2019

Creating Games

Cathie Leblanc
Plymouth State University

Creating Games

Creating Games

CATHIE LEBLANC

This book was produced with Pressbooks (https://pressbooks.com) and rendered with Prince.

Contents

This book is a combination of remixed and adapted material from Ian Schreiber's game design course offered during the Summer of 2009 and original material by Cathie LeBlanc.

Introduction

I have been teaching a course called *Creating Games* since 2008. It fulfills a Creative Thought requirement in our General Education program and so is focused on the creative process as it applies to game design and development. The course has evolved significantly since I first started teaching it. My thoughts about the course have evolved significantly since I first started teaching it. Since the beginning, however, the course has been project-based with a significant portion of the semester dedicated to the development of an original board game.

Over the years, I have used a variety of texts to support the course. These texts have been wonderful and helpful but as my own thoughts about the material evolved, I increasingly honed in on what I thought were the important points from the texts and supplemented and editorialized more and more, relying less and less on the text in my teaching. I also recognized that as tuition, room, and board costs have risen at my public institution, the extra cost of textbooks has become more and more of an obstacle to students being able to participate fully in their educational experiences in the classroom. I have also become interested in OER in particular and open pedagogical practices in general and the ways in which we can remove barriers to education. In Spring 2020, my class moved from being a 3 credit course to being a 4 credit course so I needed to rethink quite a bit of the structure. These various material realities converged so that I decided I would not require students to purchase a text but would instead create my own set of materials.

As I began the work of developing my own materials, I did something that I am kind of shocked I had never done before–I searched for open educational materials that others had created before me. I had been using online materials for years but they tended to come from places like the Game Developers Conference web site and YouTube channels like Extra Credits. There is some

amazing material about game design on the web. But I hadn't looked for actual educational materials. So I felt incredibly lucky and grateful to find Ian Schreiber's online experiment in teaching game design. His course focuses on developing an actual board game just as mine does. He has even put in some of the same constraints that I have–for example, both of us say that trivia games are off-limits for the game project and for similar reasons. I include most of the material with similar emphasis in my course as he does in his. I do things in a bit of a different order and with some different examples but Ian's material matches the way I teach my course remarkably well.

The major difference between Ian's online course and this text is that his course stands alone while this text does not. I wanted to create a textbook to support my face-to-face class while his material is intended to be an entire online course. That said, I will make the materials that I use in my course freely available as I develop it. As I said, the move from 3 to 4 credits requires some rethinking of the structure of the course so I will update this introduction with a link to my full course when it becomes available. In the meantime, I am happy with the text and look forward to using it with my students.

Please don't hesitate to give me feedback if you have any.

Cathie

PART I

WHAT IS A GAME?

1. Some Game Definitions

Ian Schreiber (the author of the course that much of this text is based on) says that his preferred definition of a game is a **play activity** with **rules** that involves **conflict**. But the question "what is a game?" is actually more complicated than that:

- For one thing, that's Schreiber's definition. Sure, it was adopted by the IGDA Education SIG (mostly because no one argued with him about it). There are many other definitions that disagree with his. Many of those other definitions were proposed by people with more game design experience than him. So, you can't take this definition (or anything else) for granted, just because Ian Says So.
- For another, that definition tells us nothing about how to *design* games, so we will be talking about what a game is in terms of its component parts: rules, resources, actions, story, and so on. We will call these things "formal elements" of games, for reasons that will be discussed later.

Schreiber goes on to say that the concept of a game is very difficult to define, at least in a way that doesn't either leave things out that are obviously games (so the definition is too narrow), or accept things that are clearly not games (making the definition too broad)... or sometimes both.

Here are some definitions from various sources:

- A game has "ends and means": an objective, an outcome, and a set of rules to get there. (David Parlett)
- A game is an activity involving player decisions, seeking objectives within a "limiting context" [i.e. rules]. (Clark C. Abt)
- A game has six properties: it is "free" (playing is optional and not obligatory), "separate" (fixed in space and time, in advance), has an uncertain outcome, is "unproductive" (in the sense of

creating neither goods nor wealth – note that wagering *transfers* wealth between players but does not create it), is governed by rules, and is "make believe" (accompanied by an awareness that the game is not Real Life, but is some kind of shared separate "reality"). (Roger Callois)

- A game is a "voluntary effort to overcome unnecessary obstacles." This definition sounds a bit different, but includes a lot of concepts of former definitions: it is voluntary, it has goals and rules. The bit about "unnecessary obstacles" implies an inefficiency caused by the rules on purpose – for example, if the object of *Tic Tac Toe* is to get three symbols across, down or diagonally, the easiest way to do that is to simply write three symbols in a row on your first turn while keeping the paper away from your opponent. But you don't do that, because the rules get in the way… and it is from those rules that the play emerges. (Bernard Suits)
- Games have four properties. They are a "closed, formal system" (this is a fancy way of saying that they have rules; "formal" in this case means that it can be defined, not that it involves wearing a suit and tie); they involve interaction; they involve conflict; and they offer safety… at least compared to what they represent (for example, American Football is certainly not what one would call perfectly safe – injuries are common – but as a game it is an abstract representation of warfare, and it is certainly more safe than being a soldier in the middle of combat). (Chris Crawford)
- Games are a "form of art in which the participants, termed Players, make decisions in order to manage resources through game tokens in the pursuit of a goal." This definition includes a number of concepts not seen in earlier definitions: games are art, they involve decisions and resource management, and they have "tokens" (objects within the game). There is also the familiar concept of goals. (Greg Costikyan)
- Games are a "system in which players engage in an artificial conflict, defined by rules, that results in a quantifiable

outcome" ("quantifiable" here just means, for example, that there is a concept of "winning" and "losing"). This definition is from the book *Rules of Play* by Katie Salen and Eric Zimmerman. That book also lists the other definitions given above, and we should thank the authors for putting them all in one place for easy reference.

By examining these definitions, we now have a starting point for discussing games. Some of the elements mentioned that seem to be common to many (if not all) games include:

- Games are an **activity**.
- Games have **rules.**
- Games have **conflict**.
- Games have **goals**.
- Games involve **decision making**.
- Games are **artificial**, they are **safe**, and they are **outside ordinary life**. This is sometimes referred to as the players stepping into the "Magic Circle" or sharing a "lusory attitude".
- Games involve **no material gain** on the part of the players.
- Games are **voluntary**. If you are held at gunpoint and forced into an activity that would normally be considered a game, some would say that it is no longer a game *for you.* (Something to think about: if you accept this, then an activity that is voluntary for some players and compulsory for others may or may not be a game… depending on whose point of view you are looking at.)
- Games have an **uncertain outcome**.
- Games are a **representation** or **simulation** of something real, but they are themselves **make believe**.
- Games are **inefficient**. The rules impose obstacles that prevent the player from reaching their goal through the most efficient means.
- Games have **systems**. Usually, it is a **closed** system, meaning that resources and information do not flow between the game

and the outside world.
- Games are a form of **art**.

This chapter was adapted from Level 1 of Ian Schreiber's *Game Design Concepts* course.

2. Costikyan's Definition

As Schreiber says, the definitions in the previous chapter don't provide enough detail to help us think about how we might design games. Several authors have tried to give us more detail about the components or building blocks of games so that we might be able to use them in our designs.

In 1994, Greg Costikyan wrote the influential article "I Have No Words and I Must Design" for *Interactive Fantasy* #2, a British journal about role-playing games. I have summarized Costikyan's article before because it represents an early attempt to define a game and because it is relatively easy to understand.

Costikyan says an activity must have six elements in order to be considered a game. If it is missing any of the six elements, it is something other than a game, perhaps some other kind of play, but not a game. His six elements are: tokens, goal(s), opposition, decision-making, information and managing resources.

A game must have **game tokens**. He means that there must be something within the game that represents the player and the player's status within the game. In *Monopoly*, for example, the player's piece (top hat, race car, horse, and so on) is a game token because it represents the player. But the cards with the various properties that the player owns (Broadway, Marvin Garden, Illinois Ave, and so on) are also game tokens because they also represent the player's status within the game. In addition, the fake money that a player has represent how wealthy or poor the player is and, therefore, are game tokens. In some games, like basketball, the player's body is one of the game tokens.

A game must have a **goal**, something the player is striving for. In *Monopoly*, for example, the goal is to be the last player with money or, in other words, to bankrupt all the other players. in *War*, the goal is to obtain all of the cards in the deck. This is an element that makes some activities that we normally consider to be games not

games in Costikyan's point of view. For example, *SimCity* and *The Sims* are not games according to Costikyan because they don't have goals that are set by the game. The player can create a goal to strive for but the game doesn't impose that on the player.

A game must have **opposition**, something that gets in the way of the player reaching their goal. In a later version of his article, Costikyan renames this as **struggle**. That is, the player must have an obstacle they must struggle to overcome in order to achieve their goal. This is a simple, yet profound, statement. By entering into the realm of the game, the player agrees to try to reach the goal of the game in a kind of circuitous manner. The participant in the game of *War* will not just grab all of the cards in the deck but will instead abide by the rules of the game and attempt to overcome the obstacles that the rules place in their way. The opposition in a game typically comes from the rules of the game as well as any opponents who are trying to achieve the same goal. In *Monopoly*, for example, the opposing players are part of the opposition, one of the obstacles that gets in the way of a player reaching their goal. But competition is only one way to add opposition to a game. The rule that player can only move their piece when they roll the dice on their turn is also part of the opposition as is the rule that the player can only buy a property that they land on and that is not already owned by another player. In fact, all of the rules of game are part of the opposition. The card game *Solitaire* is a good example of a game that has no opposing players, no competition, but still has opposition in the form of rules that force the player to struggle to achieve their goal.

A game must also have **decision-making**. This is perhaps the most important characteristic of a game. A player must be presented with a series of choices, each of which impacts on their chances of reaching the goal before their opponent. In fact, Costikyan would not consider the card game *War* a game because there is no decision-making. In *War*, a player simply flips an unknown card at random from his/her deck and hopes for the best. Decision-making allows the player to control their destiny (to an extent). Through

decision-making, the player expresses a personality, a strategy for how to win the game.

In order to make good decisions, a player must be presented with some **information** on which to base those decisions. To understand this concept, think about the game of *War* (which, again, Costikyan would not consider a game). The player in this game is not presented with any decision-making opportunities. The player simply flips a card and hopes for the best. Many students, when asked to add decision-making to the game, suggest that the player's deck of cards be split into two decks and the player must decide the deck from which to flip a card. If the cards are all faced down, no actual decision has been added to the game because the player is given no information about the contents of each deck. The information about what card is at the top of each deck exists but it is hidden from the player. In fact, all of the information in the game is hidden from the player. The player must make a random choice about which deck to pull a card from. This random choice does not represent a *meaningful* choice. The player is just guessing. So in order to have a meaningful decision to make, the player must be presented with at least SOME information. In *Chess*, the player is presented with *perfect information*, that is, no information is hidden from the player. In a game like *Texas Hold 'Em*, on the other hand, the player is presented with *imperfect information.* That is, some of the information is known to the player while some is unknown. This is also sometimes called *mixed information.*

Finally, the player must be given the opportunity to **manage resources**. A resource is something the player uses in order to achieve the goal of the game. For example, in *Monopoly*, one of the player's resources is the space they land on. If the space has not already been purchased, the player can use the information they have about who owns what, the price of each property, whether they will make a monopoly by purchasing the property, and so on, to determine whether to purchase this property or not. The relationship between decision-making, information and the

management of resources is an intimate one, one that is difficult to pull apart.

This chapter was adapted from my June 2, 2010 blog post "The Post in Which I Get Philosophical."

3. Formal Elements of Games

What are the most basic building blocks of games?

This depends on who you ask. There are many schemes of classification. Like the definition of "game," none is perfect, but by looking at all of them we can see some emerging themes that can shed light on the kinds of things that we need to create as game designers if we are to make games.

We refer to these basic building blocks as "formal elements," not because they have anything to do with wearing a suit and tie, but because they are "formal" in the mathematical and scientific sense: something that can be explicitly defined. Some designers refer to them as "atoms" – in the sense that these are the smallest parts of a game that can be isolated and studied individually.

What follows are some parts of games, and some of the things designers may consider when looking at these formal elements.

Players

How many players does the game support? Must it be an exact number (4 players only), or a variable number (2 to 5 players)? Can players enter or leave during play? How does this affect play?

What is the relationship between players: are there teams, or individuals? Can teams be uneven? Here are some example player structures; this is by no means a complete list:

- Solitaire (1 player vs. the game system). Examples include the card game *Klondike* (sometimes just called "Solitaire") and the video game *Minesweeper*.
- Head-to-head (1 player vs. 1 player). *Chess* and *Go* are classic examples.
- "PvE" (multiple players vs. the game system). This is common in MMOs like *World of Warcraft*. Some purely-cooperative board games exist too, such as *Knizia's Lord of the Rings*, *Arkham Horror*, and *Pandemic*.

- One-against-many (1 player vs. multiple players). The board game *Scotland Yard* is a great example of this; it pits a single player as Mr. X against a team of detectives.
- Free-for-all (1 player vs. 1 player vs. 1 player vs. …). Perhaps the most common player structure for multi-player games, this can be found everywhere, from board games like *Monopoly* to "multiplayer deathmatch" play in most first-person shooter video games.
- Separate individuals against the system (1 player vs. a series of other players). The casino game *Blackjack* is an example, where the "House" is playing as a single player against several other players, but those other players are not affecting each other much and do not really help or hinder or play against each other.
- Team competition (multiple players vs. multiple players [vs. multiple players…]). This is also a common structure, finding its way into most team sports, card games like *Bridge* and *Spades*, team-based online games like "Capture the Flag" modes from first-person shooters, and numerous other games.
- Predator-Prey. Players form a (real or virtual) circle. Everyone's goal is to attack the player on their left, and defend themselves from the player on their right. The college game *Assassination* and the trading-card game *Vampire: the Eternal Struggle* both use this structure.
- Five-pointed Star. I first saw this in a five-player *Magic: the Gathering* variant. The goal is to eliminate both of the players who are *not* on either side of you.

Objectives (goals)

What is the object of the game? What are the players trying to do? This is often one of the first things you can ask yourself when designing a game, if you're stuck and don't know where to begin. Once you know the objective, many of the other formal elements

will seem to define themselves for you. Some common objectives (again, this is not a complete list):

- Capture/destroy. Eliminate all of your opponent's pieces from the game. *Chess* and *Stratego* are some well-known examples where you must eliminate the opposing forces to win.
- Territorial control. The focus is not necessarily on destroying the opponent, but on controlling certain areas of the board. *RISK* and *Diplomacy* are examples.
- Collection. The card game *Rummy* and its variants involve collecting sets of cards to win. *Bohnanza* involves collecting sets of beans. Many platformer video games (such as the *Spyro* series) included levels where you had to collect a certain number of objects scattered throughout the level.
- Solve. The board game *Clue* (or *Cluedo*, depending on where you live) is an example of a game where the objective is to solve a puzzle. Lesser-known (but more interesting) examples are *Castle of Magic* and *Sleuth*.
- Chase/race/escape. Generally, anything where you are running towards or away from something; the playground game *Tag* and the video game *Super Mario Bros.* are examples.
- Spatial alignment. A number of games involve positioning of elements as an objective, including the non-digital games *Tic-Tac-Toe* and *Pente* and the video game *Tetris*.
- Build. The opposite of "destroy" – your goal is to advance your character(s) or build your resources to a certain point. *The Sims* has strong elements of this; the board game *Settlers of Catan* is an example also.
- Negation of another goal. Some games end when one player performs an act that is forbidden by the rules, and that player loses. Examples are the physical dexterity games *Twister* and *Jenga*.

Rules (mechanics)

There are three categories of rules: *setup* (things you do once

at the beginning of the game), *progression of play* (what happens during the game), and *resolution* (what conditions cause the game to end, and how is an outcome determined based on the game state).

Some rules are automatic: they are triggered at a certain point in the game without player choices or interaction ("Draw a card at the start of your turn" or "The bonus timer decreases by 100 points every second"). Other rules define the choices or actions that the players can take in the game, and the effects of those actions on the game state.

Let's dig deeper. Salen & Zimmerman's *Rules of Play* classifies three types of rules, which they call operational, constituative, and implied (these are not standard terms in the industry, so the concepts are more important than the terminology in this case). To illustrate, let's consider the rules of *Tic-Tac-Toe*:

- Players: 2
- Setup: Draw a 3×3 grid. Choose a player to go first as X. Their opponent is designated O.
- Progression of play: On your turn, mark an empty square with your symbol. Play then passes to your opponent.
- Resolution: If you get 3 of your symbol in a row (orthogonally or diagonally), you win. If the board is filled and there is no winner, it is a draw.

These are what *Rules of Play* calls the "operational" rules. Think for a moment: are these the only rules of the game?

At first glance, it seems so. But what if I'm losing and simply refuse to take another turn? The rules do not explicitly give a time limit, so I could "stall" indefinitely to avoid losing and still be operating within the "rules" as they are typically stated. However, in actual play, a reasonable time limit is implied. This is not part of the formal (operational) rules of the game, but it is still part of what *Rules of Play* calls the "implied" rules. The point here is that there is some kind of unwritten social contract that players make when playing a game, and these are understood even when not stated.

Even within the formal rules there are two layers. The 3×3 board and "X" and "O" symbols are specific to what Costikyan calls the *color* of this game, but you could strip them away. By reframing the squares as the numbers 1 through 9 and turning spatial alignment into a mathematical property, you can get *Three-to-Fifteen*. While *Tic-Tac-Toe* and *Three-to-Fifteen* have different implementations and appearances, the underlying abstract rules are the same. We do not normally think in these abstract terms when we think of "rules" but they are still there, under the surface. *Rules of Play* calls these "constituative" rules.

Is it useful to make the distinction between these three types of rules? I think it is important to be aware of them for two reasons:

- The distinction between "operational" and "constituative" rules helps us understand why one game is fun in relation to other games. The classic arcade game *Gauntlet* has highly similar gameplay to the first-person shooter *DOOM*; the largest difference is the position of the camera. For those of you who play modern board games, a similar statement is that *Puerto Rico* is highly similar to *Race for the Galaxy*. The similarity may not be immediately apparent because the games look so different on the surface, unless you are thinking in terms of game states and rules.
- Many first-person shooters contain a rule where, when a player is killed, they re-appear ("respawn") in a specific known location. Another player can stand near that location and kill anyone that respawns before they have a chance to react. This is known as "spawn-camping" and can be rather annoying to someone on the receiving end of it. Is spawn-camping part of the game (since it is allowed by the rules)? Is it good strategy, or is it cheating? This depends on who you ask, as it is part of the "implied" rules of the game. When two players are operating under different implied rules, you will eventually get one player accusing the other of cheating (or just "being cheap") while the other player will get defensive and say that

they're playing by the rules, and there's no reason for them to handicap themselves when they are playing to win. The lesson here is that it is important for the game designer to define as many of these rules as possible, to avoid rules arguments during play.

Resources and resource management

"Resources" is a broad category, and we use it in this text to mean everything that is under control of a single player. Obviously this includes explicit resources (wood and wheat in *Settlers of Catan*, health and mana and currency in *World of Warcraft*), but this can also include other things under player control:

- Territory in *RISK*
- Number of questions remaining in *Twenty Questions*
- Objects that can be picked up in video games (weapons, powerups)
- Time (either game time, or real time, or both)
- Known information (as the suspects that you have eliminated in *Clue*)

What kinds of resources do the players control? How are these resources manipulated during play? This is something the game designer must define explicitly.

Game State

Some "resource-like" things are not owned by a single player, but are still part of the game: unowned properties in *Monopoly*, the common cards in *Texas Hold 'Em*. Everything in the game together, including the current player resources and everything else that makes up a snapshot of the game at a single point in time is called the *game state*.

In board games, explicitly defining the game state is not always necessary, but it is sometimes useful to think about. After all, one way to think about rules is that they are the means by which the game is transformed from one game state to another.

In video games, someone must define the game state, because it includes all of the data that the computer must keep track of. Normally this task falls to a programmer, but if the game designer can explicitly define the entire game state it can greatly aid in the understanding of the game by the programming team.

Information

How much of the game state is visible to each player? Changing the amount of information available to players has a drastic effect on the game, even if all other formal elements are the same. Some examples of information structures in games:

- A few games offer total information, where all players see the complete game state at all times. *Chess* and *Go* are classic board game examples. These games are said to contain *perfect information*.
- Games can include some information that is private to each individual. Think of *Poker* and other card games where each player has a hand of cards that only they can see.
- One player can have their own privileged information, while other players do not. This is common in one-against-many player structures, like *Scotland Yard*.
- The game itself can contain information that is hidden from all players. Games like *Clue* and *Sleuth* actually have the victory condition that a player discover this hidden information.
- These can be combined. Many "real-time strategy" computer games use what is called "fog of war" where certain sections of the map are concealed to any player that does not have a unit in sight range. Some information is therefore hidden from all players. Beyond that, players cannot see each other's screens, so each player is unaware of what information is and isn't available to their opponents.

Sequencing

In what order do players take their actions? How does play flow

from one action to another? Games can work differently depending on the turn structure that is used:

- Some games are purely turn-based: at any given time it is a single player's "turn" on which they may take action. When they are done, it becomes someone else's turn. Most classic board games and turn-based strategy games work this way.
- Other games are turn-based, but with simultaneous play (everyone takes their turn at the same time, often by writing down their actions or playing an action card face-down and then simultaneously revealing). The board game *Diplomacy* works like this. There is also an interesting *Chess* variant where players write down their turns simultaneously and then resolve (two pieces entering the same square on the same turn are both captured) that adds tension to the game.
- Still other games are real-time, where actions are taken as fast as players can take them. Most action-oriented video games fall into this category, but even some non-digital games (such as the card games *Spit* or *Speed*) work this way.
- There are additional variations. For a turn-based game, what order do players take their turns? Taking turns in clockwise order is common. Taking turns in clockwise order and then skipping the first player (to reduce the first-player advantage) is a modification found in many modern board games. I've also seen games where turn order is randomized for each round of turns, or where players pay other resources in the game for the privilege of going first (or last), or where turn order is determined by player standing (player who is currently winning goes first or last).
- Turn-based games can be further modified by the addition of an explicit time limit, or other form of time pressure.

Player Interaction

This is an often-neglected but highly important aspect of games

to consider. How do players interact with one another? How can they influence one another? Here are some examples of player interactions

- Direct conflict ("I attack you")
- Negotiation ("If you support me to enter the Black Sea, I'll help you get into Cairo next turn")
- Trading ("I'll give you a wood in exchange for your wheat")
- Information sharing ("I looked at that tile last turn and I'm telling you, if you enter it a trap will go off")

Theme (or narrative, backstory, or setting)

These terms do have distinct meanings for people who are professional story writers, but for our purposes they are used interchangeably to mean the parts of the game that do not directly affect gameplay at all.

If it doesn't matter in terms of gameplay, why bother with this at all? There are two main reasons. First, the setting provides what Costikyan calls "color" in the game which helps players connect emotionally to the game. Most people find it hard to really care about the pawns on a chessboard the way they care about their *Dungeons & Dragons* character. And while this doesn't necessarily make one game "better" than another, it does make it easier for a player to become emotionally invested in the game.

The other reason is that a well-chosen theme can make a game easier to learn and easier to play, because the rules make sense. The piece movement rules in *Chess* have no relation to the theme and must therefore be memorized by someone learning the game. By contrast, the roles in the board game *Puerto Rico* have some relation to their game function: the builder lets you build buildings, the mayor recruits new colonists, the captain ships goods off to the Old World, and so on. It is easy to remember what most actions do in the game, because they have some relation to the theme of the game.

This chapter was adapted from Level 3 of Ian Schreiber's *Game Design Concepts* course.

4. Weaknesses of Definitions

Which of the earlier definitions is correct?

None of them are perfect. If you try to come up with your own definition, it will likely be imperfect as well. Here are a few common edge cases that commonly cause problems with definitions:

- **Puzzles**, such as crossword puzzles, Sudoku, Rubik's Cube, or logic puzzles. Are these games? It depends on the definition. Salen & Zimmerman say they are a *subset* of games where there is a set of correct answers. Costikyan says they are not games, although they may be contained within a game.
- **Role-playing games**, such as *Dungeons & Dragons*. They have the word "game" right in the title, yet they are often not considered games (for example, because they often have no final outcome or resolution, no winning or losing).
- **Choose-your-own-adventure books**. These are not generally thought of as games; you say you are "reading" a book, not "playing" it. And yet, it fits most of the criteria for most definitions of a game. To make things even more confusing, if you take one of these books, add a tear-out "character sheet" with some numeric stats, include "skill checks" on some pages where you roll a die against a stat, and call it an "adventure module" instead of a "choose-your-own-adventure book," we would now call it a game!
- **Stories**. Are games stories? On the one hand, most stories are linear, while games tend to be more dynamic. On the other hand, most games have some kind of story or narrative in them; we even have professional story writers that work on multi-million-dollar video game projects. And even beyond that, a player can tell a story about *their* game experience ("let me tell you about this Chess game I played last night, it was awesome"). For now, keep in mind that the concepts

of **story** and **game** are related in many ways, and we'll explore this more thoroughly later in the course.

This chapter was adapted from Level 1 of Ian Schreiber's *Game Design Concepts* course.

PART II

GAME ANALYSIS FOR GAME DESIGNERS

5. Critical Analysis of Games

What is a critical analysis, and why do we care?

Critical analysis is not just a game review. We are not concerned with how many out of five stars, or any numbers from 0 to 10, or whether or not a game is "fun" (whatever *that* means), or aiding in the consumer decision of whether or not to buy a game.

Critical analysis does not just mean a list of things that are wrong with the game. The word "critical" in this context does not mean "fault-finding" but rather a thorough and unbiased look at the game.

Critical analysis is useful when discussing or comparing games. You can say "I like the card game *Bang!* because it's fun" but that does not help us as designers to learn *why* it is fun. We must look at the parts of games and how they interact in order to understand how each part relates to the player experience. The more we analyze the games we play, the more we understand what makes a game *good*.

Critical analysis is also useful when examining our own works in progress. For a game that you're working on, how do you know what to add or remove to make it better?

There are many ways to critically analyze a game, but here is a three-step process that we can begin with:

1. Describe the game's formal elements. Do not interpret at this point, simply state what is there.
2. Describe the results of the formal elements when put in motion. How do the different elements interact? What is the play of the game like? Is it effective?
3. Try to understand why the designer chose those elements and not others. Why this particular player structure, and why that set of resources? What would have happened if the designer had chosen differently?

Some specific Costikyan-related questions to ask yourself during a critical analysis of a game:

- Describe the game by answering the following questions
    - What is the name of the game?
    - Who is the audience of the game?
    - What materials are needed to play the game?
    - Is the game turn-based? If yes, what does a typical turn look like? If not, how do players begin to play the game?
- What is the overall goal of the game? That is, what are the players trying to accomplish?
- Where does the opposition in the game come from? That is, what conspires to prevent a player from reaching her goals?
- What items constitute the game tokens in the game? Explain why each of these things is a token.
- What are the decision-making opportunities that the player has? Be complete in listing these decisions.
    - Which decisions allow a player to manage their resources? In answering this question, list the resources and then explain how the decision relates to the *management* of a particular resource.
- What kind of information can players use to make decisions in the game?
    - For each item of information that you list, who has access to it? Is that information available to all players, available to a subset of players or hidden from all players?

As you read through the other chapters in this section, you will learn about more ways to think about games. Asking yourself questions about these various frameworks will help you to better understand what you might change about your game in order to make it better.

This chapter was adapted from Level 3 of Ian Schreiber's *Game Design Concepts* course with original material added by Cathie LeBlanc

6. Games as Systems

After reading the chapter about formal elements, you should note two things.

First, if you change even one formal element, it can make for a *very* different game. Each formal element of a game contributes in a deep way to the player experience. When designing a game, give thought to each of these elements, and make sure that each is a deliberate choice.

Second, note that these elements are interrelated, and changing one can affect others. Rules govern changes in Game State. Information can sometimes become a Resource. Sequencing can lead to different kinds of Player Interaction. Changing the number of Players can affect what kinds of Objectives can be defined. And so on.

Because of the interrelated nature of these parts, you can frame any game as a **system**. (One dictionary definition of the word "system" is: a combination of things or parts that form a complex whole.)

In fact, a single game can contain several systems. *World of Warcraft* has a combat system, a quest system, a guild system, a chat system, and so on…

Another property of systems is that it is hard to fully understand or predict them just by defining them; you gain a far deeper understanding by seeing the system in action. Consider the physical system of projectile motion. There is a mathematical equation to define the path of a ball being thrown, and you could even predict its behavior… but the whole thing makes a lot more sense if you see someone actually throwing a ball.

Games are like this, too. You can read the rules and define all the formal elements of a game, but to truly understand a game you need to play it, usually more than once. You need to see how the various pieces of the game system interact with each other given different

game states. Only then can you understand how the game system might be improved.

This chapter was adapted from Level 3 of Ian Schreiber's *Game Design Concepts* course.

7. Quality of Decision-Making

As Costikyan pointed out in *I Have No Words*, we often use the buzzword "interactivity" when describing games when we actually mean "decision-making." Decisions are, in essence, what players *do* in a game. Remove all decisions and you have a movie or some other linear activity, not a game. As pointed out in *Challenges*, there are two important exceptions, games which have no decisions at all: some children's games and some gambling games. For gambling games, it makes sense that a lack of decisions is tolerable. The "fun" of the game comes from the thrill of possibly winning or losing large sums of money; remove that aspect and most gambling games that lack decisions suddenly lose their charm. At home when playing only for chips, you're going to play games like *Blackjack* or *Poker* that have real decisions in them; you are probably not going to play *Craps* or a slot machine without money being involved.

You might wonder, what is it about children's games that allow them to be completely devoid of decisions? We'll get to that in a bit.

Other than those two exceptions, most games have some manner of decision-making, and it is here that a game can be made more or less interesting. Sid Meier has been quoted as saying that a good game is a series of interesting decisions (or something like that), and there is some truth there. But what makes a decision "interesting"? *Battleship* is a game that has plenty of decisions but is not particularly interesting for most adults; why not? What makes the decisions in *Settlers of Catan* more interesting than *Monopoly*? Most importantly, how can you design your own games to have decisions that are actually compelling?

Things Not To Do

Before describing good kinds of decisions, it is worth explaining some common kinds of **uninteresting** decisions commonly found in

games. Note that the terminology here (obvious, meaningless, blind) is my own, and is not "official" game industry jargon. At least not yet.

- **Meaningless decisions** are perhaps the worst kind: there is a choice to be made, but it has no effect on gameplay. If you can play either of two cards but both cards are identical, that's not really much of a choice.
- **Obvious decisions** at least have an effect on the game, but there is clearly one right answer, so it's not really much of a choice. Most of the time, the number of dice to roll in the board game *RISK* falls into this category; if you are attacking with 3 or more armies, you have a "decision" of whether to roll 1, 2, or 3 dice… but your odds are better rolling all 3, so it's not much of a decision except in very special cases. A more subtle example would be a game like *Trivial Pursuit*. Each turn you are given a trivia question, and if you know the correct answer it could be said that you have a decision: say the right answer, or not. Except that there's never any reason to *not* say the right answer if you know it. The fun of the game comes from showing off your mastery of trivia, not from making any brilliant strategic maneuvers. This is also, I think, why quiz shows like *Jeopardy!* are more fun to watch than to play.
- **Blind decisions** have an effect on the game, and the answer is not obvious, but there is now an additional problem: the players do not have sufficient knowledge on which to make the decision, so it is essentially random. Playing *Rock-Paper-Scissors* against a truly random opponent falls into this category; your choice affects the outcome of the game, but you have no way of knowing what to choose.

These kinds of decisions are, by and large, not much fun. They are not particularly interesting. All three represent a waste of a player's time. Meaningless decisions could be eliminated, obvious decisions could be automated, and blind decisions could be randomized without affecting the outcome of the game at all.

In this context, it is suddenly easy to see why so many games are not particularly compelling.

What Makes Good Decisions?

Now that we know what makes *weak* decisions, the easiest answer is "don't do that!" But we can take it a little further. Generally, *interesting* decisions involve some kind of **tradeoff**. That is, you are giving up one thing in exchange for another. These can take many different forms. Here are a few examples (again I use my own invented terminology here):

- **Resource trades**. You give one thing up in exchange for another, where both are valuable. Which is *more* valuable? This is a value judgment, and the player's ability to correctly judge or anticipate value is what determines the game's outcome.
- **Risk versus reward**. One choice is safe. The other choice has a potentially greater payoff, but also a higher risk of failure. Whether you choose safe or dangerous depends partly on how desperate a position you're in, and partly on your analysis of just *how* safe or dangerous it is. The outcome is determined by your choice, plus a little luck… but over a sufficient number of choices, the luck can even out and the more skillful player will generally win. (Corollary: if you want more luck in your game, reduce the total number of decisions.)
- **Choice of actions**. You have several potential things you can do, but you can't do them all. The player must choose the actions that they feel are the most important at the time.
- **Short term versus long term**. You can have something right now, or something better later on. The player must balance immediate needs against long-term goals.
- **Social information**. In games where bluffing, deal-making and backstabbing are allowed, players must choose between playing honestly or dishonestly. Dishonesty may let you come out better on the current deal, but may make other players less likely to deal with you in the future. In the right (or wrong) game, backstabbing your opponents may have very negative

real-world consequences.

- **Dilemmas**. You must give up one of several things. Which one can you most afford to lose?

Notice the common thread here. All of these decisions involve the player judging the value of something, where values are shifting, not always certain, and not obvious.

The next time you play a game that you really like, think about what kinds of decisions you are making. If you have a particular game that you strongly *dis*like, think about the decisions being made there, too. You may find something about yourself, in terms of the kinds of decisions that you enjoy making.

Emotional Decisions

There is one class of decisions that is useful to consider: decisions that have an emotional impact on the player. The decision of whether to save your buddy (while using some of your precious supplies) or leave him behind to die (potentially denying yourself some AI-assisted help later on) in *Far Cry* is a resource decision, but it is also meant to be an emotional one – and certainly, an identical decision made on a real-life battlefield would come down to more than just an analysis of available resources and probabilities. Likewise, the majority of players do not play through a game with moral choices (such as *Knights of the Old Republic* or *Fable*) as pure evil – not because "evil" is a suboptimal strategy, but because even in a fictional simulated world, a lot of people can't stomach the thought of torturing and killing innocent bystanders.

Or consider a common decision made at the start of many board games: what color are you? Color is usually just a way to uniquely identify player tokens on the board, and has no effect on gameplay. However, many people have a favorite color that they always play, and can become quite emotionally attached to "their" color. It can be rather entertaining when two players who "always" play Green, play together for the first time and start arguing over who gets to be Green. If player color has no effect on gameplay, it is a meaningless decision. It *should* therefore be uninteresting, and yet some players

paradoxically find it quite meaningful. The reason is that they are emotionally invested in the outcome. This is not to say that you can cover up a bad game by artificially adding emotions; but rather, as a designer, be aware of what decisions your players seem to respond to on an emotional level.

This chapter was adapted from Level 7 of Ian Schreiber's *Game Design Concepts* course.

8. Mechanics, Dynamics, and Aesthetics

In this chapter, we will examine what makes a *good* rule as opposed to a *bad* one. We will also examine the different *kinds* of rules that form a game designer's palette and that we can look for in analyzing games. Finally, we will examine the relationship between the game rules and the player experience.

One of the few academic papers that achieved wide exposure within the game industry is *MDA Framework* by LeBlanc (no relationship to me!), Hunicke and Zabek. It probably helps that the authors are experienced game designers. There are two parts of this paper that made it really influential. The first is the Mechanics/Dynamics/Aesthetics (MDA) conceptualization, which offers a way to think about the relationship of rules to player experience, and also the relationship between player and designer. The second part is the "8 kinds of fun" which we will discuss in a later chapter.

LeBlanc *et al.* define a game in terms of its Mechanics, Dynamics, and Aesthetics:

- Mechanics are a synonym for the "rules" of the game. These are the constraints under which the game operates. How is the game set up? What actions can players take, and what effects do those actions have on the game state? When does the game end, and how is a resolution determined? These are defined by the mechanics.
- Dynamics describe the *play* of the game when the rules are set in motion. What strategies emerge from the rules? How do players interact with one another?
- Aesthetics (in the MDA sense) do not refer to the visual elements of the game, but rather the *player experience* of the game: the effect that the dynamics have on the players

themselves. Is the game "fun"? Is play frustrating, or boring, or interesting? Is the play emotionally or intellectually engaging?

Before the *MDA Framework* was written, the terms "mechanics" and "dynamics" were already in common use among designers. The term "aesthetics" in this sense had not, but has gained more use in recent years.

The Process of Design

With the definitions out of the way, why is this important? This is one of the key points of the MDA paper. The game designer only creates the Mechanics directly. The Dynamics emerge from the Mechanics, and the Aesthetics arise out of the Dynamics. The game designer may *want* to design the player experience, or at least that may be the ultimate goal the designer has in mind… but as designers, we are stuck building the rules of the game and hoping that the desired experience emerges from our rules.

This is why game design is sometimes referred to as a **second-order design problem**: because we do not define the solution, we define something that creates something else that creates the solution. **This is why game design is hard**. Or at least, it is one reason. Design is not just a matter of coming up with a "Great Idea" for a game; it is about coming up with a set of rules that will implement that idea, when two-thirds of the final product (the Dynamics and Aesthetics) are not under our direct control.

The Process of Play

Designers start with the Mechanics and follow them as they grow outward into the Aesthetics. You can think of a game as a sphere, with the Mechanics at the core, the Dynamics surrounding them, and the Aesthetics on the surface, each layer growing out of the one inside it. One thing the authors of MDA point out is that this is *not* how games are experienced from the player's point of view.

A player sees the surface first – the Aesthetics. They may be *aware* of the Mechanics and Dynamics, but the thing that really makes an immediate impression and that is most easily understood is the Aesthetics. This is why, even with absolutely no knowledge

or training in game design, *anyone* can play a game and tell you whether or not they are having a good time. They may not be able to articulate *why* they are having a good time or *what* makes the game "good" or "bad"... but anyone can tell you right away how a game makes them feel.

If a player spends enough time with a game, they may learn to appreciate the Dynamics of the game and now their experience arises from them. They may realize that they do or don't like a game because of the specific kinds of interactions they are having with the game and/or the other players. And if a player spends even more time with that game, they may eventually have a strong enough grasp of the Mechanics to see how the Dynamics are emerging from them.

If a game is a sphere that is designed from the inside out, it is *played* from the outside in. This is one of the key points of MDA. The designer creates the Mechanics and everything flows outward from that. The player experiences the Aesthetics and then their experience flows inward. As designers, we must be aware of **both** of these ways of interacting with a game. Otherwise, we are liable to create games that are fun for designers but not players.

One Example of MDA in action

In a First-Person Shooter video game, a common mechanic is for players to have "spawn points" – dedicated places on the map where they re-appear after getting killed. Spawn points are a **mechanic**. This leads to the **dynamic** where a player may sit next to a spawn point and immediately kill anyone as soon as they respawn. And lastly, the **aesthetics** would likely be frustration at the prospect of coming back into play only to be killed again immediately.

Suppose you are designing a new FPS and you notice this frustration aesthetic in your game, and you want to fix this so that the game is not as frustrating. You cannot simply change the aesthetics of the game to "make it more fun" – this may be your goal, but it is not something under your direct control. You cannot even change the dynamics of spawn camping directly; you cannot tell the players how to interact with your game, except through

the mechanics. So instead, you must change the mechanics of the game – maybe you try making players respawn in random locations rather than designated areas – and then you hope that the desired aesthetics emerge from your mechanics change.

How do you know if your change worked? Playtest, of course!

How do you know *what* change to make, if the effects of mechanics changes are so unpredictable? We will get into some basic tips and tricks later. For now, the most obvious way is designer intuition. The more you practice, the more you design games, the more you make rules changes and then playtest and see the effects of your changes, the better you will get at making the right changes when you notice problems… and occasionally, even creating the right mechanics in the first place. There are few substitutes for experience… which, incidentally, is why so much of this course involves getting you off your butt and making games :).

Mechanics, Dynamics and Complexity

Generally, adding additional mechanics, new systems, additional game objects, and new ways for objects to interact with one another (or for players to interact with the game) will lead to a greater complexity in the dynamics of the game. For example, compare *Chess* and *Checkers*. *Chess* has six kinds of pieces (instead of two) and a greater number of actions that each piece can take, so it ends up having more strategic depth.

Is more complexity good, or bad? It depends. *Tetris* is a very simple but still very successful game. Some games are so simple that they are not fun beyond a certain age, like *Tic-Tac-Toe*. Other games are too complex for their own good, and would be better if their systems were a bit more simplified and streamlined.

Do more complex mechanics *always* lead to more complex dynamics? No – there are some cases where very simple mechanics create extreme complexity (as is the case with *Chess*). And there are other cases where the mechanics are extremely complicated, but the dynamics are simple (imagine a modified version of the children's card game *War* that did not just involve comparison of numbers, but lookups on complex "combat resolution" charts). The

best way to gauge complexity, as you may have guessed, is to play the game.

Feedback Loops

One kind of dynamic that is often seen in games and deserves special attention is known as the **feedback loop**. There are two types, **positive feedback loops** and **negative feedback loops**. These terms are borrowed from other fields such as control systems and biology, and they mean the same thing in games that they mean elsewhere.

A positive feedback loop can be thought of as a reinforcing relationship. Something happens that causes the same thing to happen again, which causes it to happen yet again, getting stronger in each iteration – like a snowball that starts out small at the top of the hill and gets larger and faster as it rolls and collects more snow.

As an example, there is a relatively obscure shooting game for the NES called *The Guardian Legend*. Once you beat the game, you got access to a special extra gameplay mode. In this mode, you got rewarded with power-ups at the end of each level based on your score: the higher your score, the more power-ups you got for the next level. This is a positive feedback loop: if you get a high score, it gives you more power-ups, which make it easier to get an even higher score in the next level, which gives you even more power-ups, and so on.

Note that in this case, the reverse is also true. Suppose you get a low score. Then you get fewer power-ups at the end of that level, which makes it harder for you to do well on the next level, which means you will probably get an even lower score, and so on until you are so far behind that it is nearly impossible for you to proceed at all.

The thing that is often confusing to people is that *both* of these scenarios are *positive* feedback loops. This seems counterintuitive; the second example seems very "negative," as the player is doing poorly and getting fewer rewards. It is "positive" in the sense that the effects get stronger in magnitude on each iteration.

There are three properties of positive feedback loops that game designers should be aware of:

1. They tend to destabilize the game, as one player gets further and further ahead (or behind).
2. They cause the game to end faster.
3. The put emphasis on the early game, since the effects of early-game decisions are magnified over time.

Feedback loops usually have two steps (as in my *The Guardian Legend* example) but they can have more. For example, some Real-Time Strategy games have a positive feedback loop with four steps: players explore the map, which gives them access to more resources, which let them buy better technology, which let them build better units, which let them explore more effectively (which gives them access to more resources... and the cycle repeats). As such, detecting a positive feedback loop is not always easy.

Here are some other examples of positive feedback loops that you might be familiar with:

- Most "4X" games, such as the *Civilization* and *Master of Orion* series, are usually built around positive feedback loops. As you grow your civilization, it lets you generate resources faster, which let you grow faster. By the time you begin conflict in earnest with your opponents, one player is usually so far ahead that it is not much of a contest, because the core positive feedback loop driving the game means that someone who got ahead of the curve early on is going to be *much* farther ahead in the late game.
- Board games that feature building up as their primary mechanic, such as *Settlers of Catan*. In these games, players use resources to improve their resource production, which gets them more resources.
- The physical sport *Rugby* has a minor positive feedback loop: when a team scores points, they start with the ball again,

which makes it slightly more likely that they will score again. The advantage is thus given to the team who just gained an advantage. This is in contrast to most sports, which give the ball to the opposing team after a successful score.

Negative feedback loops are, predictably, the opposite of positive feedback loops in just about every way. A negative feedback loop is a balancing relationship. When something happens in the game (such as one player gaining an advantage over the others), a negative feedback loop makes it harder for that same thing to happen again. If one player gets in the lead, a negative feedback loop makes it easier for the opponents to catch up (and harder for a winning player to extend their lead).

As an example, consider a "Kart-style" racing game like *Mario Kart*. In racing games, play is more interesting if the player is in the middle of a pack of cars rather than if they are way out in front or lagging way behind on their own (after all, there is more interaction if your opponents are close by). As a result, the *de facto* standard in that genre of play is to add a negative feedback loop: as the player gets ahead of the pack, the opponents start cheating, finding better power-ups and getting impossible bursts of speed to help them catch up. This makes it more difficult for the player to maintain or extend a lead. This particular feedback loop is sometimes referred to as "rubber-banding" because the cars behave as if they are connected by rubber bands, pulling the leaders and losers back to the center of the pack.

Likewise, the reverse is true. If the player falls behind, they will find better power-ups and the opponents will slow down to allow the player to catch up. This makes it more difficult for a player who is behind to fall further behind. Again, both of these are examples of *negative* feedback loops; "negative" refers to the fact that a dynamic becomes weaker with iteration, and has nothing to do with whether it has a positive or negative effect on the player's standing in the game.

Negative feedback loops also have three important properties:

1. They tend to stabilize the game, causing players to tend towards the center of the pack.
2. They cause the game to take longer.
3. They put emphasis on the late game, since early-game decisions are reduced in their impact over time.

Some examples of negative feedback loops:

- Most physical sports like *Football* and *Basketball*, where after your team scores, the ball is given to the opposing team and they are then given a chance to score. This makes it less likely that a single team will keep scoring over and over.
- The board game *Starfarers of Catan* has a negative feedback loop where every player with less than a certain number of victory points gets a free resource at the start of their turn. Early on, this affects all players and speeds up the early game. Later in the game, as some players get ahead and cross the victory point threshold, the players lagging behind continue to get bonus resources. This makes it easier for the trailing players to catch up.

Use of Feedback Loops

Are feedback loops good or bad? Should we strive to include them, or are they to be avoided? As with most aspects of game design, it depends on the situation. Sometimes, a designer will deliberately add mechanics that cause a feedback loop. Other times, a feedback loop is discovered during play and the designer must decide what (if anything) to do about it.

Positive feedback loops can be quite useful. They end the game quickly when a player starts to emerge as the winner, without having the end game be a long, drawn-out affair. On the other hand, positive feedback loops can be frustrating for players who are trying to catch up to the leader and start feeling like they no longer have a chance.

Negative feedback loops can also be useful, for example to

prevent a dominant early strategy and to keep players feeling like they always have a chance to win. On the other hand, they can also be frustrating, as players who do well early on can feel like they are being punished for succeeding, while also feeling like the players who lag behind are seemingly rewarded for doing poorly.

What makes a particular feedback loop "good" or "bad" from a player perspective? This is debatable, but it seems to be largely a matter of player perception of fairness. If it feels like the game is artificially intervening to help a player win when they don't deserve it, it can be perceived negatively by players. How do you know how players will perceive the game? Playtest, of course.

Eliminating Feedback Loops

Suppose you identify a feedback loop in your game and you want to remove it. How do you do this? There are two ways.

The first is to shut off the feedback loop itself. All feedback loops (positive and negative) have three components:

- A "sensor" that monitors the game state;
- A "comparator" that decides whether to take action based on the value monitored by the sensor;
- An "activator" that modifies the game state when the comparator decides to do so.

For example, in the earlier kart-racing negative feedback loop example, the "sensor" is how far ahead or behind the player is, relative to the rest of the pack; the "comparator" checks to see if the player is farther ahead or behind than a certain threshold value; and the "activator" causes the opposing cars to either speed up or slow down accordingly, if the player is too far ahead or behind. All of these may form a single mechanic ("If the player is more than 300 meters ahead of all opponents, multiply everyone else's speed by 150%"). In other cases there may be three or more separate mechanics that cause the feedback loop, and changing any one of them will modify the nature of the loop.

This chapter was adapted from Level 5 of Ian Schreiber's *Game Design Concepts* course.

9. What Makes a Game "Fun"?

Let's talk a little bit about this elusive concept of "fun." Games, we are told, are supposed to be fun. The role of a game designer is, in most cases, to take a game and make it fun. Notice that we usually enclose the word "fun" in quotation marks, on purpose. The reasoning is that "fun" is not a particularly useful word for game designers. We instinctively know what it means, sure, but the word tells us nothing about how to *create* fun. What is fun? Where does it come from? What makes games fun in the first place?

Interesting decisions seem like they might be fun. Is that all there is to it? Not entirely, because it doesn't say anything about *why* these kinds of decisions are fun. Or why *uninteresting* decisions are still fun for children. For this, we turn to Raph Koster and his book influential book *Theory of Fun*.

What a lot of Koster's *Theory of Fun* boils down to is this: **the fun of games comes from skill mastery**. This is a pretty radical statement, because it equates "fun" with "learning"... and we often think "learning" is about "school" and many of us think school is about as *not* fun as you can get. So it deserves a little explanation.

Theory of Fun draws heavily on the work of psychologist Mihaly Csikszentmihalyi (pronounced just like it's spelled, in case you're wondering), who studied what he called the mental state of "flow" (we sometimes call it being "in the flow" or "in the zone"). This is a state of extreme focus of attention, where you tune out everything except the task you're concentrating on, you become highly productive, and your brain gives you a shot of neurochemicals that is pleasurable – being in a flow state is literally a natural high.

Csikszentmihalyi identified three requirements for a flow state to exist:

- You must be performing a **challenging activity** that requires **skill**.

- The activity must provide **clear goals** and **feedback**.
- The outcome is **uncertain** but can be **influenced** by your actions. (Csikszentmihalyi calls this the "paradox of control": you are in control of your actions which gives you indirect control over the outcome, but you do not have *direct* control over the outcome.)

If you think about it, these requirements make sense. Why would your brain need to enter a flow state to begin with, blocking out all extraneous stimuli and hyper-focusing your attention on one activity? It would only do this if it needs to in order to succeed at the task. What conditions would there have to be for a flow state to make the difference between success and failure? See above – you'd need to be able to influence the activity through your skill towards a known goal.

Csikszentmihalyi also gave five effects of being in a flow state:

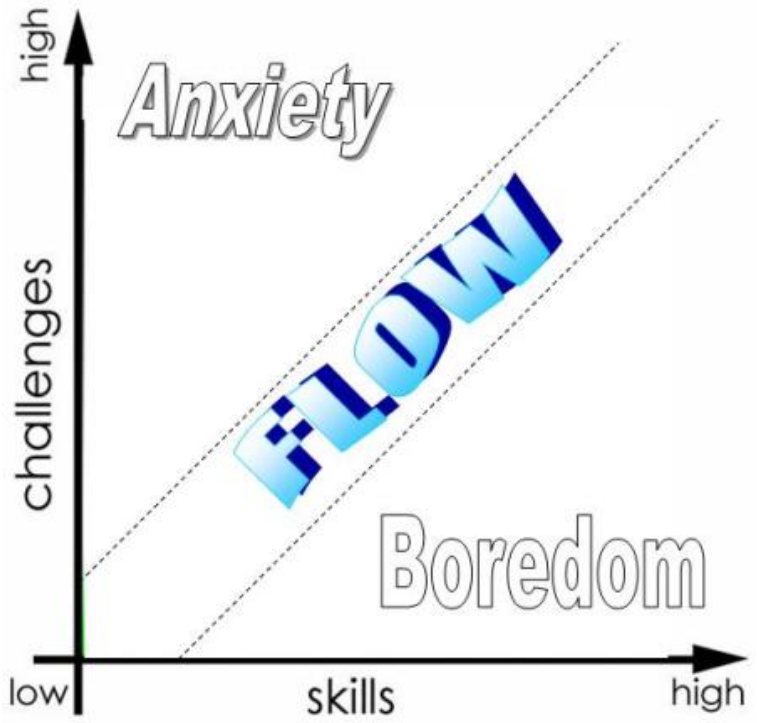
- A merging of action and awareness: spontaneous, automatic action/reaction. In other words, you go on autopilot, doing things without thinking about them. (In fact, your brain is moving faster than the speed of thought – think of a time when you played a game like *Tetris* and got into a flow state, and then at some point it occurred to you that you were doing really well, and then you wondered how you could keep up with the blocks falling so fast, and as soon as you started to think about it the blocks *were* moving too fast and you lost. Or maybe that's just me.)
- Concentration on immediate tasks: complete focus, without any mind-wandering. You are not thinking about long-term tradeoffs or other tasks; your mind is in the here-and-now, because it *has* to be.
- Loss of awareness of self, loss of ego. When you are in a flow state, you become one with your surroundings (in a Zen way, I suppose).
- There is a distorted sense of time. Strangely, this can go both

ways. In some cases, such as my *Tetris* example, time can seem to slow down and things seem to happen in slow motion. (Actually, what is happening is that your brain is acting so efficiently that it is working *faster*; everything else is still going at the same speed, but you are seeing things from your own point of reference.) Other times, time can seem to speed up; a common example is sitting down to play a game for "just five minutes"... and then six hours later, suddenly becoming aware that you burned away your whole evening.

- The experience of the activity is an end in itself; it is done for its own sake and not for an external reward. Again, this feeds into the whole "here-and-now" thing, as you are not in a mental state where you can think that far ahead.

Flow States in Games

Simplifying this a bit, we know that to be in a flow state, an activity must be **challenging**. If it is too easy, then the brain has no reason to waste extraneous mental cycles, as a positive outcome is already assured. If it is too difficult, the brain *still* has no reason to try hard, because it knows it's just going to fail anyway. The goal is to hit that sweet spot where the player can succeed... but only if they try hard. You'll often see a graph that looks like this, to demonstrate:



All this says is that if you have a high skill level and are given an easy task, you're bored; if you have a low skill level and are given a difficult task, you're frustrated; but if the challenge level of an activity is comparable to your current skill level… flow state! And this is good for games, because this is where a lot of the fun of games comes from.

Note that "flow" and "fun" are not synonyms, although they are related. You can be in a flow state without playing a game (and in fact without having fun). For example, an office worker might get into a flow state while filling out a series of forms. They may be operating at the edge of their ability in filling out the forms as efficiently as possible, but there may not be any real learning going on, and the process may not be fun, merely meditative.

One Slight Problem

When you are faced with a challenging task, you get better at it. It's fun because you are *learning*, remember? So, most people start out with an activity (like a game) with a low skill level, and if the game provides easy tasks, then so far so good. But what happens when the player gains some competency? If they keep getting the same easy tasks, the game becomes boring. This is essentially what happens in *Tic-Tac-Toe* when a child makes the transition to understanding the strategy of the game.

By the way, we can now answer our earlier question: why can children's games get away with a lack of meaningful decision-making? The answer is that young children are still learning valuable skills from these games: how to roll a die, move a token on a board, spin a spinner, take turns, read and follow rules, determine when the game ends and who wins, and so on. These skills are not instinctive and must be taught and learned through repeated play. When the child masters these skills, that is about the time when decision-less games stop holding any lasting appeal.

Ideally, as a game designer, you would like your game to have slightly more lasting playability than *Tic-Tac-Toe*. What can you do? Games offer a number of solutions. Among them:

- Increasing difficulty as the game progresses (we sometimes call this the "pacing" of a game). As the player gets better, they get access to more difficult levels or areas in a game. This is common with level-based video games.
- Difficulty levels or handicaps, where better players can choose to face more difficult challenges.
- Dynamic difficulty adjustment ("DDA"), a special kind of negative feedback loop where the game adjusts its difficulty during play based on the performance of the player.
- Human opponents as opposition. Sure, you can get better at the game… but if your opponent is also getting better, the game can still remain challenging if it has sufficient depth. (This can fail if the skill levels of different players fall out of synch with one another. I like to play games with my wife, and

we usually both start out at about the same skill level with any new game that really fascinates us both… but then sometimes, one of us will play the game a lot and become so much better than the other, that the game is effectively ruined for us. It is no longer a challenge.)

- Player-created expert challenges, such as new levels made by players using level-creation tools.
- Multiple layers of understanding (the whole "minute to learn, lifetime to master" thing that so many strategy games strive for). You can learn *Chess* in minutes, as there are only six different pieces… but then once you master that, you start to learn about which pieces are the most powerful and useful in different situations, and then you start to see the relationship between pieces, time, and area control, and then you can study book openings and famous games, and so on down the rabbit hole.
- Jenova Chen's *flOw* provides a novel solution to this: allow the player to change the difficulty level while playing based on their actions. Are you bored? Dive down a few levels and the action will pick up pretty fast. Are you overwhelmed? Run back to the earlier, easier levels (or the game will kick you back on its own if needed).

This chapter was adapted from Level 7 of Ian Schreiber's *Game Design Concepts* course.

10. Kinds of Fun

In the previous chapter, we discovered that "fun" is really just another word for "learning" and that putting players in a flow state is where this elusive "fun" comes from. We currently have an idea of *what* is fun, but it would help to know *why* these things are fun. What if there are new kinds of fun waiting to be discovered?

In their article *MDA Framework*, LeBlanc, et al. listed 8 kinds of fun. These are:

- **Sensation**. Games can engage the senses directly. Consider the audio and video "eye candy" of video games; the tactile feel of the wooden roads and houses in *Settlers of Catan*; or the physical movement involved in playing sports, *Dance Dance Revolution*, or any game on the Nintendo *Wii*.
- **Fantasy**. Games can provide a make-believe world (some might cynically call it "escapism") that is more interesting than the real world.
- **Narrative**. Games can involve stories, either of the embedded kind that designers put there, or the emergent kind that are created through player action.
- **Challenge**. Some games, particularly retro-arcade games, professional sports, and some highly competitive board games like *Chess* and *Go*, derive their fun largely from the thrill of competition. Even single-player games like *Minesweeper* or activities like mountain climbing are fun mainly from overcoming a difficult challenge.
- **Fellowship**. Many games have a highly social component to them. This alone is likely the reason that many American board games like *Monopoly* continue to sell many copies per year, in spite of the uninteresting decisions and dull mechanics. It is not the game, but the social interaction with family, that people remember fondly from their childhood.

- **Discovery**. This is rare in board games, but can be found in exploration-type games like *Tikal* and *Entdecker*. It is more commonly found in adventure and role-playing video games, particularly games in the *Zelda* and *Metroid* series.
- **Expression**. By this, the MDA authors mean the ability to express yourself through gameplay. Examples include games like *Charades* or *Poker* where the way that you act is at least as important as what other actions you take within a game; *Dungeons & Dragons* where the character you create is largely an expression of your own personal idea; or open-world and sim video games like *The Sims* or *Grand Theft Auto* or *Oblivion* or *Fable*, which are largely concerned with giving the player the tools needed to create their own custom experience.
- **Submission**. This kind of fun captures the idea of games as an ongoing hobby rather than an isolated event. Consider the metagame and the tournament scene in *Magic: the Gathering*, the demands of a guild to show up at regular meetings in *World of Warcraft*, or even the ritualized play of games at a weekly boardgame or tabletop-roleplaying group.

These are not all-or-nothing propositions. Games can contain several kinds of fun, in varying quantities.

Why not just create a game that has all eight kinds of fun? Wouldn't that be the holy grail of games, the game that's fun for everyone? Unfortunately, no. Just because these are different kinds of fun does not mean that *everyone* finds *all eight* of these things fun at all. Not only do different games provide different combinations and relative quantities of the various kinds of fun, but different players find different combinations more or less fun than others. For example, some people think that *Chess* is fun while many others do not; the "fun" Aesthetic arises not from the game alone, but the combination of game and player.

Are these eight the *only* kinds of fun? No; even the authors admit the above list is incomplete. There are many classification schemes

out there to identify different kinds of fun, including Nicole Lazzaro's four fun keys, or Pierre-Alexandre Garneau's fourteen forms of fun. Even the 8 kinds of fun from the MDA paper are debatable. Is it meaningful to separate Fantasy and Narrative, or are they just two ways of looking at the same kind of fun? Is submission really a kind of fun, or is it what happens when you have a game compelling enough to earn the status of "hobby" – is it a *cause* or an *effect*? What, exactly, counts as "expression" and what does not?

And where does the whole "fun is learning, learning is fun" thing from the last chapter come into this discussion?

Evolution (sans Pokemon)

In *Natural Funativity*, Noah Falstein answers the question by taking a trip back to early pre-history, when humans were at their hunter-gatherer stage. Primitive humans had to learn many skills in order to survive and reproduce. If we found it fun to learn certain skills, we would be more likely to practice them, and thus more likely to survive, reproduce, and pass on our genes to the next generation. Over time, those things that made us most likely to survive ended up being the things that we find "fun" today. Not all primitive hunter-gatherer skills are necessarily *useful* today, mind you, but our genetics haven't had time to catch up with our technology yet.

In short: **if a caveman found it useful, you'll find it fun.**

Falstein proposes three kinds of fun: "physical fun" (useful for any physical feats that allow us to fight or escape danger), "mental fun" (the problem-solving part of our brain that gave us such useful things as the wheel and fire), and "social fun" (the benefits of banding together in groups for mutual survival… and, of course, reproduction).

We can apply this evolutionary thought process to any "kinds of fun." Let's look at some of the MDA's 8 kinds of fun in this context:

- Sensation includes physical movement (good for building muscle) and looking at and hearing things that are interesting (good for detecting opportunities or dangers).

- Fantasy allows the kind of "what-if" scenario part of our brain to get stronger, allowing us to come up with novel ideas.
- Narrative is useful for passing on vital information and experience to others in your group, increasing the chance that *all* of you will survive.
- Challenge is a convenient way for different humans to show dominance over one another in a relatively safe way – "I can throw this rock further than you" is more useful than "let's fight to the death" if you're trying to build a colony.
- Fellowship opens up the possibility of new food sources (a single one of us might get killed hunting a large beast, but a group of us together can take it down). It's also rather hard to pass on your genetic material to the next generation if you're alone.
- Discovery is what makes us want to explore our nearby territory. The more territory we know, the more potential places for us to find food and shelter.
- Expression probably comes from the same part of us that is hardwired to communicate through language. Language, and communication in general, are pretty useful.
- Submission is in many cases a kind of social fun because we engage in these game hobbies with others. But some cases don't fall into this category. Maybe submission really *is* an effect of fun rather than the cause.

Discovering New Kinds of Fun

We can do this in reverse. Instead of taking something that's fun and tracing it back to the reptilian parts of our brain, we can isolate skills that our hunter-gatherer ancestors might have needed to survive, and then use that to figure out what we would find fun. For example, here are some activities that are often found in games:

- **Collection**. This is the "gathering" part of hunting-and-gathering, so you would expect it to be fun. And it is. Before video games became ubiquitous, the world's most popular

hobby was stamp collecting. In many board games you collect resources or tokens. Trading Card Game players collect cards. In the video game world, we've been collecting things since Mario first started collecting coins.

- **Spatial Reasoning**. Primitive humans needed to figure out spatial relationships in order to build useful tools (for example, if you want to find a big stick to make a crude ladder or bridge, you need to be able to estimate length; if you want to stick two pieces of wood together, you need to be able to figure out how to make them fit). Many games make use of spatial relationships, from *Tetris* to *Pente*.
- **Advancement**. This is kind of a meta-skill, the skill of learning new skills, which is obviously useful to a primitive human that needs to learn a lot of skills. We see this formalized in games all the time, from the overt Experience Points and Levels to finding new items or buying new weapons that give us better stats or new capabilities.
- **Finding Shortcuts**. Finding novel, undiscovered ways to work around problems in ways that take less effort than normal helped primitive humans to conserve their energy; in that sense, laziness can be a virtue. Ironically, in games, this often takes the form of deliberate rule-breaking and cheating.
- **Griefing**. Like other forms of competition, putting other people down is a way to show dominance and superiority over your peers. (Yes, some of us find it annoying and immature, but cavemen are not exactly known for their emotional sensitivity.)

This chapter was adapted from Level 8 of Ian Schreiber's *Game Design Concepts* course.

11. Kinds of Players

In "Hearts, Clubs, Diamonds, Spades: Players Who Suit MUDS," Richard Bartle identified 4 types of players. As with kinds of fun (and definitions of games), we find no shortage of people willing to advance their own theory of player types. Why read Bartle, then, and not someone else? First, Bartle's was the first essay of its kind to gain widespread interest and acceptance, so it is important historically; second, because there are certain aspects of it that make for interesting dissection.

Let us look at the four proposed types of players in a MUD (or MMO):

- **Achievers** find it enjoyable to gain power, level up, and generally to "win" the game (to the extent that an ongoing, never-ending game can be "won").
- **Explorers** want to explore the world, build mental maps of the different areas in their heads, and generally figure out what is in their surroundings.
- **Socializers** use the game as a social medium. They play for the interaction with other players. The gameplay systems are just a convenient excuse to get together and play with friends.
- **Killers** (today we call them "griefers") derive their fun from ruining other people's fun.

What is the motivation of each player type? Why do they do what they do? This relates back to the different kinds of fun.

Comparing the lists of Bartle's player types and MDA's 8 kinds of fun, we see parallels. Achievers favor Challenge fun. Explorers seem to like Discovery fun. Socializers are all about Fellowship fun. And Killers… well, they don't map to a specific kind of fun in MDA, but the Griefing fun that we proposed as an addition seems to work well.

Other player type schemes show similar correlations: each "player type" is really a kind of fun, or a combination of several kinds of fun,

personified. The two concepts (player types and kinds of fun) are really the same concept expressed in different ways.

This suggests that you can start with a list of kinds of fun, and invent new player types based on some combination of fun types. Car racing games combine Sensation and Challenge fun; we could propose a "Racer" player type as the kind of player who likes these kinds of games. And then we could make a guess that other games, such as "Xtreme Sports," might appeal to the same player type since they have a similar "fun signature."

You could also go the other way. If you manage to isolate a new player type (i.e. a pattern of play that appears in a nontrivial percentage of your playtesters), by studying that type and what the players are doing, you may be able to discover new kinds of fun.

Which Comes First?

If we can go back and forth between player types and kinds of fun, we may wonder if this is a classic chicken-and-egg problem. Is it better to start with players, or fun?

Consider this: as game designers, we create rules (mechanics). The rules create the play dynamics when set in motion, and those cause the aesthetic of fun in the players. The things that we create, are a root cause of fun. Therefore, it is the *kinds of fun* that are of greatest concern to us.

We do not create players. (Well, those of us who are parents could say that they do, but you know what we mean.) As game designers, our rules do not create new players or player types. Therefore, any list of player types is only useful to the extent that it is correlated with kinds of fun.

This chapter was adapted from Level 8 of Ian Schreiber's *Game Design Concepts* course.

PART III

GAME DESIGN

12. What is Game Design?

We will use the word "design" a lot in this text, and unfortunately it is a term that is a bit overused, so we must clarify what we mean by design. As Ian Schreiber and Brenda Braithwaite say in their book *Challenges for Game Designers*, game design is the creation of the rules and content of a game. It **does not** involve programming, art or animation, or marketing, or any of the other myriad tasks required to make a game. All of these tasks collectively can be called "game development" and game design is one part of development.

Multiple Types of Game Design

As mentioned in *Challenges*, there are many tasks associated with game design: system design, level design, content design, user interface design, world building, and story writing. You could fill several textbooks with any one of these, so this text will *not* be a full treatment of the entire range of game design. The majority of this course focuses on **system design** (also sometimes called "systems design" or "core systems design").

System design is about defining the basic rules of the game. What are the pieces? What can you control? What actions can you take on your turn (if there are "turns" at all)? What happens when you take each action, and how does it affect the game state? In general, system design is the creation of three things:

- Rules for setup. How does the game begin?
- Rules for progression of play. Once the game begins, what can the players do, and what happens when they do things?
- Rules for resolution. What, if anything, causes the game to end? If the game has an outcome (such as winning or losing), how is that outcome determined?

What is a Game Designer?

As you may have noticed, game design is an incredibly broad field. Professional game designers sometimes have trouble explaining

what they do to their families and friends. Part of the reason for this is that they do so many things. Here are some analogies I've seen when trying to explain what it is like to be a game designer:

- Game designers are **artists**. The term "art" is just as difficult to define as the word "game"... but if games can be a form of art (as we saw in Costikyan's definition, at least), then designers would be artists.
- Game designers are **architects**. Architects do not build physical structures; they create blueprints. Video game designers also create "blueprints" which are referred to as "design docs." Board game designers create "blueprints" as well – in the form of prototypes – which are then mass-produced by publishers.
- Game designers are **party hosts**. As designers, we invite players into our space and try our best to show them a good time.
- Game designers are **research scientists**. As I will touch on later today, we create games in a manner that is very close to the scientific method.
- Game designers are **gods**. We create worlds, and we create the physical rules that govern those worlds.
- Game designers are **lawyers**. We create a set of rules that others must follow.
- Game designers are **educators**. As we will see later when we start reading *Theory of Fun*, entertainment and education are strongly linked, and games are (at least sometimes) fun because they involve learning new skills.

If game design is all these things, where would it fit in a college curriculum? It could be justified in the school of education, or art, or architecture, or theology, or recreation management, or law, or engineering, or applied sciences, or half a dozen other things.

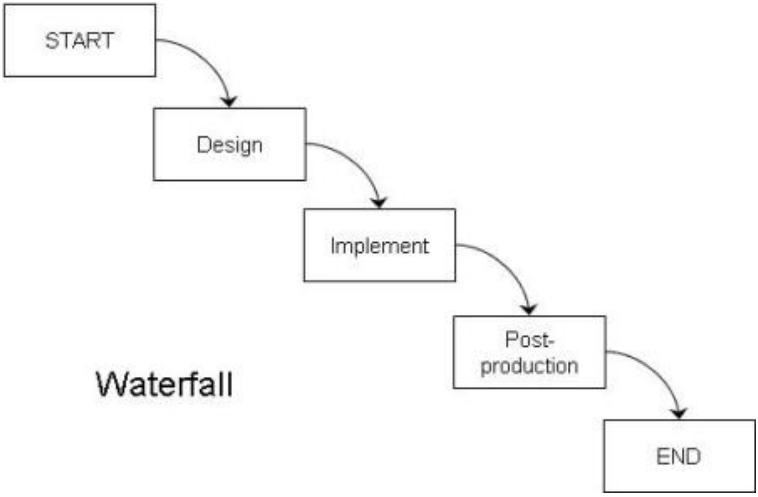
Is a game designer all of these things? None of them? It is open for discussion, but it seems clear that game design has *elements* of

many other fields, but it is still its own field. And you can see just how broad the field is! As the field of game design advances, we may see a day where game designers are so specialized that "game design" will be like the field of "science" – students will need to pick a specialty (Chemistry, Biology, Physics, etc.) rather than just "majoring in Science."

Speaking of Science...

How is a game designed? There are many methods.

Historically, the first design methodology was known as the **waterfall** method: first you design the entire game on paper, then you implement it (using programming in a video game, or creating the board and pieces for a non-digital game), then you test it to make sure the rules work properly, add some graphical polish to make it look nice, and then you ship it.

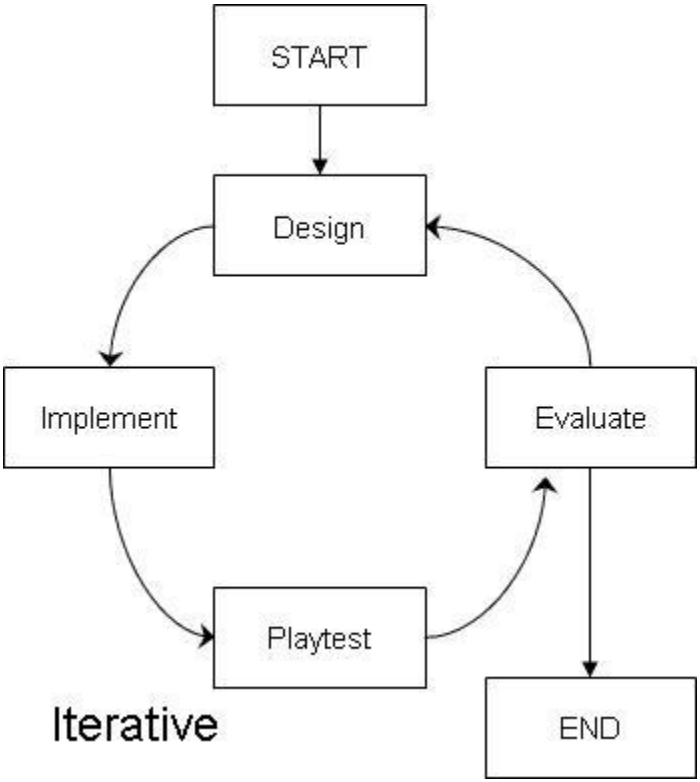


Waterfall

Waterfall is so named because, like water in a waterfall, you can only move in one direction. If you're busy making the final art for the game and it occurs to you that one of the rules needs to change,

too bad – the methodology does not include a way to go back to the design step once you are done.

At some point, someone figured out that it might be a good idea to at least have the *option* of going back and fixing things in earlier steps, and created what is sometimes known as the **iterative** approach. As with waterfall, you first design the game, then implement it, and then make sure it works. But after this you add an extra step of evaluating the game. Play it, decide what is good and what needs to change. And then, make a decision: are you done, or should you go back to the design step and make some changes? If you decide the game is good enough, then that is that. But if you identify some changes, you now go back to the design step, find ways to address the identified problems, implement those changes, and then evaluate again. Continue doing this until the game is ready.



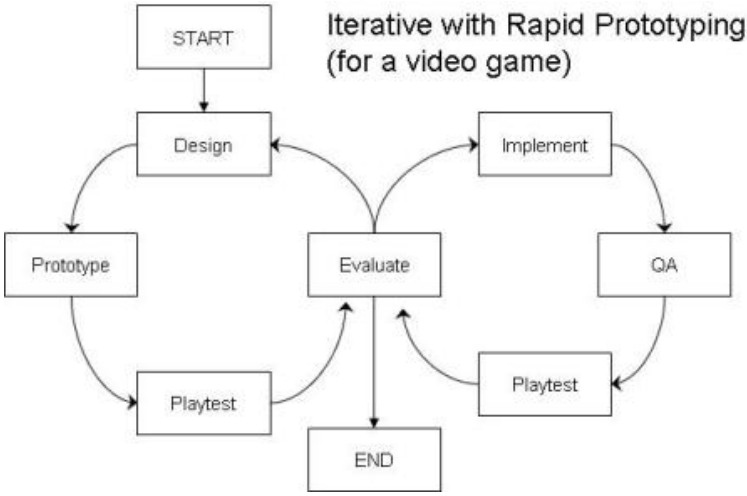
If this sounds familiar, it is because this is more or less the Scientific Method:

1. Make an observation. ("My experience in playing/making games has shown me that certain types of mechanics are fun.")
2. Make a hypothesis. ("I think that this particular set of rules I am writing will make a fun game.")
3. Create an experiment to prove or disprove the hypothesis. ("Let's organize a playtest of this game and see if it is fun or not.")
4. Perform the experiment. ("Let's play!")
5. Interpret the results of the experiment, forming a new set of observations. Go back to the first step.

With non-digital (card and board) games, this process works fine, because it can be done quickly. With video games, there is still one problem: implementation (i.e. programming and debugging) is expensive and takes a long time. If it takes 18 months to code the game the first time and you only have two years, you will not get a lot of time to playtest and modify the game.

In general, **the more times you iterate, the better your final game will be.**

Therefore, any game design process should involve iterating (that is, going through an entire cycle of designing, implementing and evaluating) as much as possible, and anything you can do that lets you iterate faster will usually lead to a better game in the end. Because of this, video game designers will often prototype on paper first, and then only get the programmers involved when they are confident that the core rules are fun. We call this **rapid prototyping**.



Iteration and Risk

Games have many kinds of risk associated with them. There is **design risk**, the risk that the game will not be fun and people won't like it. There is **implementation risk**, the possibility that the development team will not be able to build the game at all, even if the rules are solid. There is **market risk**, the chance that the game will be wonderful and no one will buy it anyway. And so on.

The purpose of iteration is to lower design risk. The more times you iterate, the more you can be certain that the rules of your game are effective.

This all comes down to one important point: the greater the design risk of your game (that is, if your rules are untested and unproven), the more you need iteration. An iterative method is not as critical for games where the mechanics are largely lifted from another successful game; sequels and expansion sets to popular games are examples of situations where a Waterfall approach may work fine.

That said, most game designers have aspirations of making games that are new, creative, and innovative.

Why This Course is Non-Digital...

Some of you would rather make board games anyway, so you don't care how video games are made. But for those of you who would love to make video games, you may have wondered why this text is focused on making board and card games. Now you know: it is because iteration is faster and cheaper with cardboard.

Later in this text, we will discuss in detail methods of paper prototyping, both for traditional board games and also for various types of video games.

There is another reason why we will concentrate primarily on board and card games. This is a course in **systems design**, that is, creating the rules of the game. In board games, the rules are laid bare. There may be some physical components, sure, but the play experience is almost entirely determined by the rules and the player interactions. If the rules are not compelling, the game will not be fun, so working in this medium makes a clear connection between the rules and the player experience.

This is not as true in video games. Many video games have impressive technology (such as realistic physics engines) and graphics and sound, which can obscure the fact that the gameplay is stale. Video games also take much longer to make due to programming and art/audio asset creation.

The connection between rules and player experience is also muddied in tabletop role-playing games. Keep in mind that an RPG is essentially a collaborative story-telling exercise (with a rules system in place to set boundaries for what can and can't happen). As such, a wonderful system can be ruined by players who have poor story-telling and improv skills, and a weak system can be salvaged by skillful players. As such, we will stay away from these game genres.

This chapter was adapted from Level 2 of Ian Schreiber's *Game Design Concepts* course.

13. The Design Process: Game Ideas

One way to create a game is to take a bunch of elements, throw them together, and call it a game. The results of this type of design can be expected to be hit-and-miss. Some games created using such a process might be ok but many of the games will be terrible.

Is there a process that can be followed that will lead to *better* games? There is the iterative process that we saw in the previous chapter, but we have not gone into detail on any of the iterative steps (design, playtesting, evaluation). How exactly do you come up with an initial design? What is the most effective way to playtest? When evaluating a game, what do you look for, and how do you know what to change? These are the things we will be concerned with throughout the rest of this section of the text.

Generating Ideas

The first thing that happens in a design is that you must come up with the basic core of an idea. This isn't necessarily fully-formed, but just a basic concept. There are many different starting points for a game's design. Here are some examples, in no particular order:

- Start with the **core "aesthetics"** – what do you want the player to feel? How do you want them to react? What should the player experience be like? Then work backwards from the player experience to figure out a set of rules that will achieve the desired aesthetic. Think about the best experience you've ever had while playing a game; what game rules led to that experience?
- Start with a **rule** or **system** that you observe in everyday life, particularly one that requires people to make interesting decisions. Look at the world around you; what systems do you see that would make good games?

- Start with an **existing, proven design**, then make modifications to improve on it (the "clone-and-tweak" method). This often happens when making sequels and ports of existing games. Think of a game that you thought had potential, but didn't quite take the experience as far as they could; how would you make it better?
- Start with **technology**, such as a new game engine (for video games) or a special kind of game piece (like a rotateable base for miniature figures). Find a way to make use of it in a game. What kinds of items do you have lying around your living space that have never been used in a board game before, but that would make great game "bits"?
- Start with **materials** from other sources, such as existing art or game mechanics that didn't make it in to other projects. Design a game to make use of them. Do you have an art portfolio, or earlier game designs that you didn't turn into finished products? What about public domain works, such as Renaissance art? How could you design a game around these?
- Start with a **narrative** and then design game rules to fit, making a story-driven game. What kinds of stories work well in games?
- Start with **market research**: perhaps you know that a certain demographic is underserved, and want to design a game specifically for them. Or maybe you just know that a certain genre is "hot" right now, and that there are no major games of that type coming out in a certain range of dates, so there is an opportunity. How do you turn this knowledge into a playable game?
- Combinations of several of these. For example, starting with **core aesthetics** and **narrative** at the same time, you can make a game where the story and gameplay are highly integrated.

When you think of new ideas for games, what kinds of ideas do you

have? What are your starting points? What does this say about you as a designer, and the kinds of games you are likely to make?

Other Methods of Idea Generation

If you are stuck with "designer's block" (the game design equivalent of "writer's block") there are a number of strategies you'll see mentioned in various places. Here are a few:

- Keep a permanent collection of all of your ideas for games, mechanics, stories, and everything else. Look back through it from time to time to see if there's anything from years ago that you can use. Add to it whenever an idea occurs to you that you can't use immediately, but that you want to return to later.
- Think of something random. Try to find a way to integrate it into your game.
- Do some research. Learn about some aspect of the game in more depth, and you will likely find new ideas.
- Go back to the basic. Think of the formal elements of your game. What are the player goals? Rules? Resources? And so on. Note that you'll need to define these anyway in order to have a game, so by focusing on these one at a time it may give you new questions to answer.
- Formalized brainstorming, either alone or in a group. Some people swear by this method, while others say the results are questionable. The best I can say is that the results are highly unpredictable… as is the case with most R&D.
- Think critically about games. You may have this textbook on game design that contains some of what I have learned over the years, but you should write your *own* book over the course of your lifetime (whether you publish it or not, at least keep it for yourself). When you discover something that does or doesn't work in a game and you think you can identify the root cause as a "law" (or at least a guideline) of game design that is broadly applicable, write it down! If you don't know why, write *that* down too, and come back to it periodically until you find the answer.

- Play lots of games! But… play as a *designer* and not just a player. Don't just play for enjoyment. Instead, play critically. Ask yourself what choices were made by the designer of the game, and why you think those choices were made, and whether or not they work. Play games in genres that you don't like or have never tried, and try to figure out why other people find them fun. Also, published hint guides can be useful to read – they are basically glorified design documents that detail all of the systems of a game!
- And lastly, practice. Work on your own projects. The more you make games, the better you get at making them… just like any other art form.

This chapter was adapted from Level 4 of Ian Schreiber's *Game Design Concepts* course.

14. Formal Abstract Design Tools

In the Gamasutra article *Formal Abstract Design Tools*, Doug Church advocates for the creation of a set of design tools for making games. First, he mentions three aspects of games that are worth putting in our design toolbox:

- **Player intention** is defined as the ability of the player to devise and carry out their own plans and goals. We will come back to this later on in this text, but for now just realize that it can be important in many games to allow the player to form a plan of action.
- **Perceivable consequence** is defined in the reading as a clear reaction of the game to the player's actions. Clarity is important here: if the game reacts but you don't know how the game state has changed, then you may have difficulty linking your actions to the consequences of those actions. "Perceivable consequence" is known by a more common name: **feedback**.
- **Story** is the narrative thread of the game. Note that a game can contain two different types of story: the "embedded" story (created by the designer) and the "emergent" story (created by players). Emergent story happens, for example, when you tell your friends about a recent game you played and what happened to you during the play: "I had taken over all of Africa, but I just couldn't keep the Blue player out of Zaire." Embedded story is what we normally think of as the "narrative" of the game: "You are playing a brave knight venturing into the castle of an evil wizard." Church's point is that *embedded* story competes with intention and consequence – that is, the more the game is "on rails", the less the player can affect the

outcome. This statement is a clearer articulation of what Costikyan meant in "I Have No Words" when he said that games are not stories.

Here is an example of why player intention and perceivable consequence are important. Consider this situation: you are playing a first-person shooter game. You walk up to a wall that has a switch on it. You flip the switch. Nothing happens. Well, actually something *did* happen, but the game gives you no indication of *what* happened. Maybe a door somewhere else in the level opened. Maybe you just unleashed a bunch of monsters into the area, and you'll run into them as soon as you exit the current room. Maybe there are a series of switches, and they all have to be in *exactly the right pattern* of on and off (or they have to be triggered in the right order) in order to open up the path to the level exit. But you have no way of knowing, and so you feel frustrated that you must now do a thorough search of everywhere you've already been... just to see if the switch did anything.

How could you fix this? Add better feedback. One way would be to provide a map to the player, and show them a location on the map when the switch was pulled. Or, show a brief cut scene that shows a door opening somewhere. I'm sure you can think of other methods as well.

On another subject, Church also included an interesting note at the end of the article about how he values beta testing, and half of his readers found the first two pages slow, so start at page 3 if you're in that half. This would be an example of **iteration** in the design of this essay, of exactly the sort we talked about.

This note was likely partly in jest, but let's take it at face value. There's a slight problem with this fix: you don't see the note until you've already read all of the way through the article, and it's too late to do anything about it. If Church were to iterate on his design a second time, what would you suggest he do?

This chapter was adapted from Level 3 of Ian Schreiber's *Game Design Concepts* course.

15. The Design Process: Prototyping

Remember, the more times you can iterate on your idea, the better the final game will be. Once you have a basic idea, the next step is to get it in playable form as quickly and cheaply as possible. That will leave you with as much time as possible to playtest and iterate.

As mentioned last time, iteration is the most critical for those parts of your game that have high **design risk**. For "clone-and-tweak" games where you are mostly lifting gameplay from an existing game, rapid prototyping is less important. This does not mean that "clone" games do not benefit from iteration, but simply that you should use it selectively in those areas where you *are* innovating.

"Laws" of Prototyping

Remember that the entire purpose of prototyping is to maximize the number of iterative cycles. Corollary: do everything you can to reduce the time required in each iteration. Now, consider that each iterative cycle consists generally of four steps: design, prototyping, playtesting, and evaluation. Of these steps, where can you save time?

- You can't really reduce the time it takes to design the rules of the game, without compromising your goals. You can't rush creativity.
- You *can* reduce time spent in playtesting by being efficient about scheduling and designing playtests to give maximum information for minimum play time… but there is a natural limit to this, and beyond a certain point you can't rush through playing the game.
- Evaluation doesn't take very long; you're making a simple yes/no decision of whether the game is "done" or "good enough"

based on playtest results. There is little to be gained by rushing through this further.
- So, that leaves reducing the time it takes to create a prototype.

Some things to keep in mind when building a playable prototype:

- Build it as fast as possible. Cut corners. Make it as ugly and cheap as you can get away with.
- Minimize what you need to build. Only do what is absolutely necessary to evaluate your game. If you're trying to test out a new combat system, you do not need to build the entire exploration system. If you're making a card game, hand writing on index cards is faster to make than typing everything into Powerpoint, printing on heavy card stock, and cutting it all out manually. There is a time and place for making nice-looking components, and the early stages of game design is **not** that time or that place.
- Make your prototype easy to change. You **will** find problems in playtesting, so make it easy to adjust on the fly.

All of these guidelines push designers towards one inevitable direction…

Prototyping in Paper

You can call it "paper" or "cardboard" or "non-digital" or "analog" or any number of things, but the idea is to have a physical, tabletop game that is playable without computers (or at least, without requiring programming code). Programming is wonderful and powerful but it is also slow and expensive in comparison to paper prototypes. Here are some advantages of paper prototyping:

- It is cheap. Most systems can be prototyped with little more than a pencil and some paper, although I will give suggestions for other components for those of you that have some money to spend.
- It's fast. You don't have to mess around with programming, or

layouts, or artwork. Just write a few words on a scrap of paper.
- It's easy to change. Don't like one of your numbers? Erase it and write in a new one.
- There is no guilt about throwing it away. You came up with an idea that didn't work? Oh well, you lost a whole half hour. Big deal. It's like making stick-figure drawings: if your first attempt at drawing a stick figure doesn't work, it only took you a few seconds, so just cross it out and try again.
- Paper can be used to model most gameplay systems. Yes, even most of the ones we normally associate with being specific to video games.
- By making something *playable*, you are forced to actually **design the systems**. No more handwaving of "this game will have 50 undefined cards". You have to actually do your job as the game designer, and design the game!

Limitations of Paper

Paper prototypes do have some limitations that you should be aware of:

- They cannot always handle "twitch" (dexterity or timing based) mechanics... although be aware that there are many dexterity-based non-digital games. Consider the similarities and differences between the *Street Fighter* series of video games, and James Ernest's real-time card battle game *Brawl*. Some things carry over well... others, not so much.
- Information that is hidden to both players but that still requires bookkeeping, such as the "Fog of War" mechanics prevalent in Real-Time Strategy video games. Again, note that this can sometimes be worked around – the classic children's game *Battleship* has "fog-of-war-like" mechanics, and the board game *Clue* has information hidden from all players.
- Extremely complex calculations are tedious on paper, and the systems that use them may be better suited to "prototyping" in a spreadsheet program like *Excel*. However, if the complex

systems are a necessary and core part of the game, it may be a sign that "the computer is having more fun than the player" (to quote Sid Meier), and that perhaps some simplification would make the game more accessible.
- "Eye candy" such as high-quality art and animation is obviously not prototyped easily with stick-figure drawings and handwritten cards. Then again, these are not part of the game mechanics. If your game relies on visuals rather than systems, that is a sign that you are not doing a strong enough job as the systems designer.
- Paper prototypes are not very well suited for testing the user interface (UI) of a video game. Computer UIs are dynamic, but paper is static. You can get an idea of the visual layout with some paper sketches, but to know how it will actually be used on a computer, you'd need a digital prototype.

As you can see, the advantages of paper prototyping are very general and the limitations are specific, so the ability to prototype in paper is an important skill for any game designer to develop, whether they work in video games or board games or anything in between.

Prototyping Realtime Systems

For a turn-based game like *Battleship*, a non-digital prototype is easy enough to put together. What if you wanted to prototype a First-Person Shooter video game like *Halo*? Is there any *possible* way to do that on paper, when most of the game is running around and shooting things in real time? The answer is yes, absolutely. Here are some hints:

- One "turn" of a board game is equivalent to some amount of time (say, 3 seconds) of real-time play
- For "twitch" mechanics like dodging and accuracy that require accurate timing, either a player succeeds or fails at these based on how difficult they are and how skilled the player is. This can be modeled with a random die roll. Note that even though the

video game's system is not random at all, it may as well be random from the *opponent's* perspective: if I shoot at you and you either do or do not successfully dodge, I have no control over that.
- Many real-time games take place on an open 3D map that is not subdivided into "spaces". This does not prevent you from making a game board that has spaces anyway.

A Short Note about Grids

There are many ways to make a game board, but here are three common ways to get you started:

- Subdivide into a grid of squares. Square grids are easy to navigate and are familiar to most players, so they will not intimidate casual players as much as some other methods. For grids that include lots of obstacles and movement challenges, grids are ideal because it is easy to block off a path: a single impassable square forces you to go quite a bit out of your way to get to the other side. The drawback of squares is that you inevitably run into a problem with diagonal movement: does it count as one space or two in order to move diagonally? One space feels too fast; two spaces feels too slow. (The actual value is the square root of 2, or about 1.4 spaces… but if you're dealing with whole-number values this obviously does not work.)
- Subdivide into a grid of hexes. Hexes have some nice mathematical properties to them, in that something that is 3 hexes away is *always* that many hexes, no matter which of several paths you take; this gets around the "how fast to move along a diagonal" problem of square grids. On the down side, hex boards make it much easier to move around obstacles, so movement is a lot less constrained. This may be desireable or not, depending on the nature of your game. Also, hexes are quite "geeky" and are likely to put off players who are not that experienced with this style of play.

- Open area, no board. Use a tape measure instead, and move your pieces a certain number of inches (or centimetres, or what have you) per turn. This gives the most fluid and precise movement, although it has many of the same disadvantages as hex maps, and is also vulnerable to someone accidentally bumping the table and sending pieces slightly off of where they were.

Adding Features versus Keeping It Simple

Many early prototypes are simply begging for extra features, such as health and ammo and various other resources. Why not start with all of these extra systems already in place, as opposed to starting with just the simple core system? There are a few reasons to start with a simple, core rule set and then add on one rule at a time, instead of trying to design the entire game in one big effort:

- If the basic, core rules don't work, then adding extra rules on top of it will generally not make it work. Get the basics working first, *before* you start adding complexity.
- In fact, if you build extra rules on an unstable foundation, the real underlying problems in your design could be obscured! Something might *seem* wrong, but if there are a lot of systems and resources and game objects it can be hard to tell if you're experiencing a problem with the core mechanics, or the balance of a particular resource, or the design of the map, or something else.

Early on in a design process, it's generally better to keep things as simple as possible. For every rule or mechanic or object or resource that you want to include, ask yourself: is this *really* necessary *right now*? At this point, let your laziness override your creativity. It is far easier to add something to your design than to take it away, so add the minimum possible to have a working, playable game.

If you have trouble with this, try writing down a list of all of the ideas you have that you want to include in the game, and then cross

off as many as you can. Ask if whatever items are left on your list would make a complete, playable game. If so, try to cross off more, until you absolutely can't anymore.

It may also help to run your idea by another designer who is not personally and emotionally attached to your pet idea. Invite them to be merciless in deciding which of your rules can be trashed. For the purposes of this course, you can offer a trade with any colleagues in your area: you look at my prototype, I'll look at yours!

Moving Forward

Once you have the core gameplay, and it *works*, then you can add new features. The temptation at this point is to add everything you originally thought of. Resist this temptation. Instead, add *one* new feature, and playtest again until the new feature works, or you have decided that it doesn't work and it needs to be abandoned.

Why not add everything at once? Because every new thing you add may have some problems with it. If you only add one new rule and a critical game system becomes broken in playtesting, you know *exactly* where the problem is, because you only changed one thing. If you add ten new rules and something breaks, it's harder to isolate which rule (or combination of rules) caused the problem. Incidentally, this part is similar to programming: if you write code in small chunks and then unit test, it's easier to find bugs than if you write ten thousand lines of code between tests.

Yes, this is tedious. You have to playtest, then change one rule, then playtest again, then change another rule, and keep doing this dozens (or even hundreds) of times. The first few playtests are fun, but you will quickly become sick of the whole business. This is part of the process of design. Sometimes, game design is hard work that is not particularly fun. This is something you need to accept if you have aspirations to become a professional designer. Just remember that the purpose of this is to make a game that *is* fun, and if it's not there yet, that should be your incentive to change something and playtest again until you reach your goal.

This chapter was adapted from Level 4 of Ian Schreiber's *Game Design Concepts* course.

16. The Design Process: Playtesting

The word "playtesting," like the word "game," is overused and can mean different things to different people. In general, the term covers any activity where you are playing a game in progress for the purpose of improving it. But different playtests may have different goals, and it is important to know what your goals are before you do anything.

The concepts in these descriptions of different kinds of playtesting are more important than the labels.

Bug Testing (or Quality Assurance)

The purpose of QA is to find errors in the game's behavior relative to its design. "Fun" does not enter the equation. If the designer says that the game should do one thing and it actually does another (even if what the game *is* doing may be superior), that is a bug that needs to be identified.

Normally, we think of bug testing as specific to video games. Board games do have a corresponding kind of testing, where the purpose is to find holes in the rules and dead ends in gameplay – gaps in the game that the designer did not cover.

Focus Testing

In a focus test, you bring together players that are part of the target audience's demographic in order to determine how well a game serves their needs. This is normally done for marketing purposes, but if game designers are involved it can also help to make the game more enjoyable for that particular demographic.

Usability Testing

In a usability test, players are given specific tasks to accomplish in an attempt to see whether they understand how to control the game. This is done frequently in the greater software industry to make sure that a piece of software is easy to learn and easy to use.

Video games can take advantage of this as well, and results from a usability test can be used to either change the controls or modify the early levels to teach those controls more effectively.

In board games, usability is doubly important, because there is no computer to respond to player input for you. If you misunderstand how houses work in *Monopoly* and place them on Community Chest spaces, the game will not stop you. By observing players who are trying to play your game, you can learn a lot about how to design the various game bits so that they are easy and intuitive to use.

Balance Testing

A fun game can quickly become boring if some kind of play exploit exists that lets a player bypass most of the interesting choices in the game. If only one strategy can win and it is just a matter of which player follows that strategy the best, it is not as interesting as if there are multiple paths to victory. Likewise, if one player has a clear advantage over the others, it is important to identify that so that players do not feel the game is being unfair. The purpose of this kind of test is to identify imbalances in the game so that the designer can fix them.

Fun Testing

A game can be usable, balanced and functional and still be uninteresting. That elusive "fun factor" may be hard to design intentionally, but when people are playing the game it is pretty obvious whether they are having fun or not. Certain aspects of the game may be more fun than others, so it is also important to figure out what parts of the game need to stay the same… not just what to change.

All of these forms of testing have some elements in common. Best practices are similar if not identical. All are important to the success of a project. So why make a distinction?

The reason is that each is appropriate at different stages of completion in a project. Each kind of testing has different goals, and you need to know what your goal is before you can achieve it.

Order of Effects

When should you do which kind of playtesting? What order do

you do them in? A lot depends on your particular project, so some of this will be up to your judgment as the designer. However, there are some guidelines.

- Very early on in the project, you need to make sure your project will meet its design goals (usually the "design goal" is to make a game that's fun to play). Testing for fun is necessary to make sure you do not spend a lot of time building on the wrong foundation. If you are making a game for a specific market, focus testing may be involved at an early stage as well, simply to ask the target audience if a game with a particular concept sounds interesting to them at all.
- Once you know that you have something, you need to solidify the mechanics. Design the whole game, making sure that all the details are taken care of. Test for "bugs." (Note that bug testing in software projects is often done continually throughout the project, increasing in intensity toward the end. Non-digital games are easier to "debug" though, and a "bug" can stop a playtest in its tracks, so it is important for us to have a complete set of rules early in the process.)
- Once the game is fun and the design is complete, gradually shift from testing for fun to testing for game balance. Make sure that all the numeric values and player abilities are where you want them to be.
- When the game is working and balanced, towards the end, you'll want to think more about the usability of the game. When you change usability you are not changing any mechanics, merely the way those mechanics are presented visually to the players. This is an important step that is often neglected. If you've ever encountered a game that you could only learn by being taught by another player (as opposed to reading the rules yourself), that is the kind of usability failure you want to avoid in your own projects. You may also do additional focus testing at this time, to make sure that the theme and visual elements of the game appeal to the target

audience.

Remember, these are just guidelines. If it is incredibly important that your game be well received by a particular demographic, for example, you may be doing focus testing throughout the project at all stages. Do not let this order of things be your master.

Different Kinds of Playtesters

As there are different kinds of testing, there are also different kinds of testers. Each kind of tester has their own strengths and weaknesses, and some are more important for some kinds of testing than others.

- Yourself. You are your own most valuable playtester. Do not forget your ability to play your game on your own. You know your game better than anyone.
- Other game designers. If you are lucky enough to personally know some other skilled game designers, you can get some very useful testing done through them. They are able to critically analyze your game and propose design solutions. (If you do not know any professional designers, perhaps you can at least make contact with other participants of this course.)
- Close friends, family, and confidantes. People close to you who are willing to provide their time to test your game are very useful. They are approachable and can make themselves available as a favor to you. Take good care of them, and do not abuse their kindness. Note that these people may not fall into any of the other categories, so while they are good for early tests, they may not be appropriate in more focused testing for bugs or balance since they may not know what to look for.
- Experienced gamers. Skilled game players are great at finding exploits and dominant strategies in a game, and are appropriate for balance testing.
- Complete strangers. People in your target audience are appropriate for focus testing and usability testing, and they are absolutely critical when testing for fun. Finding them can be

tricky, though, because it is not in most of our natures to just walk up to someone we've never met and ask them to play a game.

Order of Familiarity

In general, you will want to go through testers in order from more to less familiar. Test with yourself first, then with close friends, then with acquaintances that are useful (because they are designers, gamers, or part of the target market), and then with strangers.

If you show your work to other people too early, it will likely be in such a rough state with multiple design flaws and holes in the rules that it will waste their time and frustrate them, and you want to treat your playtesters better than that. Also, if you start playtesting with strangers too early in the process, you may not get useful feedback – if your game prototype is in a rough state with only crude art and components, for example, the playtesters may be so busy commenting on the poor quality of the pieces that they will not be able to concentrate on the gameplay.

At this point you might be tempted to just do *all* of the playtesting by yourself, so that you don't need to rely on other people or keep track of them. In practice, the designer eventually gets too close to their own project and is so familiar with the game's systems that they can miss some really obvious flaws. If you keep the same set of playtesters for long enough, they will suffer from this problem as well. You need to bring in fresh sets of eyes to look at your game on a continuing basis throughout the project.

This chapter was adapted from Level 12 of Ian Schreiber's *Game Design Concepts* course.

17. The Design Process: More About Playtesting

In the early part of playtesting, when you are playing the game with your team, here are some things you should be looking for:

Does the game meet your design goals?

Is it fun, at least for you? While you are not the ideal playtester to judge effectiveness most of the time, if you are not having fun then most other people will probably not either.

Are there any holes in the rules?

A "hole" is a situation where the rules simply do not say how to proceed. For example, perhaps one of your rules is that a player's army can attack another player's army, but you don't yet have rules for resolving the attack. What happens in this case? In practice, what happens is that the players sit around and wait while the designer figures out what to do!

As an example, consider these rules for Tic-Tac-Toe played on a 4×4 grid:

- Players: 2
- Objective: Get a straight line of symbols.
- Setup: Draw a 4×4 square grid.
- Progression of play: On your turn, place your symbol ("X" or "O") on an empty square.
- Resolution: If either player on their turn has a set of four of their symbol in a straight line (across, down, or diagonally), they win.

If you try to play this game just following the rules, you'll quickly realize that you can't even start – nowhere does it say which player is X or O, or who takes the first turn! To fix this, you would add a situation to handle this. For example:

Setup: Draw a 4×4 square grid. Choose a player to go first, who is assigned the symbol "X". The other player is given the symbol "O".

Are there any dead ends?

A "dead end" is a game state where there is no way to proceed further, but the game is not resolved. Consider our revised 4×4 *Tic-Tac-Toe* rules above. Suppose that both players fill up all squares on the board without anyone winning. At this point the game cannot proceed, because the rules say a player must place their symbol on an empty square. There is no empty square, so the player cannot take a turn. But there is also no resolution, because neither player has won. In this case, a new rule would have to be added (such as: in the resolution, if neither player can make a legal move and no one has won, then the game ends in a tie).

Are any of the rules unclear?

It is natural for us to assume things that are in our head, to the point that we often forget to write them down in our rules. Try to look at your rules and see if there is anything you are assuming that your players might not.

Are there any really obvious rules exploits?

Is there a single strategy that wins the game easily? Try to find it. It's much less embarrassing if you find and fix it yourself, as opposed to having it discovered by your playtesters (or worse, your players *after* you release the game). Clarity and exploits are often hard to find in your own game; you tried to design this game to not have any problems, after all. Still, make an honest effort, and sometimes you will be rewarded by finding and fixing errors early (which saves a *lot* of time in the long run, leaving you more time to iterate on other parts of your design).

You might think that looking for exploits is something to do later in the project when balancing the game. Sometimes it is. It is a matter of degree. If an exploit is *so* powerful and *so* obvious that it prevents your playtests from giving you real information about your game, fix it now.

At this point in the project, you should have a playable prototype of your game, and a set of rules. You should have playtested on

your own with your team at least once, identified any really obvious problems, and iterated on your design. You should continue to do this until your design is at a point where you are confident that you can play all the way through without having to make major changes.

Once you reach that point, your goal shifts from "make this game work" to "make sure the core mechanics are fun" (or whatever your design goal happens to be, if not "fun"). Who would make the best playtesters to help with this?

Normal players (such as friends and family, or even complete strangers) are marginally useful here. By watching them, you can determine if they are having a good time and if your game is meeting its design goals. However, if there is a problem, a typical gamer will not be able to give you useful feedback other than "it's great" or "it sucks." It will be up to you as the designer to identify and fix the problems. Therefore, normal testers can be used if necessary, but their usefulness is limited.

Far better is to playtest with other game designers. Game designers can also let you know if the game is fun, *and* they can offer suggestions on where the problem points are and what can be changed to make your game better. You can often have wonderful discussion following the play of the game, on the design of your game and sometimes on game design in general. These kinds of discussions are important, and your game can get better much faster with them.

Being a Great Designer

As other people playtest your game, keep in mind the following:

- Your game is not perfect. If your game were perfect, you wouldn't need to playtest.
- There will be problems. The goal of playtesting is to find and eliminate those problems. If all your playtest did was confirm that your game is perfect, you have just wasted your own time and everyone else's.
- It is far better to identify problems in a small playtest, than for them to be found after the game is printed and ships to

millions of players.

- If one of your playtesters finds a major problem in your game, they have given you a great gift. Do not be hostile or defensive; be gracious.
- When a problem is identified by a playtester, your goal is not to verbally defend your game or to explain why the playtester is wrong. First, even if your playtester *is* "wrong," it probably means a lot of other players will also be "wrong" in the same way, and you can't ship yourself in a game box in order to explain your Grand Vision to everyone. Second, the playtester is probably right – they are seeing your game through fresh eyes, and are more likely to have an unbiased view of the game.
- If your playtesters do identify problems, the correct response is to write the issue down in your notebook... and then discuss your design goals with the playtesters so that you can get some ideas of how to preserve your goals while changing the game.
- Not all people are tactful. Sometimes people will say things about your game (or even about you, personally) that are downright hateful. Sometimes people will make fun of your game, or will taunt or berate you for a problem with your design. Keep in mind that, no matter how it is delivered, this is still extremely useful content.
- It takes a strong person to hear a statement like "your game sucks, it is the worst game I've ever played, and by extension *you* suck and you are nothing better than a waste of space" and to genuinely reply: "You have just helped me identify some major flaws in my game. **Thank you.**" Getting to the point in your life where you are emotionally strong enough to have an exchange like this should be one of your long-term goals as a game designer. You do not have to be like this right now. I'm not. But I have seen an exchange like this before from a great designer, and it made me realize how far I have to go.

Running a Great Playtest Session

If you want your playtesters to keep coming back for your future designs, be as respectful of their time as possible. Here are some things to consider:

- Before you show your game to other players, make sure the rules are fresh in your mind so that you do not need to look them up. Try explaining all of the rules to yourself in the mirror to make sure you can do it. This will save time, if it only takes you a couple minutes to explain rather than half an hour.
- If you already know there are problems (and you just don't have the solutions) or if you have specific design goals other than "make a fun game," let your playtesters know this up front. It will help them to be more aware of potential solutions.
- End your playtest as soon as you can. If you have received as much useful information as you are likely to after a half hour of play, stop there (even if the full game would last three hours). Remember that the purpose of the playtest is to identify problems, not to "play games." If you're not identifying problems, you are wasting everyone's time.
- Bring your playtest notebook and take good notes. You *will* forget everything that takes place, no matter how obvious your playtest results seem at the time, so make sure you write down every piece of information that you don't want to lose.

Being a Great Playtester

Here are some of the things you should keep in mind when testing other people's games:

- When testing, give the designer and the game your undivided attention. You would want others to extend the same courtesy to your game, after all.
- Don't leave in the middle of a test. Aside from being rude, it can throw off the results (not all games can gracefully handle it when a player leaves). At minimum, if you know you have

limited time or that you may get called away in mid-game, let others know this up front so they can handle it accordingly.
- Be as detailed as possible. Don't just say that the game is "fun" or "boring," try to analyze why. You should have enough of a background at this point to give meaningful feedback. Make use of your design skills!
- Allow some time after the game for discussion with the other testers and the designer. Talk about your play experience, and how it was related to the mechanics.
- Remember that there are many possible playtest goals. Are you playing to see if the game is fun? Are you playing to win? Are you playing to find holes in the rules? Play accordingly. We are so used to playing games in our own personal style, that it can be difficult to remember that there are other ways to play. Keep the goals of the playtest in mind.
- Be polite. Attack the *game* mercilessly, but do not attack the *designer*.

This chapter was adapted from Level 12 and Level 13 of Ian Schreiber's *Game Design Concepts* course.

18. How to Improve a Game

Recall that Greg Costikyan articulated six elements that an activity must contain if it is to be considered a game. In his article, "I Have No Words and I Must Design," he also articulated some things that we can think about adding to a game if we want to more fully engage the player.

Variety of Encounter

When we play a game, we typically repeat certain actions over and over. The action that the player repeats most often is called the **core mechanic** (which will be discussed more in a later chapter). For example, the core mechanic in *Monopoly* is that we roll the dice and move our piece that many spaces on the board. A number of different possibilities for our action arise when we land on a particular space. If the property that the space represents is not already owned by someone, we have a decision to make: should we buy the property or not? If the property is already owned, we must pay rent to the owner. If the space is not a property, we must follow the instructions on the space–pick up a Community Chest card or go directly to jail, for example. Costikyan calls this **variety of encounter**.

If the variety of encounter is not great enough, a game will quickly become boring. Think about the game *Tic Tac Toe*, for example. Little kids typically love the game until they suddenly don't. This is because there is actually little variety of encounter and once children learn the patterns for how to play the game optimally so that it always ends in a draw, they grow bored and no longer want to play. We might be tempted to say that's because the core mechanic is just one simple action–mark your symbol in a spot. But having a good variety of encounter doesn't always mean that the core mechanic is more than one simple action. *Tic Tac Toe* becomes boring because once the player marks their symbol in a spot, there are a limited number of possibilities for what happens next. There

are fairly small number configurations of the 3×3 board and after you have played the game for a while, you have seen all of those configurations numerous times.

Contrast this with *Chess* which also has just one simple action as its core mechanic–move one of your pieces to a new spot on the board. Like *Tic Tac Toe*, *Chess* is a game of perfect information, that is, none of the information about the game is hidden from the player. Like *Tic Tac Toe*, there is no randomness in *Chess*. And yet, *Chess* is a game that takes a lifetime to master and provides endless fodder for analysis. No one would ever say something similar about *Tic Tac Toe*. This is because the variety of encounter in *Chess* is much higher than *Tic Tac Toe*. The variety of movement rules for each *Chess* piece moves in a different way, the much larger board, and the goal of capturing the King result in a much higher variety of encounter in *Chess* than in *Tic Tac Toe*. These factors mean that the number of possible configurations of the game after a player takes their turn is very large and so it takes a lifetime of play to have encountered all of the possibilities. As Costikyan says, "[P]layers like to encounter the unexpected." This is the essence of variety of encounter.

When we think about variety of encounter in a game, Costikyan tells us to ask questions like: What things do the players encounter in this game? Are there enough things for them to explore and discover? What provides variety? How can we increase the variety of encounter?

Diplomacy

Not everything in a game needs to be totally good for one player while being totally bad for another. Costikyan argues that "Whenever multiple players are involved, games are strengthened if they permit, and encourage, **diplomacy**." Diplomacy is the idea that players can combine their efforts on some kind of action that is mutually beneficial. This might involve a trade that benefits both parties or ganging up on a mutual opponent. The decision of whether to engage in a temporary alliance is inherently

engaging. As alliances form and dissolve, the game become more interesting.

For example, in *Monopoly*, two players might trade properties so that they each form a monopoly. The damage cause by the opponent gaining a monopoly is outweighed by the player gaining their own monopoly. Both sides benefit from the trade even though both sides are also harmed. *Monopoly* allows this kind of trading but the rules of the game don't actually encourage it. The game might be strengthened if the rules were changed to encourage trading and bargaining.

Costikyan tells us to ask questions such as: How can players help or hinder each other? What incentives do they have to do so? What resources can they trade?

Color

Why is the card game *War* called *War*? (For the moment, ignore the fact that Costikyan would not consider it a game because it provides no decision-making opportunities for the player.) Each round of flipping over the top card of the deck is a battle between the two cards with the highest ranked card winning the battle. When the two cards are of equal rank, the players must each risk even more cards in what the rules call a "war." But the war metaphor in the game goes no further than this. What if we extended the metaphor by using a deck of cards that had drawings of different types of soldiers, each with a particular rank? Such a deck, although functionally equivalent to the regular deck of cards, adds **color** to the game.

Color is that set of things in a game that helps the player become immersed in what the game is about. Color adds to the setting or sense of place in the game. These details add to the game's emotional appeal. For example, *Monopoly* isn't really about anything. But calling the squares on the board *properties* and giving each of them a real place name provides the player with a sense that the game is about buying and selling real estate. The paper money, plastic houses and hotels, collection of rent, and so on add to the

sense that the game is about real estate. All of this is part of the color of the game and helps to emotionally engage the player.

Costikyan tells us to ask the following questions about color: How does the game evoke the ethos and atmosphere and pageantry of its setting? What can you do to make it more colorful?

Position Identification

Have you ever been a spectator of a team sport and said "We won!" when the team won? If so, you already understand what Costikyan calls **position identification**. To strengthen a game, to increase its emotional impact, encourage the player to identify with their "side" in the game.

One way to encourage player to identify with their side is to allow them to control a single token in the game. For example, some players get quite invested in the particular token they use in *Monopoly*. But it might be more challenging when the player controls multiple tokens. In *Chess*, for example, few players are saddened when they lose a pawn. But in *Chess*, each side has a color for their pieces to help them clearly identify their side. In addition, *Chess* doesn't take the metaphors of the player's pieces to a level of detail that weakens the player's focus on their overall goal. For example, even though one piece is called a knight, the player doesn't have to worry about whether the knight is getting enough food to eat. Instead, the player's point of view in the game focuses on using the special properties of the knight to capture the opponent's king. To strengthen a game, the game designer should think about how to help the player clearly identify their point of view within the game.

Costikyan tells us to ask the following questions: What can you do to make the player care about his position? Is there a single game token that's more important than others to the player, and what can be done to strengthen identification with it? If not, what is the overall emotional appeal of the position, and what can be done to strengthen that appeal? Who "is" the player in the game? What is his point of view?

Other Ways to Strengthen a Game

Costikyan also talks about simulation, role-playing, and

socializing as ways to improve a game. Simulation means having the game simulate some real world situation. Role-playing is a specific way of increasing a player's position identification by having them play a particular role. Socialization is encouraging players to interact with one another as they play the game. You can read more about these in Costikyan's original article.

This chapter was written specifically for this book.

19. A Note About Intellectual Property

At this point, some of you may be thinking that by talking about your game in class or posting your game online, you run the risk that someone will Steal Your Great Idea. How can you protect yourself from the threat of someone taking your basic idea, turning it into a working, sellable game, and leaving you with nothing?

One of the participants of Ian Schreiber's course, Dan Rosenthal, wrote an article that details the basics of IP (intellectual property) law as it pertains to games. The article admits to being US-centric, but the core idea (which is worth repeating here) should be sound no matter where you are:

> *Remember, ideas are not copyrightable, they're not trademarkable, not trade secretable, and both difficult and prohibitively expensive to patent. You can't protect them anyway, and you shouldn't try – instead you should try to come up with new ones, and start working on the good ones. Don't freak out when you see things like Game Jams, or this course and think "Ian says I should post my work to the discussion forum, but I came up with a Great Idea(tm) and I don't want other people to steal it." Ideas are commonplace in games, and the value of your idea is nothing compared to the value of the implementation of that idea, your expertise and hard work in developing it into something that's going to make you real money. But most importantly, our industry is very lateral, very tight-knit, very collaborative. You'll find people sharing their ideas at GDC, doing collaborative projects between studios, or using inspiration from one game's mechanics to improve another. Don't fight it. That's the way things work, and by embracing that open atmosphere, you'll be far better off.*

This chapter was adapted from Level 3 of Ian Schreiber's *Game Design Concepts* course.

PART IV
WHAT STUDENTS HAVE TO SAY

The following chapters were written by Plymouth State University students in CMDI-1105.

20. Digging Deep Inside the Game of Go Fish

JACOB COSTELLO

Go Fish is a very popular game for children. But there is more to this game than asking for random cards. There is a dynamic to this game and more meaning to it. If we truly analyze the game if Go Fish, we will see there are several parts of this game that many complicated and mature games have in common with Go Fish. In this analysis, I have stated important definitions from David Parlett, Clark C. Abt, and Bernard Suits. Their definitions of games states what games should contain, prevent from happening, and what should happen while you play a certain game. I have used these definitions because they all relate to the game of Go Fish and have several elements Go Fish contains.

The name of the game is Go Fish. The intended audience and the game are designed for children. It is made this way to make a simple card game for younger kids to understand. The only materials needed to play this game is a deck of playing cards, without the jokers. **The game is turn-based. A typical turn consists of a player asking the opponents for a certain card that they have, if the opponent has that card, they must give it to them.** The player keeps asking other players for any card they have until they have a "book" or a group of four to put down. If the opponent says "Go Fish" that player must pick up a card and it is the next persons turn. The player with the most books wins the game.

"A game has "ends and means": an objective, an outcome, and a set of rules to get there." David Parlett said this about games having an ending, where the game is over, and someone wins. In order **to win a game you need to have an objective, outcome and rules to**

complete the game. The objective of the game is to get the most books or groups of four. The player with the most groups of four wins, you need to have more books than your opponents to be victorious.

Go Fish is an easy game. The rules are simple, you use a full deck of cards, without the jokers, every player starts off with five cards, **one player at a time asks their opponents for cards that they have, if you don't have a specific card, you can't ask for it. Once you have a group of four cards, you place it down on the table.** You keep asking for cards until someone doesn't have the card you are looking for and they say "Go Fish" then it is the next person's turn. You can ask any person and it doesn't have to be in any order. Go Fish contains all the elements in the definition, there is an ending to Go Fish, when the pile of cards is gone, the player with the most books wins. There is the objective of getting the most books, the outcome is the end goal and what you are trying to accomplish, and there are a set of rules of what you can or cannot do in the game of Go Fish.

"A game is an activity involving player decisions, seeking objectives within a "limiting context" [i.e. rules]." Clark C. Abt wrote this about **games being involved with decision making and players seeking the objective with limited context or rules.** The game of Go Fish is based on player decisions. Making the decision of which card to call out and which player to choose from, it makes the game intriguing.

In Go Fish there is very limiting context. There are a set of rules you must follow to complete the objective of the game. For example, you can't ask someone for a 5 card if you don't have that card. **There are rules set in place so you can have an idea of what you can or cannot do.** You have no idea which players have which cards and it makes it hard to make decisions. Go Fish is a fun activity that **all players must make decisions each turn**. The context is limiting, what players can do is to listen to what their

opponents need for cards. If they are asking for cards that you have, when it is their turn, ask for the card that they wanted previously. In Go Fish there is "limiting context" within the rules so there is something to think about when making your move.

"A game is a 'voluntary effort to overcome unnecessary obstacles.'" Bernard Suits stated this definition. **When playing games, you need to overcome obstacles along the way.** For example, you must get more books than your opponents to be victorious. You must do your best to avoid what your opponent's do that will affect your progress to win.

In Go Fish, you can't take everyone's cards in one turn. Everyone goes around asking other players if they have a specific card, and if they don't the opposing player says, "Go Fish" and you must pick up a card from the pile. **If one of your opponents ask for a card that you have, it makes it harder for you to win the game. Which makes you want to put in the extra effort to win. The obstacles you face in Go Fish will want you to use strategy and think of ways to make your chances of winning easier.** That is what makes Go Fish more challenging.

The rules impact the game. The rules are connected because they get in the way to make the game easy to win and **they are the unnecessary obstacles in the game**. The rules cause an inefficiency on purpose. Go Fish is voluntary, it has goals and rules as well. The effort is how you want to spend your moves, you want to make the best out of your turn. **You need a strategy when it's your turn, which creates effort.** Depending on your opponent's moves, you must change your strategy based on what they do. The game is voluntary, and the rules are the unnecessary obstacles. The rules make the game fun, challenging, to make you put in more effort and playable. The unnecessary obstacles are the rules because they make the game harder and not easy to win.

As you can see, Go Fish relates with the definitions about games.

There is much more to Go Fish than most people think. There are several parts to Go Fish that make it more complicated and intriguing. Go Fish has many elements to it that many games don't. There are several rules to Go Fish. There are objectives and unnecessary obstacles must face when playing Go Fish. There are outcomes that could change the game. To play Go Fish you need to make decisions to keep the game going and give the game an ending or means. Overall, the game of Go Fish, is more fascinating than you think. There are rules, objectives, outcomes, and obstacles that consider Go Fish a game.

21. A Game to the Core: A "Go Fish" Analysis

LYDIA FINCH

The name of the game I am going to analyze is "Go Fish." This game was designed for anyone who wants to pass time with a simple card game. It also requires two or more players. The people playing this game would likely need to be able to read numbers. Because of this, the typical "Go Fish" game suggests players should be at least four years old. A regular, or "standard," deck of playing cards is used. "Go Fish" is a turn-based game, meaning each player has a set turn. The typical turn starts with the player asking any other player if they have a specific card. The player who asks for a card must also have this card before asking. If that player does have the card, then they will hand it over and the original player gets to ask for more cards (either to that same person or a different person). This goes on until the other player does not have the card the original player asks for. Once this happens the player is told to "Go Fish" and they need to draw a card from the deck.

Clark C. Abt's definition of a game reads "A game is an activity involving player decisions, seeking objectives within a 'limiting context'" In "Go Fish," the "player decisions" would be the players figuring out who to ask for a card or who to listen to. They also need to decide what cards in their hand they should ask for and when they should ask for each of them. The "objectives" that the players seek would be obtain a book of cards with the same rank from the cards they own. For example, if the player has four cards with a three on them, then they would want to put them together on the table. They'd want to obtain books of all their cards before the other players manage to. The "limiting context" that Abt's definition refers to is simply what we know as rules. In "Go Fish" there are a

small handful of simple rules. Each player is dealt five cards unless there are only two players, in which case they are dealt seven. The rest of the deck is placed in the center so everyone can draw from it. The player who's turn it is will ask for cards from other players only if they themselves have that card. Players can't look at other player's cards. When a player gets a book (four cards of the same rank), they must put those cards face up in front of them. When a player gets rid of all their cards in books, they win. "Go Fish" contains every element of this particular definition.

The definition from the book Rules of Play by Katie Salen and Eric Zimmerman reads as follows. Games are a "system in which players engage in an artificial conflict, defined by rules, that results in a quantifiable outcome." At first glance, it seems as though "Go Fish" doesn't fall into this definition, but English is an interesting language, making it so "Go Fish" actually fits perfectly into this definition. The game itself, governed by rules and turns, is a "system." The "artificial conflict" in this case would be the competition between the players to get the most books. Even simpler, the "conflict" could simply be trying to empty one's hand as fast as possible while still following the rules. Speaking of which, "Go Fish" is defined by a small handful of rules. Each player is dealt five cards if there are more than two players and seven if there are only two players. Players aren't allowed to look at other's cards or lie about the cards they have. They must ask another player for cards on their turn only if they themselves have said card number. When a player gets a book, they must put it on the table, face-up, in front of them. A player must "go fish" (draw a card from the deck) when the other player doesn't have the same card they do. A player wins when their hand is empty and all their cards are in books. The quantifiable outcome would be the cards that the player puts in books. There is a quantifiable amount of cards they can do this with.

Games are a "form of art" in which players – the people participating in the game – "make decisions in order to manage resources through game tokens in the pursuit of a goal." This is Greg

Costikyan's definition. This is a seemingly simple definition that applies to any game we can think of, however, we see that the definition is more complicated when we give it a closer look. In order to figure out if a game is a "form of art," we first need to define what art is. "Art" is something that can take many forms – physical or non-physical – in which the purpose is to provoke an emotion from the viewer. This emotion can be negative, positive, or neutral. Typically, I say that the artist needed to have the intention that what they were making was art, however, we will rarely ever know if the author of a game intended for it to be art. Therefore, I'm disregarding this part. If we use this definition of art, then "Go Fish", is, in fact, a "form of art" because one of the game's purposes is to elicit joy or fun from it. The next part of this definition is "decision making." There are many decisions to be made by players whilst playing this game. The players must decide what cards to ask for, when to ask for them, and whether or not they will listen to the other players asking for numbers (this delves into strategy relating to this game). Game tokens are something that represents the player's status within a given game. The "game tokens" in the case of "Go Fish" are the cards the player owns. This includes cards in the player's hand as well as any cards they put together to create books. These things are considered game tokens because they indicate how well the player is doing in the game. The "resources" would be the shared deck of cards in the middle and other players' cards. There are three goals of "Go Fish." The first goal is to create books (groups of four) of cards with the same number. The second goal is to clear your hand of cards by doing the former. The third goal is to meet goals one and two before the other players do. Knowing this, we can see that "Go Fish" is a game under Costikyan's definition.

Going through all of these definitions and comparing them with "Go Fish" has made me realize that "Go Fish" will probably hold up against any viable definition of a game. No matter how one looks at it, "Go Fish" will always be considered a game. I feel as though one

would be hard-pressed to prove it's not a game. This has also made me wonder how many games can be considered games when being compared to a definition. Going off of this, is a game still a game if it doesn't completely fit the definition?

22. Getting A Better Understanding of War

JACOB COSTELLO

The game of War is a game for people of all ages to play. The game is very simple, you use a deck of cards, half of the deck goes to one player and the other half to the opponent. Both players flip cards at the same time and the person with the highest card takes their own and the opponent's card. In this analysis, I will discuss what the game of War is about. I will talk about how to play the game, what you need to use to play, who the audience is, and the elements of a game that Costikyan states in his article. I will express the weakness of War and how it fits in to Costikyan's six game elements.

The name of the game is War. The intended audience for the game can be anyone. War is simple and it is very easy to catch onto it. The game can be designed for kids, older children or adults. War is a fun game for people of all ages. The materials needed for War is a full deck of cards, minus the jokers. The dealer must hand out cards to both players so one player has half the deck and the other player has another half of the deck. There aren't any materials you need besides that.

When it comes to the game of War, it is considered turn based. It goes by rounds, both players flip over their top card to see who wins in the round, the winner of the round gets both cards that were flipped over, then the players go again, and it repeats. Players keep flipping over their cards until their pile is empty, after that with the cards the player has one, uses that pile to continue playing. War is an easy game, Aces are the highest card and beats everything, King is the second highest and beats everything besides and Ace and so on. **The number card two doesn't beat anything unless there is a**

War. A War is when both players flip over the same card. When that happens, both players place down three cards, face down and flip over the fourth card, whoever has the higher card, gets all the cards that were placed down. Once one of the players run out of cards to play and doesn't have any more cards that they won from previous rounds, loses and the other player wins.

In 1994, Greg Costikyan wrote an article called, "I Have No Words and I Must Design." In this article he explains his six elements of a game, **those six elements are; tokens, goal(s), opposition, decision-making, information and managing resources.** Costikyan said an activity must have six elements in order to be considered a game, if the six elements aren't present, the activity is not a game.

According to Costikyan, these are what the six elements mean. **Game tokens are something that represents a player and the player's status within the game.** For example, in Monopoly, the fake money represents how wealthy or poor someone is. **A goal is something the player is striving for.** In Monopoly, the goal is to be the last player with money or to bankrupt all the players. **The opposition is something that gets in the way of the player reaching their goal.** The opposition is also the struggle in the game, the obstacle you must overcome to be victorious and achieve the goal.

There are many important factors in War, but **decision-making is the most important characteristic of a game. A player must be presented with a series of choices which will impact their chances of reaching their goal. In order to make these decisions, you need to have some sort of information.** In Chess, the player is presented with perfect information, which means there is nothing hidden from the opponent. Lastly, managing resources. **A resource is something a player uses in order to achieve the game.** In Monopoly, the resource is the spaces you land on. Using information, the player can know whether the space has been purchased or not, so they can use money to purchase it. Tokens,

goal(s), opposition, decision-making, information and managing resources are the six elements an activity must have to be considered a game.

According to Costikyan's definition, the single major weakness of War is decision-making. The definition states, "Games are a "form of art in which the participants, termed Players, make decisions in order to manage resources through game tokens in the pursuit of a goal." There is no decision making because you flip over the cards in your pile face down, so you are blind to what cards you contain.

The major weakness of decision-making in War affects other elements. The definition says players make decisions in order to manage their resources through game tokens. **In War, the game does not have decision-making, you flip cards at random to see who has the higher card.**

You can't manage your resources because again, you are flipping cards at random, you have no idea what card you will flip next. You have no control over the card your opponent flips over as well. Lastly, **War does contain game tokens.** The number of cards you have in your pile determines how well or how badly you are doing in the game; **it represents your status**. There is a goal to War. **You need all the cards in the deck to win.** Once your opponent runs out of cards, the game is over. The opposition is your opponent, **War is a head-to-head, one player against the other.** There is no information in War. You don't know what cards you have nor what cards your opponent has either. Without any information, there are no decisions to be made in the first place. Decision-making is the single major weakness in the game of War. It also affects the elements of managing resources and game tokens.

As you can see, War is a game for all ages, and it is very simple. The rules are easy to understand, and you can play with anyone. In the War analysis, I discussed the rules of War, the materials needed and how War is considered rule based. I analyzed Costikyan's six

elements that must be included for an activity to be considered a game. If an activity doesn't include tokens, goal(s), opposition, decision-making, information and managing resources, it is not a game. War's major weakness is decision-making, and it also affects the elements of managing resources and game tokens. According to Costikyan, War is not a game. Overall, War is a fun activity to play with anyone and there are interesting factors that make it fun and enjoyable.

23. What is War Good For? Still Nothing, Apparently

LYDIA FINCH

Card games are a fantastic way to pass the time, and socialize. But, what if I told you that not all card games are actually games? Definitions of games come in many shapes and sizes, each one changing what it means to be a game. Greg Costikyan's definition includes many pieces, all vitally important to the system of a game. I'll be using his definition to analyze the card game known as "War."

"War" is a card game played with a single, "standard" deck of cards. The amount of players in "War" is only two. The typical recommended age is three or above. Players only need to know how to count or tell what number is higher than another. The deck is split in two equal halves and given to each player. The game is played in rounds rather than turns. Both players simultaneously flip a card up from their respective decks. Whichever player has the higher value card gets the other player's card. These cards are then put on the bottom of that players deck. If the cards are the same value or rank, then it is "war." When this happens, both players put a card face down and a card face up. If it is war again, they repeat this until one of them has a higher ranking card. Whichever player has the higher ranking card gets to take all of the cards from the war and put them on the bottom of their deck. The game repeats like this until one player has all the cards and then they win.

The Costikyan definition of a game revolves around six main elements. These elements are game tokens, goals, opposition/ struggle, decision-making, managing resources, and information. Every game requires these elements. If it is missing one, then it's not considered a game. I will delve deeper into what these elements mean alone and in terms of the card game "War."

Game tokens are any entity that a player can manipulate directly. These entities represent how well the player is doing in-game. Depending on the game, the game token could be the player's cards, their character, or something that represents the player. In "War," this would be the player's deck of cards. This is because the deck represents how the player is doing in the game. If the deck is bigger, then they're winning. If it's smaller, then they're losing. Resources are things you can use or collect in-game. They help you accomplish your goals. In "War," your resources are all the cards in the game. In Costikyan's definition, you have to be able to manage these resources using your game token(s). While the players "use" their respective deck halves, there's no decisions to be made about the resources. Everything is straight-forward and there is only one thing you can do with your resources. Since there's no choices to be made about the resources, there's no managing of them. This is one of the downfalls in "War."

Information is an interesting thing in "War" because both players constantly know the same information. Each player knows all the rules, types of cards, and the value of the card both players draw in any given round. There is hidden information (not knowing the next card in each deck), but the game is hiding it from the players rather than the players hiding it from each other.

Goals are something the player strives toward in a game. In "War", the only goal is to collect all of the cards in the entire original deck. The opposition to this goal isn't the other player, as one would suspect in a two-player game, but rather it's the game's system or chance. I say this because the players aren't making any decisions or particularly controlling much of what goes on in the game. Neither player has control over whether anyone wins or loses.

This brings me to the most important aspect of "War:" it had no decision making. The game completely lacks the ability to provide this. The only thing the player does is flip a card over and give it away or take their opponent's card depending on the value.

According to Costikyan, this would be the biggest weakness in this "game." Due to the fact that there's no decision making, managing resources is extremely simple. Goals feel meaningless because everything is left to chance. There's no strategy involved whatsoever, as players need to be able to decide things in order to make a strategy. The "game" eventually stops being fun due to this lack of decision-making.

"War" was an extremely fun game when I was a little kid. I used to play it for hours on end while sitting in bed with my siblings. Now that I'm older, I really can't see myself playing it for more than ten minutes, much less over an hour. I would have wondered about why but after analyzing it I realize it's because of the lack of decision-making. Knowing this, I have to wonder how many successful children's games don't have a decision making element in them.

24. Important Foundations

LYDIA FINCH

Everything is easier when there's a template or foundation to work upon. The same goes for game design. As we've seen in other chapters of this text, designing a game is hard work that requires insight into many areas. The MDA theory, which can be found here, gives us a good foundation in order to work on games. MDA stands for mechanics, dynamic, and aesthetics. Breaking down games in this way makes it easier to build and analyze them. The reason it's easier is due to the fact that this method attacks designing a game from both the player and designer side.

So what do I mean by mechanics, dynamics, and aesthetics? Mechanics are the rules of the system that make the game work; dynamics are the complex outcomes one gets from the system rules interacting with the player; and aesthetics are how the player experiences the game (whether or not a game is considered fun). Each of these pieces are interconnected. This is something that game designers need to heavily consider when working on a game. The way the player experiences the game is affected by the dynamics of the game and the dynamics of the game are affected by the base system rules. Designers need to create a specific experience for the player (aesthetics) by working from the opposite end (mechanics).

In order to understand how a game you design works, you need to understand it from every level. MDA is a way to help understand how these are all connected. It's a way to try and overcome not knowing how the player will experience your game. Remember when I said designers work from the mechanic side to achieve the aesthetic? That makes game design extremely tricky because we don't know how the player will experience the game. That's

why play-testing is such an important part of game design. That's why MDA is so important.

How do you use MDA? Is it just a matter of understanding what it is? Yes and no. Understanding what MDA is can be helpful, but it's not the end all for game design. We can use MDA as a tool. Each part of MDA can be used as a lens to look at a game through. For example, if we look at a game through a mechanic lens, we're going to see a very different game than if we looked at it through an aesthetic lens. Designers need to consider all of these lenses from both the player and designer end.

What do these lenses mean? Aesthetics, as mentioned before, is how the player experiences the game. Is the game fun? How do we know the game is fun? In order to answer this, we stop using the word "fun" and use more descriptive words that help us break down the term "aesthetics." Aesthetics can mean sensation (sense-pleasure), fantasy (make-believe), narrative (drama, story), challenges (obstacles), fellowship (social interaction), discovery (uncharted territory, exploring), expression (self-discovery), or submission (a way to pass the time). A game is "fun" if it has one, or a combination of these things. Most games have a combination of them. For example, Final Fantasy has fantasy, narrative (story, drama, etc.), expression (self-discovery with the characters and perhaps the player), discovery (the player explores a vast world that is unknown at the beginning), challenge (there are many obstacles such as monsters), and submission (the game can be played just to pass the time. It also has "mindless" mini games within the larger games). In order for the player to enjoy the game, it's also important that it has a clear win condition.

Dynamics are how the rules interact with the player. Dynamics will heavily determine the player's experience, therefore you want the dynamics to be as concrete as possible. Keep in mind, there will always be unforeseen dynamics. One of the ways you can understand the dynamics is feedback systems. They can tell us

where to change something in a game's design. For example, in the board game Monopoly you have a feedback system that repeats like this: Role the dice, land on your own property or someone else's, pay money or get money (win/ lose), roll the dice again. Looking at the game as a feedback system shows us how easy it is for a poor Monopoly player to stay poor.

Mechanics are the rules and systems that make a game work. Mechanics will create dynamics when interacting with a player. For example, in a card game you might have the mechanics of shuffling, trick-taking, and betting. This will create the dynamic of bluffing. Adjusting mechanics will inevitably change the game dynamics. This is a big reason why play testing and fine tuning a game are so important.

As we can see, changing one small aspect of a game's design can change the entire game. This is why MDA is so important and useful. By looking through each lens we can see, in depth, what are decisions for the game can do. Structured, formal guides aren't needed for everything, but it certainly helps for game design.