characters, and actions that will take place in the game. Frequently, the writers respond to this with interesting new ideas that change the whole game design.

Engineering

4. **Technical Design Document.** Often, a videogame has many complex systems that have nothing to do with game mechanics and everything to do with getting things to appear on the screen, sending data over networks, and other crunchy technical tasks. Usually, no one outside the engineering team cares much about these details, but if the engineering team is more than one person, it often makes sense to record these details in a document so that when others join the team they can understand how the whole thing is supposed to work. Like the Detailed Design Document, it is rare for this to stay up to date more than half-way through a project, but writing this document is often essential to getting the necessary systems architected and the coding underway.
5. **Pipeline Overview.** Much of the challenging work of engineering a videogame comes from properly integrating art assets into the game. There are often special “do’s and don’ts” the artists must adhere to, if the art is to appear properly in the game. This brief document is usually generated by the engineers explicitly for the art team, and the simpler it is, the better.
6. **System Limitations.** Designers and artists are often completely unaware of what is and is not possible on the system they are designing for (or so they pretend). For some games, the engineers find it useful to create documents that make clear certain limits that should not be crossed — number of polygons on the screen at once, number of update messages sent per second, number of simultaneous explosions on screen at once, etc. Often this information is not so cut and dried, but trying to establish it (and get it in writing) can save a lot of time later — and it can help foster discussions about creative solutions to get past these limits.

7. **Art Bible.** If several artists are going to work together on a title to create a single, consistent look and feel, they must have some guidelines to help maintain this consistency. An “art bible” is simply a document that provides these guidelines. These might be character sheets, examples of environments, examples of color usage, examples of interface, or anything else that defines the look of any element in the game.
8. **Concept Art Overview.** There are many people on the team that need to understand what the game is going to look like before it is built. This is the job of concept art. The art alone doesn’t usually tell the story, though — it often makes the most sense in a design document, so often the art team works with the design team to come up with a set of images that show how they will look and feel in the context of the game design. These early images end up everywhere — in the Game Design Overview, in the Detailed Design Document, and sometimes even in technical documents, to illustrate the type of look that the technology is striving to achieve.

Management

9. **Game Budget.** While we would all like to just “work on the game until it is done,” the economic realities of the game business seldom allow this. Usually, the team is required to come up with a cost to develop the game before they completely understand what they are building. This cost is usually arrived at through a document, usually a spreadsheet, that attempts to list all the work that needs to be done to complete the game, complete with time estimates which translate into dollars. It is impossible for the producer or project manager to come up with these numbers on their own, so they generally work closely with every part of the team to make the estimates as accurate as possible. Often this document is one of the first created, since it is used to help secure the funding for the project. A good project manager will continue to evolve this document throughout the project to ensure that the project does not go over the budget it has been allocated.
10. **Project Schedule.** On a well-run project, this document will be the one most frequently updated. We know the process of game design and development is rife with surprises and unexpected changes. Nevertheless, some kind of planning is necessary, ideally planning that can change on a weekly basis at the least. A good project schedule document lists all the tasks that need to be accomplished, how long each will take, when each task must be completed, and who will do them. Hopefully, this document will take into account the fact that a single person shouldn’t do more than 40 hours in a week, and the fact that some tasks can’t be started until others are completed. Sometimes this schedule is kept on a spreadsheet, and other times on more formal project management software. Keeping this document up to date can easily be a full-time job on a medium-sized or larger game.

Writing

11. **Story Bible.** While one might think that the story of the game might be determined entirely by the writers (if any) on the project, it is often the case that everyone on the project contributes meaningful changes to the story. The engine programmers might realize that a certain story element is going to be too much of a technical challenge, and they might propose a story change. The artists might have a visual idea for a whole new part of the story that the writers never imagined. The game designers might have some ideas for gameplay concepts that require story changes. A story bible that lays down the law about what is and is not possible in this story world makes it much easier for everyone on the team to contribute story ideas, and ultimately this makes for a stronger story world that is well-integrated with art, technology, and gameplay.
12. **Script.** If the NPCs in the game are going to talk, their dialog has to come from somewhere! This dialog is often written in a script document that is either separate from, or an appendix to, the detailed design document. It is crucial that the game designers review all of the dialog, since it is all too easy for a line of dialog to be inconsistent with a rule of gameplay.
13. **Game Tutorial and Manual.** Videogames are complex, and the players have to learn how to play them somehow. In-game tutorials, Web pages, and printed manuals are how this usually happens. The text that goes into these is important — if players can't understand your game, how can they enjoy it? The details of your game design will likely continue to change up until the last minute of development, so it is important to be sure someone is continually checking this text to make sure it is still accurate with the game implementation.

Players

14. **Game Walkthrough.** The developers aren't the only ones who make documents about the game! If players like a game, they are going to write their own documents about it and post them online. Studying what your players write about your game can be a great way to find out, in detail, what players like and dislike about your game, which parts are too hard, and which are too easy. By the time a player walkthrough is written, of course, it is often too late to change your game — but at least you'll know for next time!

Again, these documents are not a magic template — there is no magic template! Each game is different, and will have different needs in terms of both memory and communication that you will have to discover for yourself.

So, Where Do I Start?

You start simply, just like you did when you started designing your game. Start with a document that is a rough bullet list of the ideas you want to include in your game. As the list grows, questions will arise in your mind about the design — these questions are crucial! Write them down so you don't forget them! "Working on your design" will mostly mean answering these questions, so you don't want to lose the questions. Each time you answer a question to your satisfaction, make a note of the decision, and why you made it. Gradually, your list of ideas, plans, questions, and answers will grow and start to fall naturally into sections. Keep writing down the things you need to remember, and the things you need to communicate. Before you know it, you will have a design document — not one based on a magic template, but one that grew organically around the unique design of your unique game.

Lens #90: The Lens of Documentation

To ensure you are writing the documents you need, and skipping the ones you don't, ask yourself these questions:

- What do we need to remember while making this game?
- What needs to be communicated while making this game?