100 PRINCIPLES OF GAME DESIGN

WENDY DESPAIN

KEYVAN ACOSTA

LIZ CANACARI-ROSE

MICHAEL DENEEN

ZACH HIWILLER

JEFF HOWARD

CHRISTINA KADINGER

CHRIS KEELING

CASEY KUCZIK

NICOLE LAZZARO

TOM LONG

MICHAEL LUCAS

DAVE MARK

DOUGLAS OBERNDORF

PATRICIA PIZER

MICHAEL PYNN

BRIAN STABILE

JASON VANDENBERGHE

100 PRINCIPLES OF GAME DESIGN

WENDY DESPAIN, EDITOR

KEYVAN ACOSTA
LIZ CANACARI-ROSE
MICHAEL DENEEN
ZACH HIWILLER
JEFF HOWARD
CHRISTINA KADINGER
CHRIS KEELING
CASEY KUCZIK
NICOLE LAZZARO
TOM LONG
MICHAEL LUCAS
DAVE MARK
DOUGLAS OBERNDORF
PATRICIA PIZER
MICHAEL PYNN
BRIAN STABILE
JASON VANDENBERGHE

100 PRINCIPLES OF GAME DESIGN
Wendy Despain, Editor

New Riders
www.newriders.com

To report errors, please send a note to errata@peachpit.com

New Riders is an imprint of Peachpit, a division of Pearson Education.

Senior Editor: Karyn Johnson
Copy Editor: Rebecca Rider
Production Editor: David Van Ness
Proofreader: Scout Festa
Illustrator: Ray Yuen
Composition: Danielle Foster
Indexer: Valerie Perry
Cover design: Aren Straiger
Interior design: Claudia Smelser
Cover photo: www.studiodespain.com
Contributors: Wendy Despain, Keyvan Acosta, Liz Canacari-Rose, Michael Deneen, Zach Hiwiller, Jeff Howard, Christina Kadinger, Chris Keeling, Casey Kuczik, Nicole Lazzaro, Tom Long, Michael Lucas, Dave Mark, Douglas Oberndorf, Patricia Pizer, Michael Pynn, Brian Stabile, Jason VandenBerghe

NOTICE OF LIABILITY

The information in this book is distributed on an “As Is” basis, without warranty. While every precaution has been taken in the preparation of the book, neither the author nor Peachpit shall have any liability to any person or entity with respect to any loss or damage caused or alleged to be caused directly or indirectly by the instructions contained in this book or by the computer software and hardware products described in it.

TRADEMARKS

Many of the designations used by manufacturers and sellers to distinguish their products are claimed as trademarks. Where those designations appear in this book, and Peachpit was aware of a trademark claim, the designations appear as requested by the owner of the trademark. All other product names and services identified throughout this book are used in editorial fashion only and for the benefit of such companies with no intention of infringement of the trademark. No such use, or the use of any trade name, is intended to convey endorsement or other affiliation with this book.

ISBN-13: 978-0-321-90249-8
ISBN-10: 0-321-90249-1

9 8 7 6 5 4 3 2 1

Acknowledgments

A ton of thanks need to be heaped into three piles, so in no particular order, I would like to acknowledge that our "inspiration" for this book is another book used the world over by great game designers; it is called *Universal Principles of Design*, by William Lidwell, Kritina Holden, and Jill Butler. So many great game designers I know count this as a valuable part of their professional library.

However, when we started introducing this book to students in the Game Design program at Full Sail University, we discovered that this classic was in some ways too advanced for them. Their inexperience made it too hard for them to make the leap from architecture or art to game design. We initially tried to write a book that bridged the gap between these wider, classic universal principles and the process of game design. But we soon found that we wanted to add a few game-design-specific principles here and there. And then we found that "a few" had turned into 85 or so, and there was no end in sight. So now there are a few principles in this book that are also covered in *Universal Principles of Design*, but not very many. In my opinion, these two books are great companions on the bookshelf of any game designer.

And the next acknowledgment goes to the person who deserves undying thanks and delicious home-made baked goods every holiday for the foreseeable future. Ray Yuen, the illustrator, who I swear has a collection of gnomes under his back porch sketching adorable, funny characters doing wacky things like bringing a gun to a knife fight, did great things for this book in an insanely short amount of time and did an admirable job of adding zombies. Thanks, Ray. I hope you share the cookies with the gnomes.

And third, but not last, is the long list of contributors who went off this cliff with me. Thanks so much, everybody—you know what for. Specifically, I think Keyvan Acosta gets credit for the original idea for this book, Chris Keeling gets credit for being a temporary figurehead, and Ricardo Aguilo gets credit for being with us in spirit but having finally learned how to say "no" when asked to volunteer for something. I'd better stop now before I get any more teary-eyed about all the great contributors (and the amazingly wonderful editor at New Riders, Karyn Johnson, who tells me I have no more room on the page). So I'll just say I love every one of you, and your baked goods are in the oven right now.

Contents

3 Universal Principles for Game Balancing

The 100 Principles

An alphabetical listing

Introduction

This book is a toolbox of possibilities. It is not a how-to manual. It contains at least four principles (I'm not telling which ones) that insist or imply there is only one true way to begin designing a game, and if it is started in any other way disaster will ensue. They cannot all be right . . . can they?

Well, I'm sure I don't know, and I'm not going to try to convince you one way or the other either. What I do know is that these ways/principles/philosophies are all coexisting in the current game industry. Different companies, rock-star designers, and schools of thought all use them and swear by them. Maybe there's a Master's thesis in there somewhere, but I'm not interested in digging it out and ranking the schools of thought according to some value of success.

I'm a collector, not a competitor. I go through life picking up ideas and adding them to my mental list of "Hey, that's interesting, I might use it someday." And when I stumbled into game design as a profession, I discovered every game designer does this same thing. They have a mental toolbox they have collected over the years, which they bring to bear on whatever problem faces them.

And this is one of the reasons it is so hard to teach game design. The tools of the trade are vast and strange. This book is a download of my own mental toolbox with additions from the collections of my professional colleagues. I find it liberating and exciting to have it out in front of me instead of floating around in a jumble in my grey matter. It's even organized into the four times I find myself reaching for these tools: when I'm trying to innovate, when I'm hacking out the "cruft" in the middle of game creation, when the nearly finished work has to be balanced, and ultimately whenever I have to troubleshoot a specific problem.

How This Book Is Organized

Did I just make a book that needs its own instruction manual? I think I did. I'm not sure if this is a good thing or not! It's certainly very meta. The fact is, this isn't like any other game design book out there, so maybe you do need a bit of help text up front to get you oriented (see Advance Organizers).

This book is riddled with phrases highlighted in orange type (like the one in the previous sentence) that reference other principles of game design. If they look a bit like web links, it's because that's what I wish they were. I wish you could poke them with your finger and have the pages turn to show you that there's a whole section right over there expanding on this idea. Maybe the digital version will work that way one day.

At any rate, they're kind of like footnotes, but I'm not a footnote kind of person. I'm a digital kind of person, so when some part of this book skims over, or touches on, or mentions in passing an idea that is explored in more depth somewhere else in this book, you'll find the principle cross-referenced in orange. Blue type is used to highlight the name of the designer who created or popularized the principle.

So let's look at the parts of this book. As mentioned earlier, these core principles of game design are organized by four themes: innovation, creation, balancing, and troubleshooting. Each page describes a different fundamental principle that may or may not come up in the process of designing games. If you open this book randomly in the middle, you'll see this: On one side is a text explanation of a principle, and on the facing page is an image that helps illuminate or illustrate the ideas. Go ahead and try it. I'll wait.

No, really. I'm not going anywhere. Take a look; then come back here when you're ready.

. . .

Welcome back! I hope you were intrigued by what you found. You now see how the book is set up, and you've also now used it in one of its intended ways.

How to Use This Book

Don't get too hung up on why a principle is categorized into one section rather than another, though (see Hick's Law). They can all be used at any point in game development. The categories are just there to help bring order to the chaos and to guide you in the right direction when you're feeling a little lost.

Here are just a few ways you can use this book:

- **Looking for random inspiration.** Different people learn better in different ways (see Gardner's Multiple Intelligences). Opening this book to a random page is a great way to kickstart a stalled brainstorming session.
- **Brushing up on the fiddly bits.** Some of these principles are convoluted, crazy ideas with lots of parts. Use this book as a reference when you just can't remember what the Fourth Key to Fun is, for instance.
- **Learning something new.** This book is the collective unconscious of many people. Even the contributors themselves were eager to read up on the sections they felt they weren't qualified to write. There's a lot of great information in here.
- **Running diagnostics.** When something in a game is just not working out, this book can suggest avenues to explore. Links between principles can help get you to the root of a problem.
- **Solving problems.** There's an entire appendix with a list of many ways to approach solving a problem. It's not a how-to manual for any particular problem, but it does suggest ways to get started.

Keep in mind that there is no way to completely cover any of these complex ideas in one two-page spread. Contributors to this book complained about how few words I allowed them to work with, and some of them blew past the restrictions altogether (see Griefing and House Rules—not as examples, but as explanations), and I had to cut out a lot of great stuff.

So think of each page of this book as an introduction to or quick summary of the topic it covers. There should be enough information, jargon, and name-dropping in each one that rudimentary Google-fu will get you falling down a deep, deep rabbit hole on any of these principles. In fact entire specific books are sometimes recommended in the text. At any rate, don't think of yourself as an expert on a topic once you've read the one-page description here.

And don't even begin to think that these are the only principles or even the most important ones. There are many we couldn't fit into this book, which are being catalogued at www.gamedesignprinciples.com, so come join the discussion there, and tell us which of your favorite principles we left out!

4

UNIVERSAL PRINCIPLES FOR TROUBLESHOOTING

Advance Organizers

An *advance organizer* is a technique from the field of education which helps students learn faster and better (see Koster's Theory of Fun) The idea is to provide a high-level overview or hook that prepares the student for what they will learn next. It can be designed to help them relate the new concepts to other things they already know and put everything in context. In the case of a player, it lets them know that new information is on the way or that a transition is about to take place. Advance organizers have several forms in video games.

The first advance organizer a player is likely to encounter is Hype. This type includes ads, previews, game press, conventions like E3, trailers, and other similar methods for advertising an upcoming game. The hype could take the form of an Alternate Reality game designed to heighten the player's awareness of an upcoming game. Each time a player encounters an advance organizer, they become aware that a new game is out or is about to be released and new game experiences or new knowledge is imminent. The hype helps players understand how the new game fits in a genre or franchise and provides general expectations for what they should focus on in the game (see Attention vs. Perception).

The next type of advance organizer players encounter is the loading screen. While the game is copying files onto the hard drive and is loading content, players are treated to a screen that introduces them to at least one character from the game or other art that prepares them for the upcoming content, thus setting the tone for the content. The loading screen will also likely have some kind of counter or time indicator to show that something is going on (a spinning ball, a timepiece, etc.) while the game is loading and doesn't appear to be doing anything. All of these cues inform the player that something new and exciting is about to happen, and that they should pay attention and get ready.

When designing the loading screen, make sure it has a consistent art style with the rest of the game. With this advance organizer, the designer emphatically sets a tone for how the game will proceed (see Cognitive Biases).

Frequently, games start with a cutscene to provide a backstory for the player and to provide context for the start of play. The cutscene may be long or quite short but will certainly inform the player that something is about to happen and set up expectations and suspense. Cutscenes can be used throughout a game to inform the players of transitional content or a scene change where they will encounter new content and, hence, new knowledge. The players have been primed by each of these advance organizers to succeed at new challenges.

Many games present the players with a menu to start them out. The menu is another advance organizer that the player knows will take them to a different venue or part of the game, depending on which option they choose. For instance, letting players choose a destination from a clearly laid-out map as a menu sets them up to be ready when they enter a new environment, rather than just a boring list of place names, which provides little information and can result in serious disorientation rather than excitement.

Finally, be careful of using advance organizers in places where new content is not going to be introduced, since this is their role. Games that use excessively long loading screens can frustrate players who are expecting something new but don't get it; this can be the sign of an asset loading/storage problem that a designer can overcome with some clever coding.

Affordance Cues

Remember as a kid pushing all the buttons in the elevator? Finger-sized circles, beveled edges that stick out just a bit from the wall plate, lights that come on . . . those buttons just cried out to be pushed. Their physical qualities communicated how to interact with them without requiring a tutorial or instruction manual. Psychologists refer to these qualities as *affordance cues,* and they are not just useful in elevators. Horizontal bars on doors imply they should be pushed; a vertical handle implies it should be pulled. Affordance works in a virtual environment with just as much power.

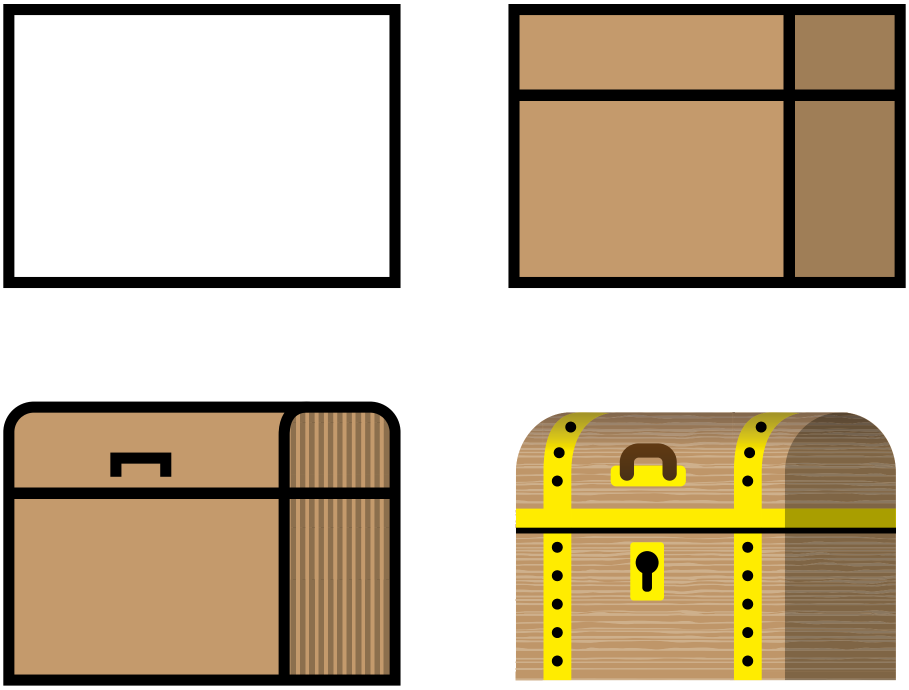
The original concept of affordance was introduced in the 1970s and was used primarily in psychology. It was openly adapted to the design process in works like Donald Norman's *The Design of Everyday Things.* Human-Computer Interaction (HCI) and interface design took affordance one step further by making intent central to the principle. Designers need to be aware of the intentional and unintentional affordance cues their products demonstrate and make an effort to use affordance to encourage the intended use—if a button is meant to be pushed, do not use affordance cues that encourage pulling.

This concept is at the root of interface design. When a web interface or game heads-up display (HUD) is filled with buttons that have beveled edges to imply depth, this is affordance at work. Many designers go one step further and try to make their buttons seem edible or tasty, appealing to not only the visual and the sensory (touch) but also the oral. Every sense that is appealed to increases the player's desire to act on the object in question, thus increasing its affordance.

Affordance is key in user interface and user experience (UI/UX) design (see Krug's First Law of Usability). Those HUD elements and interface elements must cue the player into how to interact with them in the game. If the element is a dial, players will expect that they should use their mouse or fingertip to gesture in a circular fashion while interacting with the object. An example of this occurs in a horse riding game called My Horse in which players must mix grain for their horses. This iPad game makes maximum use of affordance here by showing a circular indicator on the screen that the player must then follow with a fingertip to execute the move. Tablets open up new channels of affordance with the gestural systems they cater to naturally. Swiping, tapping, and so on are natural motions games can give easily interpreted cues for, thus optimizing affordance.

And in games, affordance goes even deeper than interface design. Environments and puzzles with appropriate affordance cues can make a game feel more fun and effortless. Affordance can cut down on time players spend in tutorials and draw them into appropriate behaviors without having to enforce those behaviors with rules.

When designing or troubleshooting the UI, it is especially important to keep affordance in mind. What action should the player perform? What path should players follow? Every element placed in the environment will influence the affordance of the player's possibility space. If the player needs to be guided from one location to another, put in a path or carpet between those points to make the path clear. With proper use and care, affordance can make a game eminently playable rather than a logistics disaster.



A blank white rectangle doesn't include enough affordance cues to prompt users to take any particular action. As more affordance is added—color, shading, handles—intended action becomes clearer. Instead of a confusing white box, players instinctively know they should try to open the treasure chest.

The Buster Principle

Too often in games, game makers punish players inadvertently by making the game too hard (see Punishment). Obviously, there are cases where making the game hard is in pursuit of *fiero,* or hard fun (see Lazzaro's Four Keys to Fun), but what about the player who is trying hard to play but lacks the motor skills to master the game's demands (see Errors Players Make)? Should this player walk away from the game, frustrated and angry at not being able to complete an experience they were previously enjoying tremendously? How many players fail to complete a game they love because of a boss battle they simply cannot win due to their fingers not moving fast enough? In these cases, even cheat guides are no help. If that player doesn't have a 10-year old nearby they can hand the controller to for a few minutes, they have no recourse but to quit in frustration or simply lose. Frustrated and angry is no way to leave a player feeling. Players should leave a game with a sense of satisfaction at having accomplished something (see Sense of Accomplishment).

The Buster Principle is simple: Be kind to your players. In the event that a player has clearly tried repeatedly to succeed at a task (something easily tracked in modern video games), try making the task just a little bit simpler. It may not even be necessary to change the task enough to be perceivable to the player. A small decrease in the difficulty may be the difference between an unhappy player throwing the controller across the room and an ecstatic player rejoicing and pumping their fists upward with a sense of accomplishment.

The Buster Principle was invented and demonstrated by a Goffin's Cockatoo named Buster Keaton. He played dexterity games with his owner, and kept a sharp eye on how frustrated the human participant was feeling during the whole endeavor. He knew that irritated, annoyed humans tended to walk away from a frustrating activity. The bird wanted the game to continue for as long as possible, however, so when sensing rising frustration in the owner, the bird would back off on the difficulty just a bit to make the human participant feel a sense of success. The adjustments were subtle enough that it took the owner some time and many games to realize the bird was going easy on her at just those moments when she was about to give up.

So the basic concept of the Buster Principle is to allow a game to automatically (and behind the scenes to the unaware player) tune the difficulty level of a particular skill to the ability of the player—or the frustration level of the player.

This is not in any way to suggest all games be made easy. It is simply recognizing that making games inordinately difficult isn't a good experience for players. Creating a good experience for the player should be the goal rather than developing a hard game to prove how clever and smart we are as game makers.

Early Infocom games were well known for how impossibly hard their puzzles were. So hard, in fact, that it led to an entire product line known as *Invisiclues*—books that included a marker with invisible ink the player used to answer a puzzle on numbered clues. The questions in the clues became increasingly direct, the first being obscure (see House Rules). Computer games have come a long way since then, and now that game makers can easily detect how much a player is working at solving a problem, better solutions exist to avoid excessive player vexation. In short, don't torture players. Throw them an easy one now and then to reward them for continuing to try.

Cognitive Biases

Players all bring their own psychological filters to games. These filters skew "what really happened" into "what I *felt* happened." Naturally, each person is an individual with their own filters; different people may perceive, understand, and react to stimuli in different ways. However, certain threads are common to most people. Each of these filters tends to operate in similar ways for all people, affecting their understanding of events and shaping the feelings they get as a result.

Collectively, these filters are referred to as *cognitive biases.* Although there are scores of different identified cognitive biases, some primary ones rear their heads in how players perceive games.

Perhaps one of the most cited is *confirmation bias.* Put simply, confirmation bias sounds a lot like "see! I told you!" People are disproportionately drawn to information that aligns with what they already believe—even if their belief is wrong. For example, when reading news stories, people are more attracted to those that state things they already feel to be true, and they tend to reject those that dispute their notions. In games, people are more likely to notice events that confirm their preconceived notions (positive or negative) about characters, places, or occurrences. Correspondingly, they are more likely to dismiss, ignore, or even not notice things that do not conform to their beliefs.

When searching for information to process a decision or answer a question, people often rely on an *availability heuristic.* Because they usually do not have mental access to *all* the information, they put a disproportionate weight on whatever comes to mind first. The belief is that it must be more important if you can remember it. In particular, emotionally charged events are more "available" to memory than mundane ones.

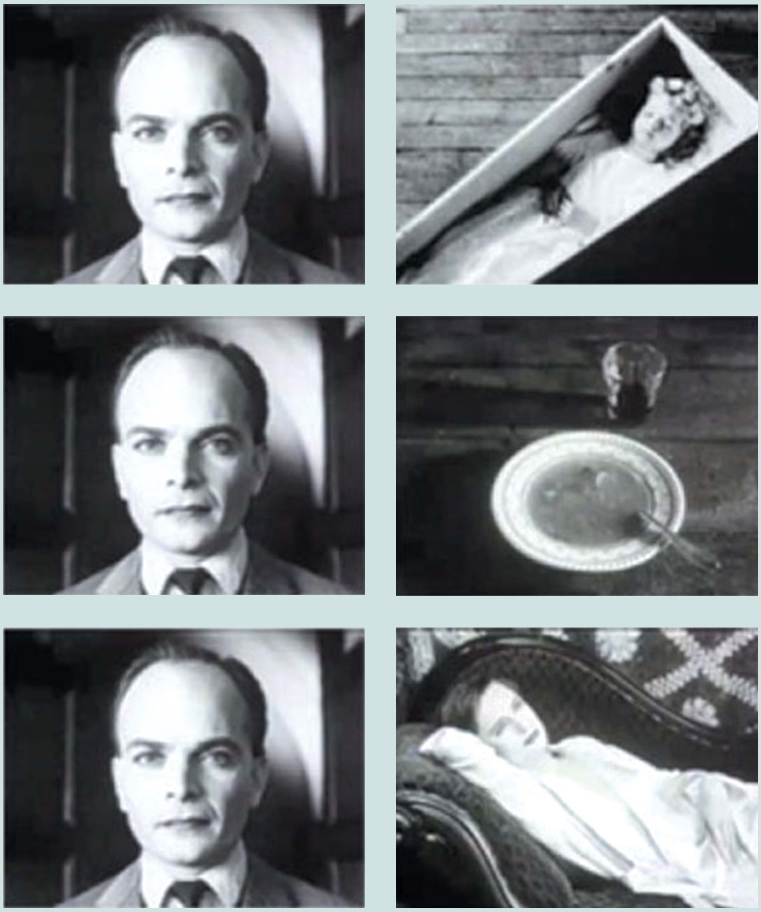
One type of availability heuristic is the *negativity bias.* Negative experiences are more emotionally charged, and thus they are more easily remembered. Because of this, people tend to put more weight on negative occurrences than positive ones. For example, players of popular word tile games (such as Scrabble or Words with Friends) might form false beliefs about the random distribution of tiles based on how often they draw difficult-to-use collections such as all vowels or all consonants. They notice those groupings more because they are frustrating compared to the (far more frequent) mixed vowel/consonant combinations.

Another type of availability heuristic is the *recency bias* where people put more emphasis on recent events. Because people have a hard time accumulating and analyzing statistics over long periods, they put more weight on things they can remember. More recent occurrences are not only easier to remember, but when combined with other forms of bias (e.g., confirmation or negativity biases), they can lead players to believe that some new trend is developing when, in actuality, they are only random events.

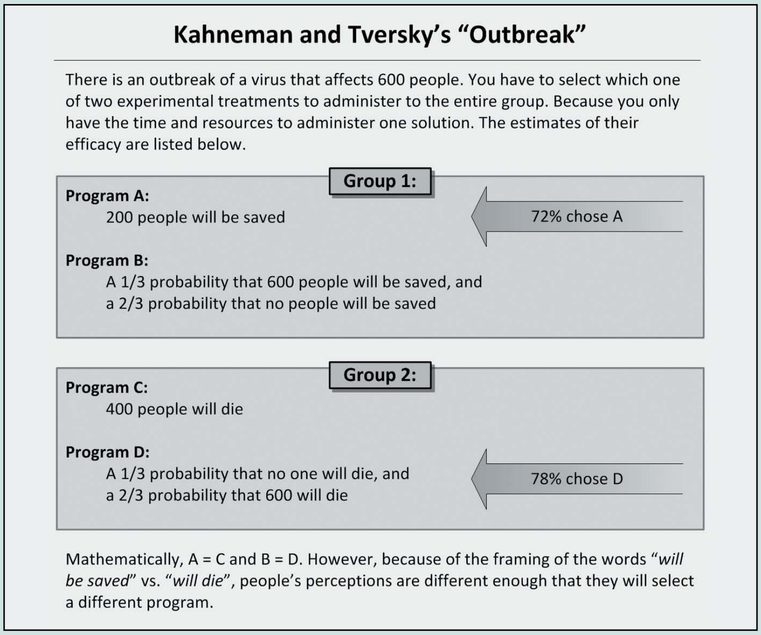
A typical type of recency bias is *anchoring.* When people are given a piece of information about something, they process all subsequent information relative to that initial information—or anchor. For example, if someone was shown an initial price of $80 and a reduced price of $50, they would view that lesser-priced item as a better value than if they had never been shown the initial price. That is, they tend to think, "it's $30 cheaper than it used to be," even when they know the initial price is arbitrary or irrelevant.

People are also very susceptible to having their perceptions of information changed by how the information is *framed* with different words or other delivery mechanisms. For example, when information is presented with positive wording, people infer very different conclusions than when the same information is presented with negative wording. In games, framing comes up in a variety of ways, such as through the user interface, the type of wording selected for the story or dialogue, and even the specific colors and sounds.

Another type of framing is shown in the *Kuleshov Effect.* Originally this term applied to film editing, but it can also be broadly applied to games. The original Kuleshov Effect experiment included a clip of an actor's expressionless face with one of three other images (a bowl of soup, a dead child, a beautiful woman). Despite the face being the same in all three examples, people thought the actor was emoting the appropriate feeling—that is, hunger, sadness, desire—based on which of the three subjects it was shown with. It illustrated that, when asked to make judgments about vague or indistinct information, people look to nearby information to infer what *should* be there, and then they believe that it really *is* there. In games, we can combine pieces of information to make the player believe that something is there that is not. For example, artificial intelligence (AI) characters with a small set of very simple behaviors suddenly take on the appearance of intelligent decision-making with the careful application of audio *barks*—short, often randomized dialogue chunks such as "Hey, is someone there?" or "Oh! I've been hit!"



Images from the film that demonstrated the Kuleshov Effect. Audiences reported that the actor's face showed a different expression when juxtaposed with each of the other images. In reality, the same footage of the actor was duplicated three times. Audiences saw what they expected to see.



Kahneman and Tversky's "Outbreak"

There is an outbreak of a virus that affects 600 people. You have to select which one of two experimental treatments to administer to the entire group. Because you only have the time and resources to administer one solution. The estimates of their efficacy are listed below.

Group 1:

Program A:
200 people will be saved

72% chose A

Program B:
A 1/3 probability that 600 people will be saved, and
a 2/3 probability that no people will be saved

Group 2:

Program C:
400 people will die

Program D:
A 1/3 probability that no one will die, and
a 2/3 probability that 600 will die

78% chose D

Mathematically, A = C and B = D. However, because of the framing of the words "*will be saved*" vs. "*will die*", people's perceptions are different enough that they will select a different program.

Dominant Strategy

Dominant Strategy refers to a frequently adopted, extremely successful approach to gameplay exploited by players to gain victory in a predictable fashion. A Dominant Strategy showing up in a game can be

- Accidental, due to a designer's inattention to balance
- Emergent, from player behavior or personality
- Deliberate, through a conscious design choice

Dominant Strategies are relevant to single- and multiplayer games. Consequently, it's the designer's responsibility to identify and troubleshoot all play strategies, especially those that negatively impact audience enjoyment.

At least a single Dominant Strategy exists in most games. Thus, you, the game designer, need to be aware of this principle and then decide whether to embrace or reject it in your design. As games seek to preserve an economy of Fairness, and Dominant Strategies threaten this, the solution comes down to balancing. If a game feels as if it has become routine, disinteresting, or too simple, maybe a Dominant Strategy imbalance is to blame.

Problems arise when a single strategy is so potent and so preferred that it leaves no variety to play. For example, if players have access to multiple configurations, yet they always use the same character class, weapon, or play method, then a game's design may feel uniform and stale. In this case, audiences may grow bored and disengaged. If it gets to a point where a single strategy hurts player interest and demotivates learning, engagement, and the Interest Curve, you must redesign or remove it. Doubling and Halving variables may be one of your first steps toward re-balancing strategies and play.

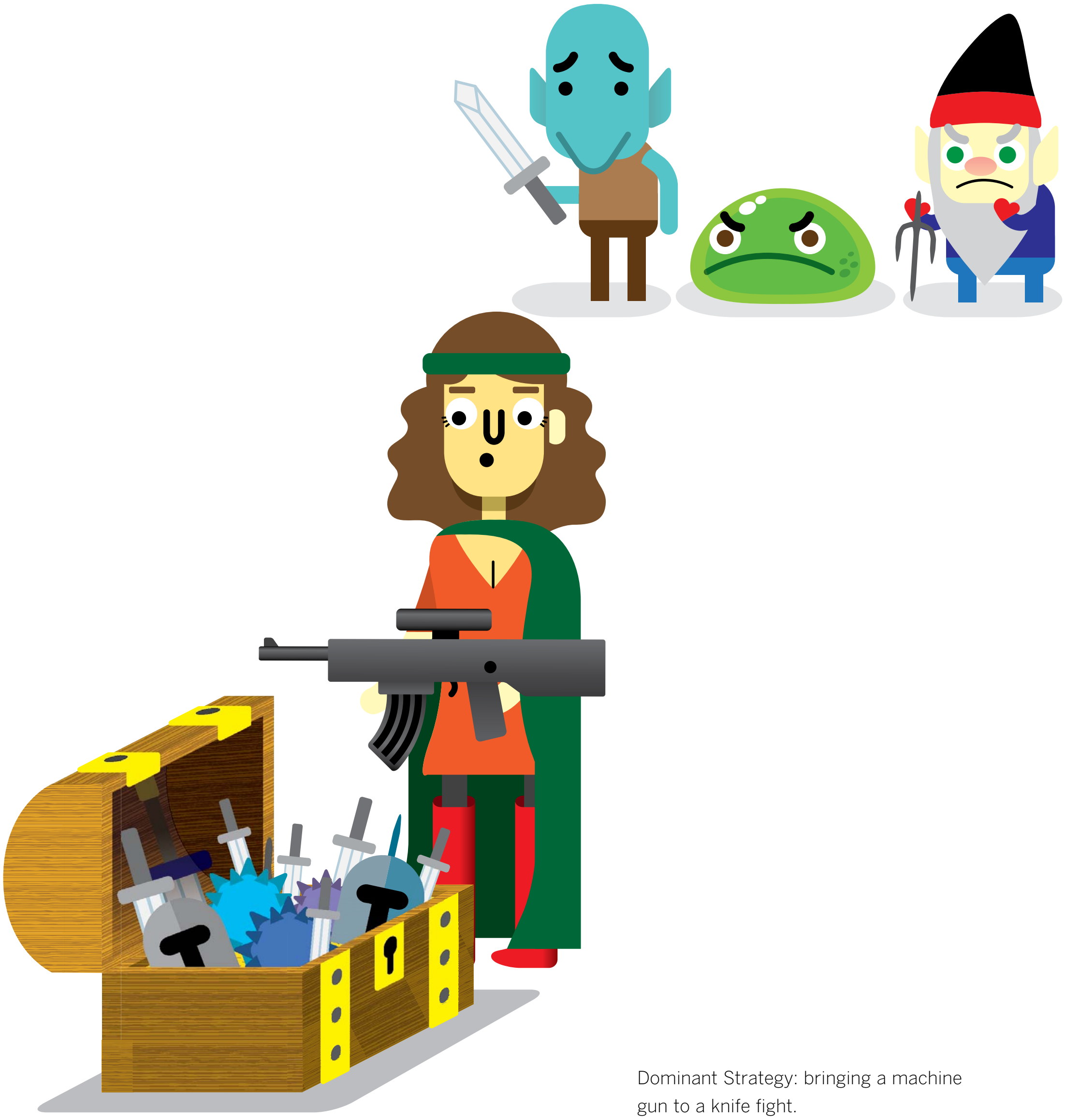
However, this isn't to say that Dominant Strategies can't benefit play experiences. Sometimes designers will nurture a Dominant Strategy, especially if it is only temporarily applicable or not too insurmountable. By providing a balanced Dominant Strategy, you can empower your players with sensations of intelligence, strength, and successfulness. A key to tuning Dominant Strategy is to understand and weigh how powerful the strategy is against existing rules and alternative strategies. So, yes, it's OK to have stretches of play where a Dominant Strategy exists. This can promote a temporary sense of mastery and feeling of omnipotence for your players; but, if this state persists for too long, players may quit because the game has failed to remain challenging.

To create a deliberate and well-designed Dominant Strategy, make sure that the best strategy does not too greatly disadvantage non-adopters of the Dominant Strategy. For example, a Dominant Strategy in competitive first-person shooters may be for players to prefer grenades and close range explosives. Often with these weapons, damage inflicted is high enough to kill at least one or two opponents, splash damage provides for unintended homicides, and lobbing grenades or placing explosives is relatively less skill-driven than precision gunplay. To balance against this strategy and create a counterstrategy, designers can provide measures that reduce the efficacy of explosives. These can involve mechanics tweaks including ways to detect explosives, armor against explosives, or long range weapons that are most effective beyond the distance of grenade throwing and blast zones.

Note that in the previous example, skill level can play a role in which Dominant Strategies are favored. For example, new players may prefer an explosive-based approach to combat while learning maps and other players' behaviors because explosives tend to be easily executed and less precise. Then, as the players gain familiarity, they may abandon the explosive-based strategy in favor of a different skill set that allows for greater precision. Maybe the experienced player's strategy becomes ranged weapons, like sniping. Remember, Dominant Strategies are subject to skill level, players' interaction with Learning Curves, and the strategies the players are competing against.

Despite the risk associated with allowing for Dominant Strategies, it is worth mentioning that certain games benefit from including them. The dominant M6D pistol in Halo: Combat Evolved was imbalanced, but it did not ruin either the single- or multiplayer experience. The Zerg Rush in StarCraft is an effective strategy that can be dominating, but it is not a certainty. In each of these games, a skilled player can circumvent the Dominant Strategy and gain a feeling of mastery and accomplishment, but it does take familiarity and practice. Perhaps you could argue that gameplay in these examples is "fun, but broken," but these are two high-profile titles that thrive with these strategies included regardless.

Another way to use a Dominant Strategy as a design advantage is to capitalize on the fact that the strategy centers on controlling a particular power, weapon, spawn, or specific advantageous location. This can create hot spots and bottlenecks, funneling players without using architecture or level design, but still bringing them together for interactions where conflict, reward, satisfaction, and fun can occur. Again, the key to handling a design that includes Dominant Strategies is in balancing the strategies against one another while leaving the opportunity for slight imbalance and the human elements of play.



Dominant Strategy: bringing a machine gun to a knife fight.

Fitts' Law

In Human-Computer Interface (HCI) studies, the tradeoff between speed and accuracy of aimed human muscle movement is referred to as *Fitts' Law.* In other words, the faster someone moves to point at a target, the less accurate the movement becomes. This can be predicted using a mathematical model. When a human uses a controller (such as a mouse), this tradeoff becomes important to game design. The speed and accuracy with which a player must interact may be set at impossible levels or at least very difficult ones. At its simplest, the larger a target is and the closer it is to the starting point, the more quickly and accurately a person can point to it.

Fitts paid particular interest to the task of pointing, something players do constantly with their cursors, long before the advent of personal computers. In his description of pointing, three parameters are of interest. The first is the time needed to point to the target. The second is the distance that must be moved from the starting position to the center of the target. Finally, the width of the target comes into play. Each of these three factors affects the ability of the player to accomplish the movement.

Though the model only considers the width of the target, it is obvious that the height must be taken into consideration as well. An object may be quite wide, but if it is only one pixel high (a thin line), it will still be harder to click than a more proportional object.

Now, though the obvious application here is in combat and targeting, the less obvious but perhaps more important application is in the user interface (UI). UI elements should be placed close to where the player is enacting pointing behaviors and should be large enough to be easily targeted and clicked when the user desires. If the UI is built such that the player is constantly having to move long distances to get to it, the user will become fatigued and frustrated. The principle is simple: The closer an object is, the easier it is to use.

Grouping objects of similar functionality in the interface will make it easier for the player to access them as it is easier to find similar objects. Also, having shorter distances between similar buttons will make them easier to use. Finally, making UI elements that need to be used often larger will make life easier for the player.

When it comes to combat and targeting moving objects, the designer must take human constraints into account. Of course, if the aim is to make the game difficult, the constraint becomes the fastest tester in the pool of people testing the game. Every test team has at least one super-human tester who can handily rise to the most difficult challenge.

7 LEVEL
ENERGY
STOP
CRUISIN'
GAS
BRAKE

Fitts' Law can be summed up as the idea that the faster someone moves to point at a target, the less accurate that movement becomes. So when the Core Gameplay Loop involves both speed and accuracy, clear and easy controls are essential.

Fundamental Attribution Error

Someone is driving down a busy yet swiftly moving highway. They have a meeting to attend and are running a little bit later than they would like. They see an old Cadillac getting closer and closer in front of them. “Come on, old lady!” they grumble through clenched teeth. They eventually pass the Cadillac; then, just after they reach a comfortable speed, they notice a sports car in their rear-view mirror. It is getting closer and closer to their rear bumper—dangerously close. They shift to another lane and the sports car speeds by at a seemingly breakneck speed. “Maniac!” they yell.

Of course, what they don’t seem to realize is that to the Cadillac, they were the maniac. And to the sports car, they were the old lady. When they are told this, they respond, “No! I was late for a meeting. That’s why I needed to pass that slow-poke Cadillac. But I’m not going to drive recklessly like that sports car!” They’ve assigned a situational reason to their behavior. Yet the other cars were driving the way they were because they were “reckless” drivers or “slow-pokes.”

In another example, a student fails a test. He complains: “It didn’t help that my job made me work overtime every day this week.” The instructor has a different view: “I wish my student wasn’t so lazy and would just make the time to study.” The student attributes his failure to a situation. The instructor attributes it to the characteristics of the student.

This is what psychologists call the *Fundamental Attribution Error.* It is the nearly universal willingness of someone to assign situational reasons for their own behavior while simultaneously treating the behaviors of others as a result of their character and not at all a result of their situation. Researchers have found that even when someone knows they are doing this, it is very hard for them to stop doing so.

This sounds like something that is only applicable in laboratory situations, but it has major implications for game design, as the following scenario shows:

A play tester sits to play a game and doesn’t have a good experience.

The play tester blames the system—certainly they won’t blame themselves! Maybe they didn’t understand how to play because they slept poorly last night, didn’t eat breakfast, or were preoccupied with a strained work or romantic relationship. Those reasons will not come up.

The designer will almost unerringly blame the play tester: “The instructions were in the Pause menu! How could they miss the shining tutorial icon?” To the designer, clearly the problem was that the play tester was dumb or bad. Or maybe the designer will blame the situation: “I hate that we have to play test with an unfinished build.” The designer’s explanation is to assign blame to the character of the play tester or to the situation of the game.

When assigning reasons for human actions, be very aware of the knee-jerk reactions that everyone makes to assign behaviors as coming from the nature of the person rather than external elements. Perhaps in the previous example the tutorial really isn’t obvious. Perhaps the game does need work. That’s what you play test for in the first place!

All human beings tend to have some trouble attributing blame appropriately when something goes wrong. If the failure is their own, they take into account mitigating factors such as lag or bad controllers. Blame is spread out. But when observing someone else's failure, humans tend to ignore mitigating factors and assign all the blame to personal shortcomings. When I miss an easy shot, it's because of lag. When you miss an easy shot, it's because you're a lousy shot. These fundamental attribution errors need to be taken into account especially when analyzing results from play testing experiments.

Golden Ratio

The *Golden Ratio,* or the Golden Mean, is a term from mathematics and the arts wherein the ratio of the sum of the quantities to the larger quantity is equal to the ratio of the larger quantity to the smaller one. It is expressed algebraically as follows:

$$\frac{a+b}{a} = \frac{a}{b} \equiv \varphi$$

The Greek letter Phi represents the golden ratio. Its value is 1.61803398875. This value is quite common in architecture and is found in nature constantly. Found by the Greeks, the Golden Ratio has been used as a guideline in proportion ever since. The common expression of this is the Golden Rectangle wherein the shorter side of the rectangle equals *a*, and the longer side equals *a* + *b*.

As early as 450 B.C., the statues of the Parthenon were made to the Golden Ratio, as were many other features of the structure, and buildings today are still built with these proportions in mind.

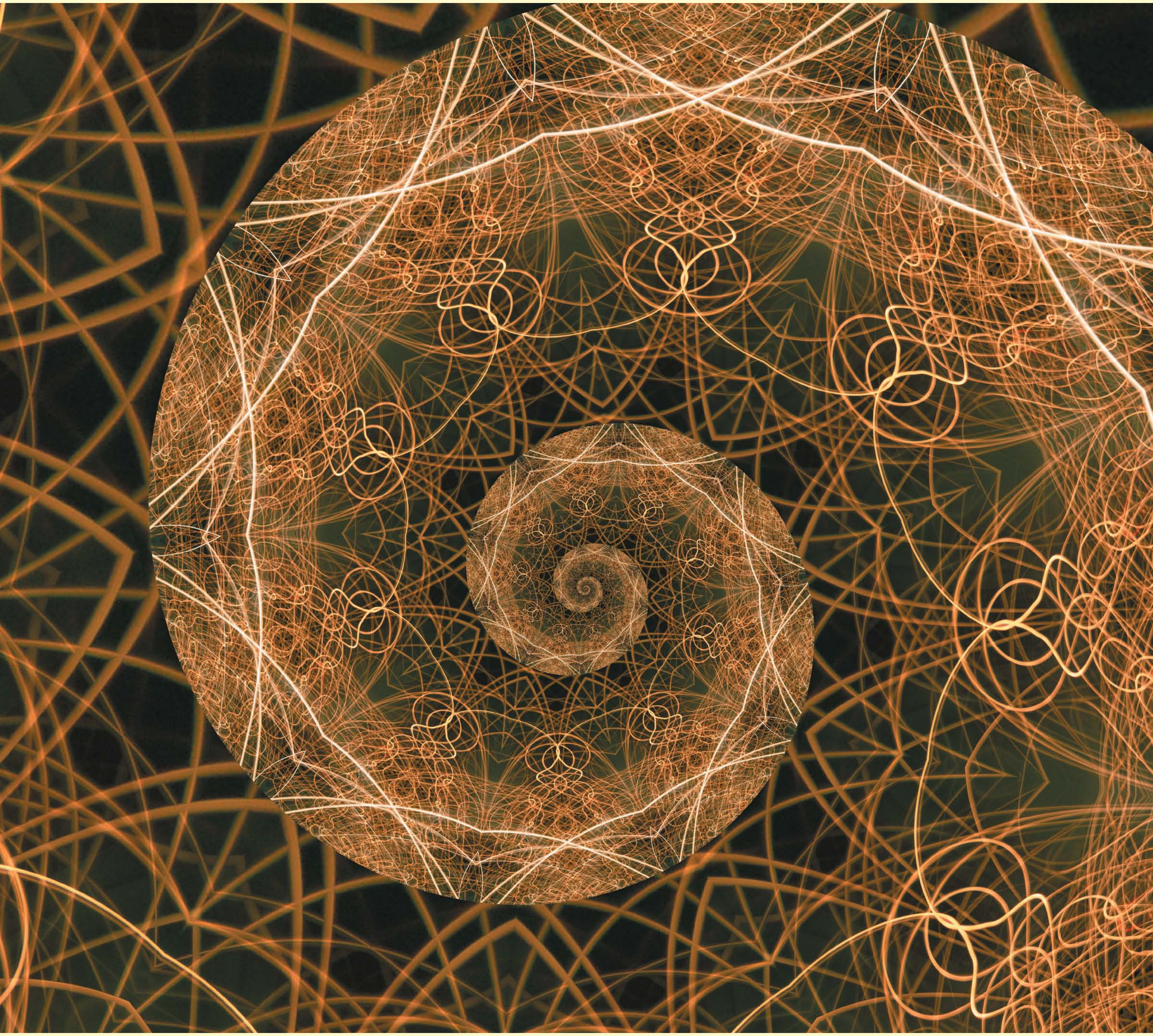
Now, what has this to do with game design?

Just as artists use the Golden Mean to proportion their works, designers should keep this principle in mind when building items that rely on proportion. Whether using the Fibonacci sequence (which approaches the Golden Ratio asymptotically) to create a curve in balancing a progression or creating an interface, the Golden Mean should be observed or at least consulted.

For instance, in creating a rectangular game board, the proportions of the board should follow the Golden Ratio. This will provide an innately pleasing proportion for the player. If a designer is constructing a building in a game and wants the player to be inherently attracted to it, they should use the Golden Mean to create the building. Any item in the game that uses proportion in any way can have the Golden Ratio applied to it.

This can, of course, be flipped on its head. In order to make players feel uncomfortable with a building, dismiss or mess with the Golden Mean in creating its proportions. The player will feel discomfort and not even know why. This can be done on the level of the building as a whole, with a single room or with a hallway.

Similarly, when creating a user interface (UI), look to the Golden Mean for proportioning its elements. Elements that use this proportional guideline will feel comfortable and natural to the user. In order to create a sense of unease, create a UI that defies the Golden Ratio. The user will still engage with the UI but will feel uncomfortable without apparent reason.



Griefing

Samaritans are players who intentionally help other players for the social rewards. These players typically reach a mid-level of experience and then focus their time on assisting other players by healing them or helping them battle monsters from a distance. This is a massively multiplayer online (MMO) behavior that is not accounted for by Bartle's types, but it is seen repeatedly. These players are not being rewarded directly by the game for their behavior; instead they are being rewarded by the other players and the social ties and positive strokes the other players provide.

Diametrically opposed to the Samaritan is the griefer. The *griefer* derives pleasure from making life (the game) hard for other players. These players will do whatever they can think of to make other players unhappy and will effectively spoil the game experience for these players. Player killing is one of the original griefer behaviors. In this case, players discovered that instead of killing monsters, they can kill one another, take loot from corpses, and other delightful things like jumping on their corpses.

About 3 percent of all players fall into this category. They block players from entering structures (if collision programming allows this), they kill steal (come in at the last second to deliver the coup de grace and claim the corpse of a monster and the loot thereon as their own), and they go to all sorts of lengths to make other players unhappy.

Sometimes griefing is an art form. Players find an exploit (a loophole that allows them to effectively cheat in the game) and lay a careful path of IP addresses and pointers that lead to the head of a rival guild.

Griefers go to great lengths to cause misery to other players. On a Player Killer server, players may hang out at the spawn point of new players and kill them instantly when they appear in the world. After doing this two or three times, they may offer to let the player join their guild. So much for benign recruiting techniques. Another well-known example involved one group of players inside a walled city who piled up crafter good that could not be removed in front of the main gate. The only other gate, the "traitor's gate," was a single person wide and could only be used by one player at a time. When players tried to leave the city, a second group of players was waiting outside to kill each player as they exited. Thus the city was held hostage.

Occasionally, griefers grief the developers. For instance, in one game, a player found that they could take the moon from the sky, put it in their inventory, and keep it there, which messed with the skybox for everyone in the game and the developers in particular.

Even though griefers constitute a tiny fraction of players, they are still numerous. If a game has 1 million players, 3 percent still equals 30,000 players who are bent on destroying the game experience for the other 97 percent of the players. Strangely, griefers are very busy people and wreak more havoc than one would think could be accounted for in such small percentages.

One of the biggest problems with griefers is how one mitigates their behavior without rewarding them. For instance, giving them a negative reputation score is a reward for these players, like becoming one of the Most Wanted in the Wild West. Many developers agree that the best method for dealing with griefers when it is clear that they have no intention of changing their griefing ways is to simply ban them from the game permanently. While one may question the permanent loss of the player revenue, losing the revenue from griefers constitutes a small percentage of revenue, and letting them stay costs money for developers in the long run when their behavior results in endless streams of customer support complaints. The simplest, most direct, and cheapest way to handle these players is to ban them from the game.

Hype

In general, *Hype* simply refers to "something that the player is expecting based on other information." Similar to Cognitive Biases, the notion of Hype involves what players bring to the game mentally, emotionally, and psychologically. The information can be real or imagined and its delivery intentional or unwitting. Naturally, this can be a factor before the game is ever directly encountered. For instance, players may have a complicated set of expectations for what they will experience in the game as a result of exposure to marketing, prior games in the same franchise, or other games in the same genre. Hype isn't just a principle for use outside a game, however. Designers can also use it within a game to prime the player's beliefs and feelings as well.

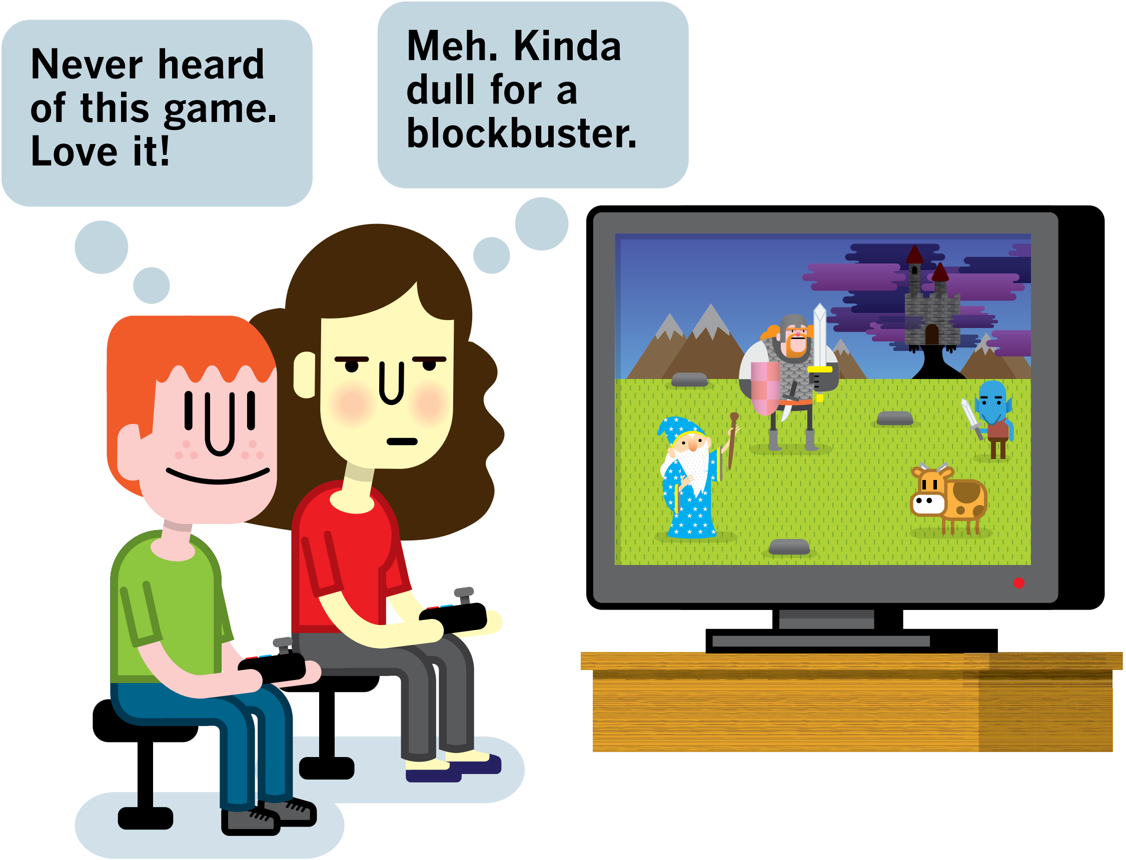
The most traditional use of the word is in the purview of marketing. For example, telling people that an upcoming game is going to be a blockbuster is an establishment of Hype. Most people will have expectations of the game's art, complexity, and scope, sight unseen, simply because it is being talked about in those terms. Similar expectations can be adopted from information about genres ("shooter," "dungeon crawl"), settings ("fantasy," "space"), or even development studios (e.g., "a typical [development studio] game"). Saying, for example, "an epic, open-world, fantasy, roleplaying game from [huge studio]" will seed different expectations in a player's mind than "an abstract platformer from [indie startup]." People will likely infer which of the two games will have longer gameplay, better graphics, and a story-based narrative. Then they will bring those expectations with them when they sit down to play the game for the first time.

The key here is this: By simply having an expectation of something, people are likely to genuinely see and feel what it is that they expect—even if it is not there. In psychology, this is called a placebo, but it doesn't just work on sugar pills. If a player is told something in a game is beneficial, they are more likely to experience it as such. If a player's character acquires a magic weapon, for instance, the player may attribute some (or all!) of their subsequent successes to the purported benefits of the weapon over and above the weapon's actual abilities. You can magnify this false belief by including properties that have nothing at all to do with performance or mechanics—for example, impressive graphics, attention-grabbing effects, or in-game narrative mystique about the item.

On the other side of the proverbial coin is "nocebo"—when a player experiences something as detrimental, whether it has the power built into it to be detrimental or not, simply because that expectation was set up for them before they encountered that thing. This nocebo effect can have an impact on specific player experiences: "I am moving more slowly because I am wearing those muddy boots." Or it can impact an entire game experience: "That reviewer really hated this game, and I do too."

Often, aspects of Hype can be utilized by game designers to shunt the responsibility of crafting the player's perception of the world from a mechanic that is difficult to implement onto one that is much easier. For example, realistic combat artificial intelligence for shooter games is difficult to develop correctly. However, players are likely to believe that the AI is "smarter" on higher difficulty levels simply because the designers increase the amount of damage an enemy can take. Nothing changes in the decisions that the AI makes—they just don't die as quickly. The players experience a "smarter" enemy (rather than a "thicker" one, so to speak) because they believe in the Hype that a harder difficulty means more advanced opponents. In this case, the use of Hype would allow the developer to present an experience for the player they may not have been able to achieve through code because of restrictions on time or money.

Naturally, Hype has its down side. If artificially crafted expectations don't hold up, the user experience can come crashing down with surprising force when a player sees past it. At that point, the opposite effect may occur—the player may believe something is worse than it actually is because they are now suspicious or resentful and the placebo has become a nocebo.


Never heard of this game. Love it!
Meh. Kinda dull for a blockbuster.

Players can experience the same game in entirely different ways depending on what they believe about it. Marketing hype and other factors set the stage before they ever begin playing.

Instant vs. Delayed Gratification

"Mr. Smith, are you sure you want to purchase this video game for $60? Tomorrow, we are having a sale ... buy one get one free!"

People work to earn money. They earn money to pay their bills from consumer purchases they have made. The video game industry is leading the way in profits from such sales. Video games use technology as their platform and it is advancing at an alarming rate, giving the world more content on a daily basis. The content is also getting to consumers faster. Due to this, societies are becoming hungrier for information, living lives faster, and losing interest in their purchases sooner, thus creating an instant gratification society. Prior to the major advances in technology, such as the Internet and smartphones, people were willing to work hard for items that they wanted, and so they gave those items more respect and consideration, and admired them more upon receiving them.

Instant and Delayed Gratification can be found in most parts of our hurry-up-and-wait lives. Hungry people rush to get food, just to wait in lines to order, or to wait at a table to be served, and to wait again for beverages prior to eating. In the game industry, this can be seen as well during the "opening night" a game is sold. If a game is being sold in a store at midnight, a line forms around the store hours before it even opens, just so the customers can be the first to purchase the game the minute it is available. The amount of sales on that opening day (instant gratification seekers) can deter others from purchasing said game (delayed gratification seekers) due to poor reviews, games being sold out, and because the Internet hosts large amounts of screenshots and information about the game almost immediately, thus spoiling it.

From a developer perspective, this behavior is added information they need while designing a game. The instant gratification players tend to rush through the games so they can get to multiplayer modes, unlock more powerful items to sell, or take advantage of newer players. Those players will often either help or hinder sales by promoting the game online with good word of mouth or bad. However, the instant players are usually *extrinsically motivated* and enjoy the game less than the delayed gratification players. The delayed players will make informed decisions based on purchase price, game reviews, and online content. They will also spend more time exploring all aspects of the game due to being *intrinsically motivated* to play the game.

From a game designer standpoint, there is a lot to take away from this difference in behaviors and many game types have been created with it in mind. Multiplayer modes are for people who want to jump right in and play some aspect of the game, while story mode takes a player through a journey. Roleplaying games (RPGs) and turn-based games are tailored to delayed gratification due to players immersing themselves in character customizations, level development, and abilities upgrades, which can take hours to perfect, just so they can experience two minutes of action.

Fun and immersion are other reasons game designers need to take these two types of players into account. If a player obtains an overpowered item too soon, they feel that they did not earn it and will exploit it more often, whereas if a player obtains an item too late in the game, then the game might have been too complicated for them to enjoy. A player may also lose their sense of immersion if they walk through an abandoned and destroyed warehouse and find that all the doors are locked and they have no method to open them. But a player can also lose their immersion if all the doors are unlocked instead. During Play Testing, designers can get a feel for a balanced approach to the instant and delayed game play styles—that is, players who check all doors instantly, those who wait to find the key to unlock the doors, and those who wait for something in the game to tell them that they are able to open the doors. Although finding the perfect balance is hard, there are ways to help balance it.

Visual clues, non-playable character (NPC) dialogue, sound cues, and story lines can set the speed for each level, for quests, item drops, item pickups, upgrading, and more, and it is good to measure and tweak those throughout the game to keep things fresh. This is also where gamification comes into play. When developers add gamified elements such as leader boards, special power-ups after reaching specific item levels, and badge rewards, they help players keep themselves immersed in games so they can unlock achievements, thus adding to their delayed gratification needs and enhancing their gaming experience.

A few gaming traits for each behavior include the following:

Instant Gratification: This type of gamer is aggressive, is fast, dies more in games, likes to upgrade more often, takes the easy route, plays offensively, likes forward moving games, does not like traveling back to areas they have already played, plays more games over a shorter amount of time, makes poor choices in longer games, plays by trial and error, gets frustrated easily when they cannot get past a specific part of a level, and has good reflexes and timing during game movement.

Delayed Gratification: This type of gamer plays defensively, stays alive longer, buys and plays fewer games but plays each for longer periods of time, enjoys working toward a goal, plays most aspects of a game, uses strategy, plans for upgrades, makes choices that affect other parts of the game and not just the level or enemy at hand, and thinks clearly during puzzle elements.

Krug's First Law of Usability

Steve Krug is widely known for his Three Laws of Usability. The foundation of these laws is his assertion that people do not use interfaces the way designers expect them to. Instead of reading directions, or carefully weighing and understanding all their options before making a choice, users scan an interface for the first, most immediately useful-seeming link; then they select it, hoping to get information they can use right away (see Satisficing vs. Optimizing). They may continue down this first, "good enough" path for as long as their actions (often clicking links) appear to bring them closer to their destination. When they no longer feel they are getting what they want, they may back out the way they came in and try another path, or they may simply quit. To avoid frustration and abandonment, design interfaces specifically and games generally with these user behaviors as guiding principles. Designers should prioritize simple navigation and clear feedback so the game leaves a positive, lasting impression.

Krug lays out three laws, but his First Law of Usability is the most profound. He states it simply: Don't make me think.

Since so many people will avoid reading (or even looking for) directions, a good interface must be intuitive, self-evident, obvious, and self-explanatory. There are two keys to achieving this.

- **Simplicity.** Every element of an interface should be boiled down to its bare essentials. Users will simply skip lengthy descriptions, obtuse in-jokes, and clever metaphors (see Details). They want what they want—not what they are told they want by some overbearing designers who think they know better. So focus on clarity and simplicity on heads-up displays (HUDs), scoreboards, rewards, and the Core Gameplay Loop. The places where players directly interface with a game—where they put their hands and their clicks—should be so simple and straightforward that it becomes transparent in the experience. If players have to think about where to click next, or how to hold the cards correctly, or which buttons to use on the controller, they will not be thinking about how fun your game is. Rather, they will be thinking the opposite (see Working Memory).
- **Consistency.** The second hallmark of good interface design is consistency. Although players do crave and appreciate a certain amount of surprise in their game experience, they don't enjoy it in the interface. Buttons with the same function should have consistent placement on the screen, general aesthetics, and affordances (see Affordance Cues). If two things look alike, players will assume they are alike. The corollary to this is that if two things look different, players will assume they are different. So if the Replay button is usually yellow, players will assume all yellow buttons involve restarting the game. Changing the color of the Replay button every time a player sees it will only frustrate them and drive them to quit. This holds true in all aspects of the game, not just the user interface. If two enemies look or act the same, players will assume they are identical (see Gestalt); they won't stop to think about subtle differences or context.

Ultimately, the thing to remember is that designers should not require the player to figure out what to do or where to go; as the rule says, if they have to think about it, they're already gone.

Keep in mind that this does not mean removing challenges, mystery, or puzzles. It merely means clearly communicating to players what the end goals are, making them transparent. For example, a player encounters a room with no obvious exits: Is it a puzzle to be solved, or a dead end to be avoided? Don't make them think. Make it clear which is which. That way they can get to the fun part—either puzzle solving or exploring, depending on the answer to the question.

Krug's other two laws of usability are also important for interface design but have fewer applications in a more general sense.

- Rule #2: It doesn't matter how many times I have to click, as long as each click is a mindless, unambiguous choice (see Risk Assessment and Satisficing vs. Optimizing).
- Rule #3: Get rid of half the words. Then get rid of half of what's left (see Doubling and Halving). Many people feel some friction when encountering words, especially a large collection of them. Reading requires thinking and . . . well, see Rule #1.

Music and Dopamine

Many people have experienced chills while listening to a piece of music. It is a strange, yet surprisingly pleasurable sensation. This feeling is brought on by dopamine being released in the brain. Studies have found that listening to music can trigger such a release.

Dopamine is a neurotransmitter that creates feelings of pleasure or enjoyment when released. It ties heavily into the brain's reward system, which releases dopamine as a response to a variety of biologically desired stimuli, such as food and sex, as well as during *reward prediction*—when a person or animal is anticipating a reward (e.g., the feelings of joy you get when you hear you just won a prize, or what Pavlov's dog felt whenever he heard a bell).

As previously mentioned, music can trigger dopamine release; however, not just any beat or melody will do. Dopamine receptors are known to adapt quickly to repetition, which is why a song that is too repetitive can be perceived as boring. The brain can perceive new, exciting changes introduced through a song as a kind of reward. In fact, the brain has been observed releasing more dopamine when the reward is even greater than what the subject expected. Anticipation also helps facilitate a larger reward response. Eating at your favorite restaurant every day is nowhere near as satisfying as when you come back to it after a four-month gap. This same effect is used in music to varying degrees. Classical music often has motifs that appear in a song multiple times throughout a piece—sometimes in a slightly different and surprising way (different arrangement, major to minor, etc.). In modern music, this is most easily observable with a song's chorus, which is often considered the most enjoyable (or catchy) part of a song; the verse, bridge, and other parts are purposely more subdued and less dynamic, so the song can build up anticipation until it hits the listener with the bombastic chorus they've been anxiously awaiting. If a song were only four minutes of the chorus, it would have less of an impact on the listener and less of an effect on their dopamine levels.

How does this relate to video games? A good game designer should be concerned with not only creating robust and interesting gameplay, but maintaining a firm command of the player's emotions. Music and sound design are pivotal components in accomplishing that. Similar motifs (melodies, arrangements, tempos, etc.) can be used to great results if there is enough variation between repetitions of them and/or the tension and anticipation build in the music. Keeping the player's attention is another important aspect of game design, and a dynamic soundtrack can greatly contribute to this.

Although repetition should not be vilified when designing music, the designer should make sure that the composition tries to maintain a balance between small, gradual changes (e.g., the same three-chord progression, but with slightly different accents on the drums, and different instruments chiming in and out), large, drastic changes, and returning to common themes or parts after anticipation builds up. Small melodies or *themes* triggered by in-game events, such as opening a treasure chest in The Legend of Zelda or beating a level in Super Mario Bros., also help these events play into reward prediction. Performing these activities, especially when pleasant music is tied to them, causes the player to associate these events with something highly enjoyable. The release of dopamine and the enjoyable sensations felt as a result are some of the main reasons people play video games, and music is a very integral component.

Pacing

If an experience simply repeats one action over and over again—even if that action is a steady stream of slapstick comedy or shocking scares—the emotion it engenders is only boredom. The best games (and comedians and horror movies) give the audience breathing room between thrills. There needs to be areas of sanctuary where the audience is lulled into a sense of security during a point and has time to breathe before they are thrust into the action once more. This ebb and flow in the low points and high points is what defines *pacing* and can be the key to maintaining an audience's engagement (see Interest Curve).

In Western entertainment, pacing is said to be built upon these basic building blocks:

- **Tempo:** The beat/rhythm/timing of the current level of action being encountered by the protagonist (Player).
- **Tension:** The mood of the level or perceived danger felt by the protagonist (Player).
- **Threat or Suspense:** The threat of danger to the protagonist (Player).
- **Movement Impetus:** The willingness of the protagonist (Player) to continue moving forward.

Each of these elements overlap, but if Play Testing shows there is a problem with a game's pacing, check each of these areas individually for problems.

Frustratingly enough, this is about as far as Western traditions go in describing good pacing. The creator is expected to have an intuitive sense of good pacing, and tune the above elements until it "feels" right.

Fortunately, creative people all over the world have been pondering this problem of pacing entertainment for centuries. Zeami Motokyo, a master of Japanese Noh theatre in the 14th century, formulated an approach for ideal pacing. He called it *Jo-Ha-Kyu* (序破急, literally "start-break-rapid") and it consists of the following:

- **Jo (start):** This is the start of the sequence. Here, the energy of the scene/action/subject is very calm and restrained, but slowly building up.
- **Ha (break):** The energy of the Jo has built up tension, and it finally "breaks" suddenly and rapidly into more intense action. This moment is sometimes called the reversal or main plot twist in Western techniques.
- **Kyu (rapid):** As the energy of the Ha continues on, more tension builds until there is a final motion that dispels all of the energy and tension built up to this point. This can be viewed as the Climax in regards to Western storytelling.

The visual that helps explain this concept is a stream of water. At the beginning (Jo), a stream is small and pretty. It moves along at a good clip, but there is not a lot of pressure. As the stream goes along its way (Ha), it grows and becomes a rushing river. There are twists and turns until finally it breaks over an edge. And then the water rushes dramatically (Kyu) down in a waterfall, inspiring great awe. What's key is the idea that the momentum continues. The water pauses peacefully in a pool at the bottom of the waterfall (Jo), then begins the cycle again.

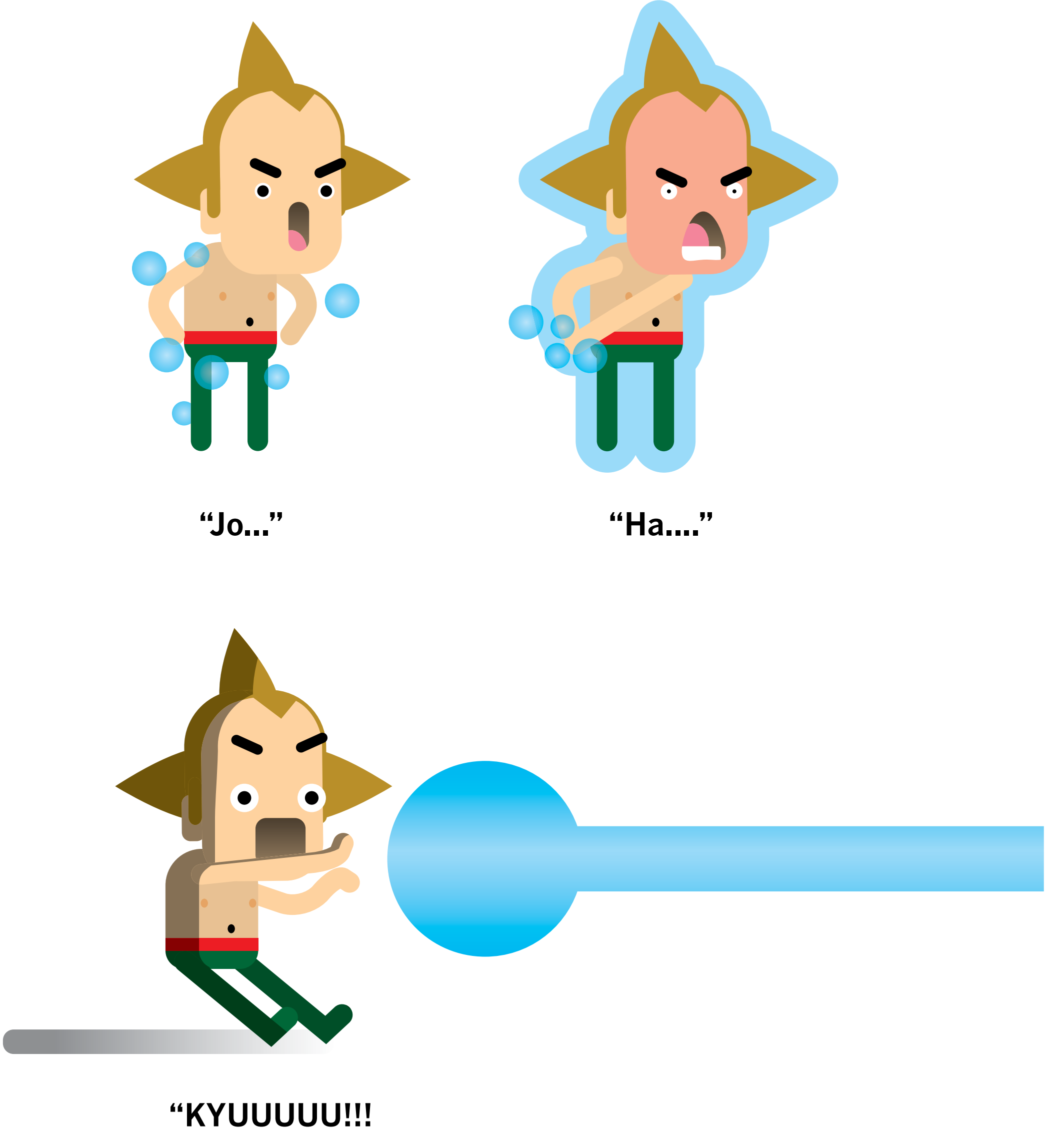
Jo-Ha-Kyu communicates the sense of movement and the satisfying feeling of a well-paced story or experience. It describes how to avoid getting reviews with vague comments such as, "The game dragged on in parts. I got bored." It describes a dynamic way of moving things along, and not just remaining a static map.

Zeami Motokyo believed that all things in nature exhibit some form of this pattern, and that it should be used extensively within the arts, especially in Noh theatre. There it is utilized in the movements of the actors, the flow of the dialogue and music, as well as the structure of scenes.

Jo-Ha-Kyu is often instinctively implemented within a "level" structure of a video game. The level starts out relatively easy, with introductory opponents (*Jo*). The difficulty in puzzles or opponents increases steadily and plot twists are introduced (*Ha*) until the difficulty significantly jumps up as the player encounters a boss fight (*Kyu*), and once defeated, the level ends, and the pattern begins again in the next level.

One particularly excellent example of *Jo-Ha-Kyu* in games is Space Invaders—the gameplay starts out calm, the difficulty slowly rises in the middle, and the end becomes very frantic before resolving once again (temporarily) into calm.

Just as in Noh theater, even small-scale actions in games demonstrate *Jo-Ha-Kyu* pacing: Mega Man's charge buster, Ryu's hadouken, and many fatalities in Mortal Kombat are paced in this manner. In the NES classic Punch-Out, as the characters waiver back and forth (*Jo*), they perform some form of action or wind up to telegraph their move (*Ha*), and finally release their punch (*Kyu*) on Little Mac. No matter the scale, *Jo-Ha-Kyu* can be the solution to most pacing problems.

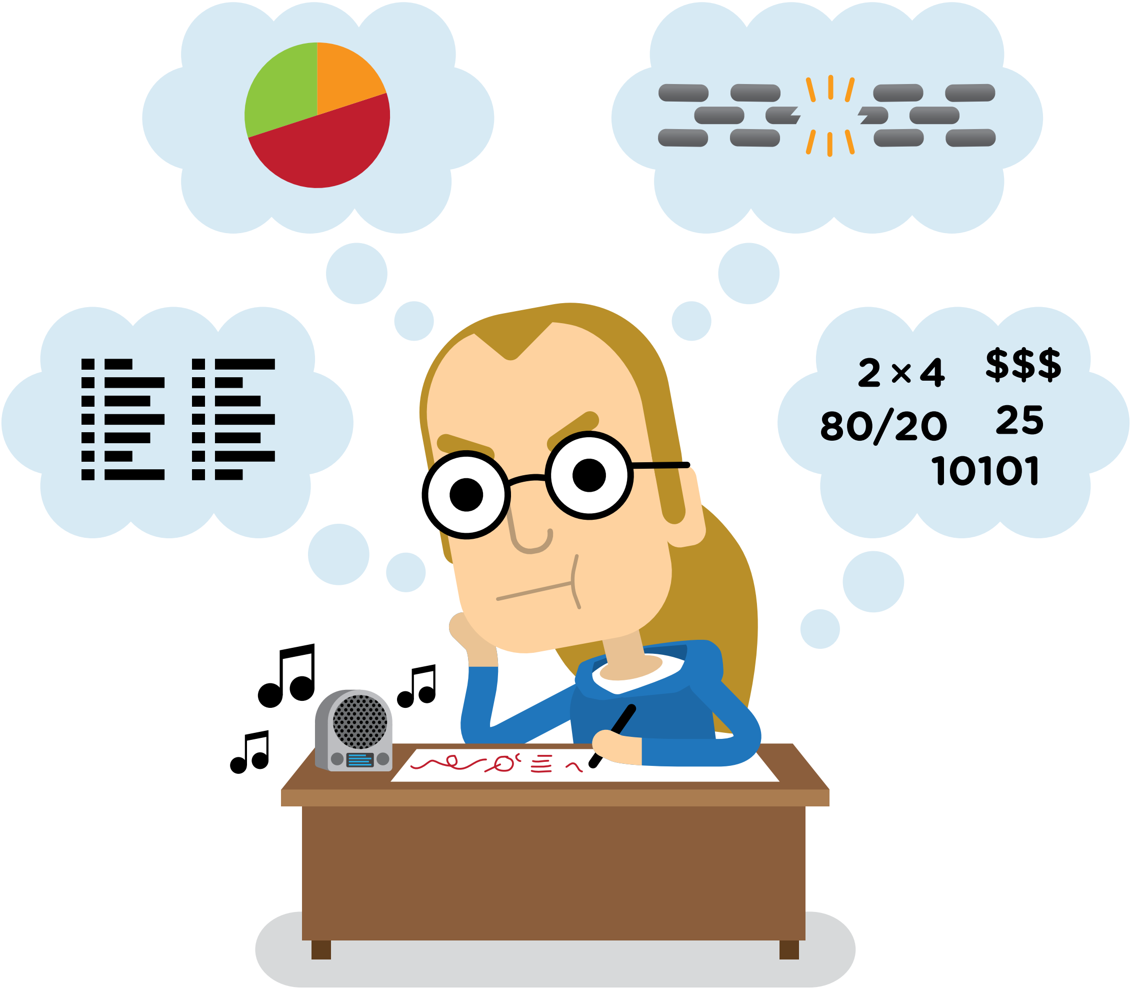

"Jo..."
"Ha...."
"KYUUUUU!!!

Problem-Solving Approaches

So much of game design is about solving problems. Whether it's the problem of how to make a functional game into a fun game or how to get a project funded, problem solving is the number one task on the designer's daily to-do list. They tackle problems from the other direction as well—inventing and creating interesting problems (aka challenges) for players to solve and tuning those problems so their solutions relate to each other and have a logical progression (see Balancing and Tuning).

Game designers need to clearly understand how humans go about solving problems. This can help them more quickly write design documents and resolve gameplay problems, but it is essential that they also be able to anticipate players' problem-solving behaviors when they are managing difficulty levels and player engagement. Designers must walk the tight rope of nudging their players into successful problem-solving strategies without being too obvious (see Advance Organizers, Hype, and Cognitive Biases).

See the appendix at the end of the book for an extensive list of problem-solving methods.



For every problem, there are many different approaches to take when solving that problem. Make a list, draw a chart, find the weakest link, writing the problem as numbers. Check the appendix of this book for many more suggestions.

Satisficing vs. Optimizing

Classical economic theory says that people make decisions through a process—either conscious or unconscious—that goes something like this:

1. Make a list of all possible outcomes.
2. Assign each possible outcome a probability of happening.
3. Assign each possible choice a value based on its outcome.
4. Then choose the optimal outcome.

Take buying a dessert as an example. A hot fudge sundae provides $3 of happiness and costs $1, whereas a banana split provides $1 of happiness and costs $2. The sundae is then the obvious choice because it provides more value and optimizes the outcome, where more value is received than is spent. Classical economic theory suggests that people will optimize their options and always choose hot fudge sundaes.

However, real humans don't make decisions this way. It is often too time-consuming, complicated, or inconvenient to gather all the information and appropriately optimize every variable. Maybe in special cases, like a splurge on a dessert, people will weigh everything carefully. But take something that everyone does every day: choosing what to wear. Nobody creates a spreadsheet listing everything in their wardrobe sorted out by temperature appropriateness, fashion trends, a clash matrix, or a list of what they wore for the last seven days before they get dressed in the morning. This kind of labor-intensive, optimizing solving for the perfect solution just doesn't happen most of the time for most everyday decisions.

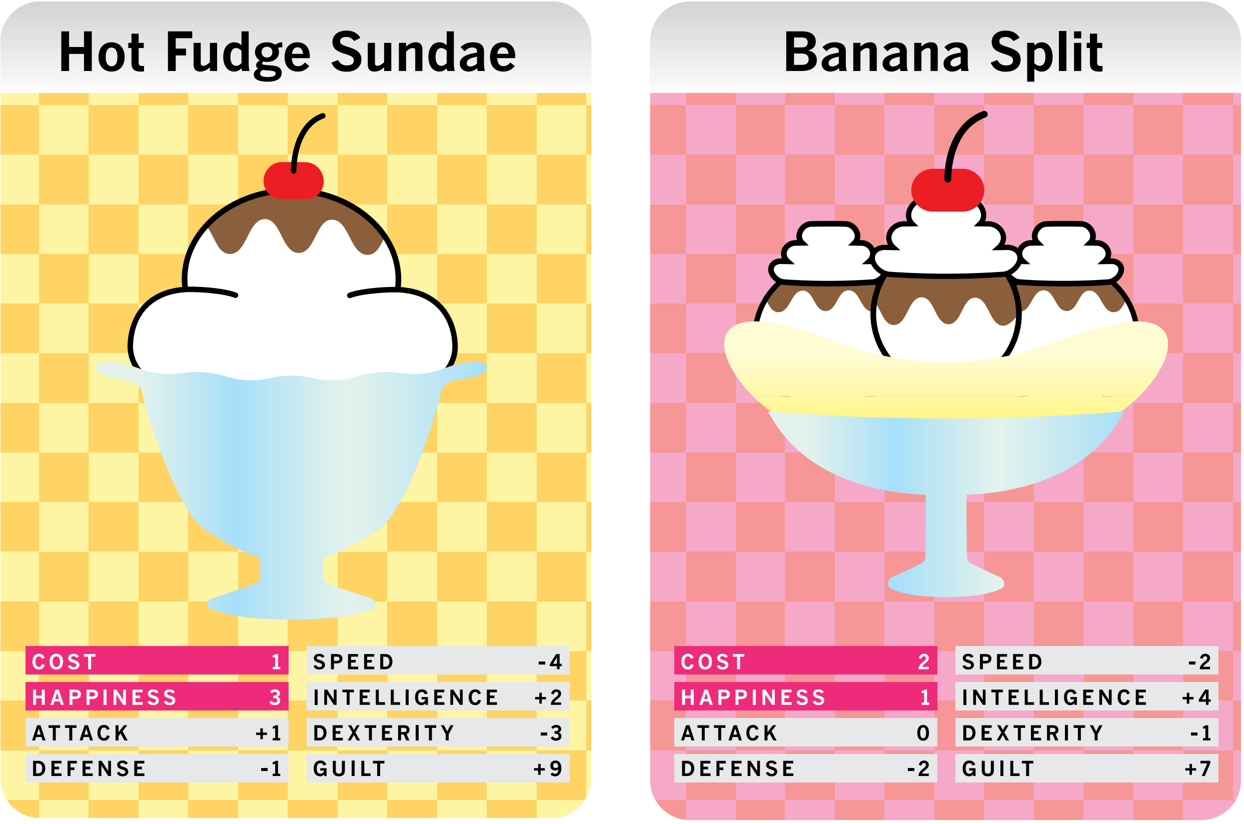
Instead, people do what is called *satisficing*. They pick the first satisfactory result based on essential rules of thumb, or requirements that get them close enough to the optimal solution. Using the dessert examples, a person might choose the banana split if it is easier to reach or becomes available sooner. It gets the chooser close enough to their desired outcome. In choosing what to wear, people often make their decisions based on a few simple rules, like what is clean and has worked as an outfit in the past.

The benefit of satisficing is that it is so much easier for a person to do than optimizing. When creating games, don't force players to optimize what can be satisficed.

Here are some examples where satisficing vs. optimizing enter into the game design process:

- Players will approach puzzles and challenges in games often using satisficing to find the fastest, easiest solution, rather than optimizing at great length. For instance, if it is easier to look up a walkthrough video on the Internet than to solve an extremely difficult puzzle by themselves, players will often take this shortcut and avoid fully experiencing the game as it was intended. This isn't a matter of cheating, it is simply that the players are weighing how valuable their time and attention are.
- Game developers are always working within constraints (see Pick Two: Fast, Cheap, Good) where optimizing may not be possible, but satisficing is a very good way to decide which features make it into the final game, how much polish a particular element gets, and how much time is spent making the game perfect, rather than just good enough.
- In games involving multiplayer cooperation, players often choose to satisfice rather than optimize their personal outcome if it will ultimately work to improve the chances of achieving the shared goal, especially if it is obvious that there is no way for an optimal outcome for all players (see Nash Equilibrium).

Personality traits do enter into this process. Some players enjoy optimizing (see Bartle's Player Types) and avoid satisficing. So, as always, it's important to understand the target market. Optimizing vs. Satisficing algorithms have been developed in the fields of economics, decision theory, and game theory. It was developed and popularized by Herbert Simon in the middle of the 20th century.



When choosing between two options, such as these desserts, people rarely weigh every possible variable involved to carefully optimize the results. Rather, they rely on rules of thumb and obvious benefits, such as focusing on the cost and happiness variables, which satisfy and get close enough to the optimal result without a lengthy deliberation.

Sense of Accomplishment

Few things in life are as rewarding as a sense of accomplishment. This can be seen on a baby's face when they first pull themselves up to a standing position, or on a Bingo player's when they shout "Bingo!" A sense of accomplishment will drive a player to continue in a game, and the lack thereof can drive them away.

A sense of accomplishment will follow naturally at the end of a task that was satisfying to complete. This may be a task that the player has found to be somewhat difficult or challenging. The extreme example of this is called *fiero,* that feeling of overcoming an obstacle that was a large challenge; it usually results in pumping one's fist in the air or shooting up both fists in victory (see Lazzaro's Four Keys to Fun). These moments in gameplay are key to keeping a player engaged and relating to the game context without getting frustrated, bored, or overwhelmed (see Interest Curve).

Think of the sense of accomplishment as being like rat food pellets. When experimenting with rats, researchers often use food pellets to reward the rats for tasks, such as pushing levers or completing mazes (see Skinner Box). Those small rewards keep the rats motivated to perform the next tasks researchers set before them. Without that regular, positive reinforcement through a sense of accomplishment, players will refuse to continue in a game. When the designer supplies a steady stream of accomplishable goals that are somewhat challenging (see Balancing and Tuning), the player has a sense of challenge and reward that is more satisfying than a simple task that is not challenging, even if it is accompanied by the same reward. The importance of the challenge is key to the feeling of satisfaction (see Koster's Theory of Fun).

A feeling of pleasure and fulfillment is key to the sense of accomplishment, and that requires real effort. A simple, single mouse click for a reward is not very satisfying. A relatively complex task followed by a significant reward will usually deliver where the single mouse click does not. One provides a tension, which is relieved by the completion; the other simply lacks any tension whatsoever.

Balancing simple and complex tasks is part of the art of game creation. If every task is a challenge, the player may get overwhelmed or discouraged. On the other hand, if every task is too simple, the player will grow bored. By supplying a good balance (see Balancing and Tuning) of easy and complex tasks, a sense of accomplishment can be continuous and sustained. It is a delicate process to maintain equilibrium between challenge and boredom (see Flow). If balanced properly, a sense of accomplishment ensues and player engagement increases (see Interest Curve).

One behavioral model often used in teaching (see Koster's Theory of Fun) attempts to dissect exactly how to deliver a sense of accomplishment. It is known as ARCS and is championed by John M. Keller. The letters stand for

- **Attention.** *Attention* in this context refers to the effort and willingness on the part of the audience or student.
- **Relevance.** *Relevance* here refers to the use of terminology and examples the audience is familiar with already. Using elements the audience (or players) is already familiar with to explain the new material or skill demonstrates exactly how the material is relevant to previous experience.
- **Confidence.** *Confidence* refers to the process of providing positive outcomes and feedback so that people understand how well they have succeeded at assimilating the new skill or information.
- **Satisfaction.** *Satisfaction* comes in the form of reward for achieving the learning; this can be as simple as praise or as complex as fiero.

The ARCS model hinges on the idea that people can be motivated to learn because the knowledge is valuable, and the job of the instructor (or game) is to demonstrate that value in easy-to-understand terms. Then when the player (or student) realizes they have acquired this new, valuable skill, they are filled with satisfaction—or a sense of accomplishment.

An easily conquered enemy or effortlessly acquired skill will never provide the same rush that comes from a sense of true accomplishment.

Spatial Awareness

Spatial awareness is being cognizant of one's position in space in relation to the surrounding objects and environment. It mostly comes into play in video games that depict virtual worlds or spaces. Some board games use it as well, but they depend heavily on players' imaginations to make the effects work. The idea that architecture and environment can influence human experience and emotion seems counterintuitive at first, until closer inspection into real examples makes it more clear.

Each of the following categories describes a real phenomenon that people experience viscerally and architects have been consciously exploiting for centuries. What works in real buildings and cities also works in virtual ones.

Positive Impacts on Human Psyche

- **Prospect-Refuge.** Prospect-Refuge is a concept on two sides of the same coin. On the one hand, people show a marked preference for environments with unobstructed views (prospects). Some instinctive part of the human mind feels safer knowing it can see or hear any threat nearby. So encountering a space that has wide-open views, such as the top of a mountain, is deeply appealing and satisfying.

  On the other hand, they also prefer areas of concealment and retreat (refuge). That same deep part of the human mind enjoys "safe" environments like hiding places and dens. If that instinctive paranoia is satisfied by being able to see threats coming from a distance, it is equally happy to find itself in an enclosed area where no threats can get in.
- **Cathedral Effect.** The Cathedral Effect is a remarkable relationship between the height of a ceiling and how people think. People who find themselves in a room with very high ceilings are prone to do more concrete and detail-oriented thinking. They think better, the higher the ceiling is. The effect's name comes from how it has been used over the centuries in the architecture of religious spaces. People feel a sense of elevation and clarity in places with high ceilings.

Negative Impacts on Human Psyche

- **Agoraphobic Spaces.** Agoraphobic spaces are related to the Prospect concept described earlier, but agoraphobia is a pathological fear of such vast, large, open areas. To tap into this feeling of anxiety, provide a sense of vulnerability along with the vast views. Mechanically these environments may have a total lack of, or extremely limited, options for the player in which to seek refuge and take cover from ranged attacks. The dynamics of these spaces set up enemy encounters such as snipers, bosses, aerial attacks, and distant mortars.
- **Claustrophobic Spaces.** Claustrophobic spaces are aesthetically narrow, constricted areas that feed into player anxiety when they perceive that they are trapped in a small space with a serious threat. Mechanically, these environments also usually include multiple hiding places and the potential for secret entrances. Conversely, sometimes the most straightforward approach is the best. This means literally trapping the player in sight of the enemy. Either of these approaches feeds into the fear players feel as the threat is closing in.

Deep and Complex Worlds

Of course, the real world is made up of a variety of these architectural approaches and not just a single, monotonous type of space. Combining a variety of environments can push players through a linear experience (see Wayfinding) in a calculated way and provide interesting adventures. Here are some established tactics for making this variety of environments work together effectively:

- **Defensible Spaces.** Defensible spaces are a balance. They are neither too constricted nor too open. Mechanically these environments provide easy access to both prospect and refuge. Aesthetically, these environments contain architectural features that convey the perception of value and ownership. The dynamics of these spaces afford the player the perception of proprietorship, control, and a sense of empowerment over the enemies in these territories through their easy access to advantageous locations.
- **Lighting Design.** As with all the other principles listed here, lighting design is a discipline with years of research behind it. A short summary here can barely do it justice. Basically, lighting is a powerful tool for manipulating the environment. For example, humans feel safer and happier in lighting that mimics daylight. Yet, they also feel drawn to a warm, yellowish glow similar to fire—especially in an overall darker room. Lighting can also focus attention on a particular corner or space using brightness and color variations.

The human psyche feels relieved, safe, and elated in environments with wide-open vistas of surrounding terrain. This is the “prospect” side of the Prospect-Refuge concept.

Time Dilation

In Einstein's theory of relativity, *time dilation* is the term used for a difference of elapsed time between two events as measured by separate observers either moving relative to each other or differently situated from gravitational masses. Essentially, time is not only *perceived* differently by the two observers, it is actually different for them.

Although literal relativistic time dilation is not regularly experienced by human beings, perceptual time dilation is very real and occurs often in games. This experience can be described as time "dragging" or "flying" during any activity. The expression that "time flies when you're having fun" is an expression of the perception of time dilation. The feeling that the last hour of work on a Friday is dragging is another common perception of time dilation. In both of these cases, time has not changed at all, objectively speaking; an hour still takes precisely one hour to pass. What can change is our perception of how time passes.

Oftentimes, when immersed in a game, a player can look up and be surprised to see that an hour or more has passed when the player feels that only a few minutes have passed. This time dilation experience is similar to that which occurs during Flow. Presumably, the player has even achieved flow, wherein challenge is at just the right level to balance frustration and engagement.

Players of the game Tetris often experience time dilation, accidentally spending hours playing when they only intended to play for minutes. This occurrence of time dilation, where the player perceives time to be passing more quickly than the clock, accounts for what happens in many of the best games. Like any pursuit that is fun, time passes quickly.

When play testing a game, players may find that time drags by; this is, of course, the inverse of what the game maker hopes for when they are designing the game. If the design is good, and is *fun*, time should dilate in a positive way rather than in a negative way.

Many times, when playing a complex RPG, players get bogged down in screens of narrative, pushing the next button with ennui or outright boredom. For these players, time is dilating in a negative way. If players complain of time dragging, this is a sure sign that the game maker has not found the elusive fun game designers seek to create. Time should, in fact, fly if the player is having fun.

Working Memory

Psychologists classify memory into two types. *Long-term memory* is accessed when someone picks up the controls for Super Mario Bros. and flawlessly remembers how everything works even if they haven't played the game for a decade. *Working memory* is the memory we use when someone tells us their phone number or how to use the controls for a new game.

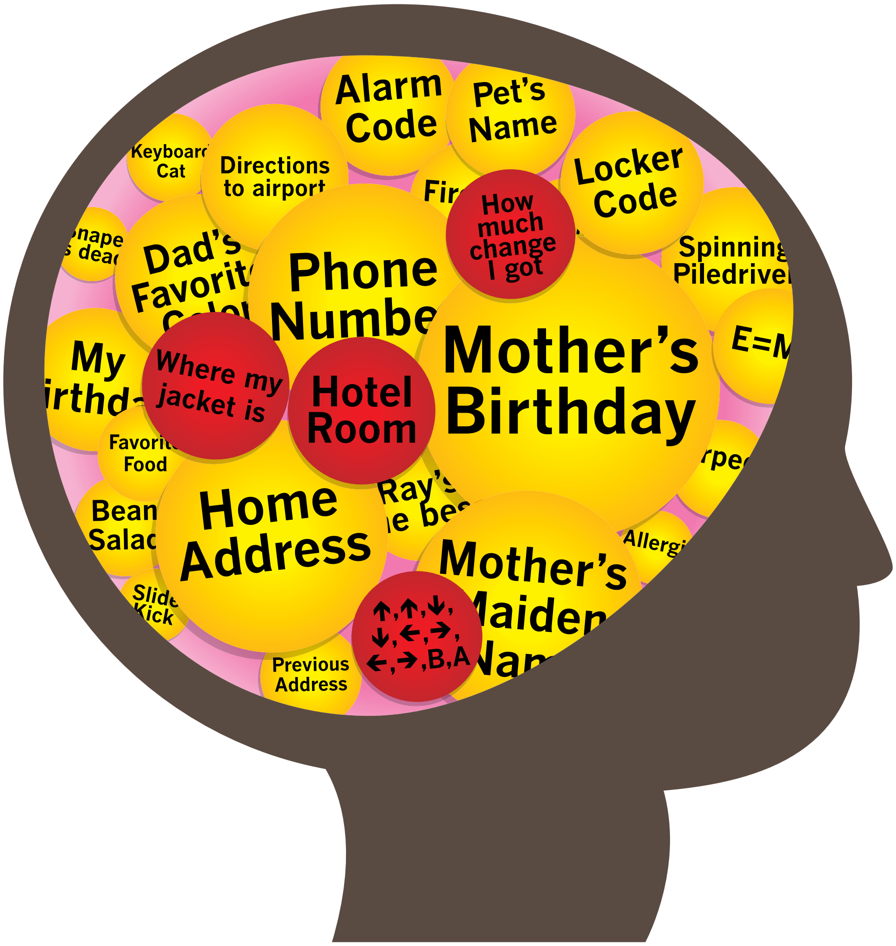
All memories are stored first in working memory; then they are moved to long-term memory if our brain realizes we may need to hold onto this information for more than a few minutes.

Working memory is not very robust. A convenient rule of thumb is that the average person only holds four things at a time in their working memory. This can seem very limiting when introducing players to a new control schema or memory puzzle. Fortunately, there are ways to make the most out of those four items. For instance, when people try to remember a phone number, they don't remember each number individually, but as what psychologists call *chunks.* The area code is one chunk, the exchange is the next chunk, and the last four digits are the last chunk. So the working memory is only holding onto three chunks, plus the name of the person associated with the number. The total amount of information is 10 digits plus a name, but working memory is usually not too tasked.

Another limitation on working memory is time. If new information is not accessed or used repeatedly and quickly, the information will be lost.

When designing games, be cognizant of the limitations of working memory. Attempting to teach players nine new things during a tutorial will probably never work. There simply isn't room in their working memory for them to integrate that much information all at once. Whatever they learned first will be pushed out by whatever they learned last, and when they move from the tutorial to the actual game, they will not have the information they need to feel successful. A better approach is to unveil new mechanics or controls at a regular pace throughout a game. By doing this, the game waits for a player to internalize an instruction into long-term memory before it taxes his working memory with a new idea.

This relates to a concept called the Learning Curve. Games with steep learning curves often push working memory to its limits, asking players to learn many new things before they can feel comfortable in the game. Examples of games with these kinds of front-loaded learning include Dwarf Fortress, Dark Souls, Eve Online, or League of Legends. Games with shallow learning curves do not necessarily have fewer features or depth, but they put less strain on working memory. Examples of such games include most Facebook social games, many racing games, and NBA Jam.



🟨 **LONG TERM MEMORY**

🟥 **WORKING MEMORY**

New information is stored in working memory as “chunks.” If it is not used quickly and repeatedly, it will not be moved to long-term memory and will be replaced when additional information is encountered.

Zero-Sum Game

A Zero-Sum Game is one in which winnings are perfectly balanced against losses. If there is a single possible outcome for which the sum is not zero, then the game cannot be Zero-Sum. For example, Poker is a Zero-Sum Game, as every player's winnings are exactly matched by another player's losses. There is no way to win more than the bet amount and no way to lose more than the bet amount. Because the win/loss ratio is fixed, all outcomes in Zero-Sum Games are Pareto Optimal (see Outcomes: Pareto Optimality).

Rock, Paper, Scissors is another example of a Zero-Sum Game; for each throw, exactly one player wins and exactly one player loses (unless there is a draw). This table shows that the total payoff for every game of Rock, Paper, Scissors is equal to zero:

	Rock	Paper	Scissors
Rock	Draw (0, 0)	Paper Wins (–1, 1)	Rock Wins (1, –1)
Paper	Paper Wins (1, –1)	Draw (0, 0)	Scissors Win (–1, 1)
Scissors	Rock Wins (–1, 1)	Scissors Win (1, –1)	Draw (0, 0)

The Prisoner's Dilemma, on the other hand, is not a Zero-Sum Game, since both players may "win" together or "lose" together and the total amount of time spent in prison does not average zero (in fact, a result of zero years spent in prison is only ever possible for one player):

		Prisoner B	
		Cooperate	**Refuse**
Prisoner A	**Cooperate**	Each receives 6 months (–1, –1)	A goes free; B serves 5 years (0, –10)
	Refuse	B goes free; A serves 5 years (–10, 0)	Each receives 2 years (–4, –4)

Zero-Sum Games may be solved with a Nash Equilibrium, or a *mixed strategy* can be used. The mixed strategy overcomes the inherent balance in Zero-Sum Games played one time by increasing the minimum payoff amount over a series of plays. In the example of Rock, Paper, Scissors, a mixed strategy would be to randomly select a throw for each game, resulting in an average of 1/3 wins, 1/3 draws, and 1/3 losses. This is superior to the minimum average value of –1 for each individual game, so the advantage goes to the mixed strategy in the long run, which allows both players to achieve success a minimum of 33 percent of the time. This type of strategy is also called a *minimax,* since it increases the minimum payoff that the player will receive (see Minimax and Maximin).

Non-Zero-Sum Games are not as clean as Zero-Sum Games because multiple players may win or lose, or all players may win or lose. Strategies for Zero-Sum Games may not apply well to complex Non-Zero-Sum Games, such as economic or psychosocial models. Global thermonuclear war is a good example of a Non-Zero-Sum Game in which all participants are likely to lose and none to be better off than they were before the "game" was played.

John von Neumann and Oskar Morgenstern‘s research in the mid-20th century showed that every Zero-Sum Game has a minimax solution, and that a mixed strategy of randomization always improves the minimum payoff, even if there is no Nash Equilibrium.

About the Contributors

Keyvan Acosta is continually developing his talents and passions as game designer, musician, writer, and teacher, sharing it all as a course director in the Game Design department at Full Sail University. In recent years, he's also participated in various industry conferences as a contributor, speaker, and judge, including Art History of Games, Game Developers Conference, Project Horseshoe, Siggraph Bogotá, and SXSW Interactive. Keyvan's credits include game designer at ZeeGee Games, programmer, and creative consultant for various companies like CyberMedia (PR), IGDA Global Game Jam, Mekensleep, and MuninuM.

Liz Canacari-Rose was born and raised in Denver, Colorado. She has studied and worked in the IT industry since 1997, from hardware support to web development to software and game development. In 2000, she had the rare privilege of working with a company to create interactive, 3D training solutions for the medical industry. This, along with a lifetime passion for video games, piqued her interest in the game industry. She went on to receive two degrees from Full Sail University—a BS in Game Design and Development in 2006, and an MS in Entertainment Business in 2009. She continues to develop her own games through her small business, in addition to teaching Game History at Full Sail University.

Michael Deneen has been a professional game designer for over seven years. He most recently designed and implemented levels for PlaysStation All Stars: Battle Royale, God of War: Ghost of Sparta, and God of War: Chains of Olympus. Previously, Michael worked in various roles on the Fantastic Four, Sopranos, Medal of Honor, and James Bond franchises.

Zack Hiwiller is a game designer, educator, and writer who lives in Orlando, Florida. He is the department chair for the Game Design degree program at Full Sail University and the lead designer at Sky Parlor Studios. Previously, in addition to his independent projects, he was a designer at Gameloft and Electronic Arts (EA). His writing, at hiwiller.com, has been reposted by Kotaku, GameSetWatch, and others and has reached over a million readers.

Dr. Jeff Howard is an assistant professor of Game Development and Design at Dakota State University, where he plays a key role leading the narrative focus area in the Computer Game Design program. He is the author of *Quests: Design, Theory, and History in Games and Narratives* (A K Peters Ltd., 2008) and the program committee chair for the Workshop on Integrated Design in Games (IDIG). He continues to develop his transmedia work, Arcana, while filling the quiet moments by researching and writing for his book on magic systems in games.

Christina Kadinger is currently a professor at Full Sail University in the Game Design program. She has a BA in Economics from Rollins College and a Law Degree from Barry University. She enjoys playing games, seeing live bands, and playing ice hockey in her spare time.

Chris Keeling is a video game industry veteran; he first worked on Panzer Elite in 1998, followed by America's Army, Twilight War, Order of War, and other military titles (which probably has something to do with his 23 years of service in the US Army and Army Reserve). He is currently chained to a desk in Eastern Europe at Wargaming.Net, where he is a senior producer on the highly successful World of Tanks, as well as on the forthcoming World of Warplanes, World of Warships, and several unannounced projects. Chris has also worked as a game writer, game designer, and game educator, and until recently, as Program Manager for the Game Design program at Full Sail University, where he still serves on the Program Advisory Board. He also volunteers on the Executive Committee of the International Game Developers Association's Game Writing Special Interest Group and blogs about his tedious life at www.aconnecticutyankee.com.

Casey Kuczik is a senior producer at Ubisoft Entertainment in Paris, where he is part of a team responsible for advancing mobile game development and publishing. During his career, Casey has held the roles of head of mobile, producer, designer, writer, and tester at the companies Bigpoint, Electronic Arts, and Seven Studios. Since 2009, Casey has been an instructor at Full Sail University, where he teaches Design and Development Analysis, a course focused on deconstructing video games with a critical perspective; this is an original course of his creation. Casey graduated from Yale University in 2003 with a BA double major in American Studies and Film Studies, and in 2010, he completed an MBA from Loyola Marymount University.

Nicole Lazzaro is a world-renowned game researcher, designer, and speaker who makes games more fun. Nicole discovered the Four Keys to Fun, a model used by hundreds of thousands of game developers worldwide, in 2004. She used this model to design the iPhone's first accelerometer game, now called Tilt World, in 2007. Nicole is referred to as one of the 100 most influential women in high-tech, one of the top 20 women working in video games, and one of the top 10 women in gamification. Nicole's work has been widely cited by global news media such as CNN, *Wired, Fast Company,* CNET, the *New York Times,* the *Wall Street Journal,* the *Hollywood Reporter, Red Herring,* and the *Boston Globe.* She has advised the White House and the US State Department on the use of games to unlock human potential to improve our world. For the past two decades, as the CEO of XEODesign, she has improved hundreds of millions of player experiences for companies such as Ubisoft, EA, Disney, and Cartoon Network; as well as worked on best-selling franchises such as Myst, Diner Dash, Pogo, and The Sims. One of the pioneers in applying game design outside of games, she designed game-inspired UI for Oracle, Cisco, Kaiser, Sun, Roxio, and others as early as 1992.

Tom Long is an award-winning game studies educator and game designer. He has worked in the game and simulation industry for more than 15 years. During those years, he has twice judged the Front Line Awards for *Game Developer* magazine, has won the Unity Great Education Giveaway Contest, and has contributed to many published game titles. In his spare time, he can be found at the local hacker space, FamiLAB, working on his Maker Faire project. Tom currently teaches Level Design II at Full Sail University in Winter Park, Florida. Follow him on Twitter at @tomlong74.

Michael Lucas was a gamer before the advent of video games; he played tournament Chess and Backgammon and worked to develop one of the best Chess-playing software systems in the world by 1979: DUCHESS (for Duke University Chess). As a National Master of Chess out of college, Michael continued in competitive gaming and worked in war-game simulations for the Navy. Michael later trained artists on *Monster House, Open Season, Beowulf, Surf's Up,* and many other films at Sony Pictures Imageworks and Sony Pictures Animation. His areas of expertise there were in hair and cloth simulation, Python and Maya scripting, and stereoscopic 3D. He has also worked in TV and on documentaries. He was the postproduction supervisor in 2010 on Mark Kistler's *Imagination Station,* and helped the show win an Emmy Award. He is now a course director at Full Sail University and teaches Level Design with UDK, Usability, and The Game Industry.

Dave Mark is the president and lead designer of Intrinsic Algorithm, an independent game development studio and AI consulting company in Omaha, Nebraska. He is the author of *Behavioral Mathematics for Game AI* (Course Technology PTR, 2009), has written for numerous other industry books and *Game Developer* magazine, and is a frequent speaker at industry conferences on the topics of game AI, game theory, and psychology. Additionally, he is the co-founder of the AI Game Programmers Guild and is the co-advisor of the GDC AI Summit. Dave continues to further his education by attending the University of Life. He has no plans to graduate any time soon.

Doug Oberndorf has a BA in Art with a focus in 2D & 3D Animation, and an MFA in Special Effects Animation. Doug is a course director at Full Sail University and teaches courses about the game industry. He owns the indie game development company Tropic Mods

Development and volunteers at events for the National Kidney Foundation (NKF).

Patricia Pizer debuted in the gaming industry at Infocom in 1988, where she made games back when you didn't even need graphics. Over the next decade, she worked at such studios as Boffo Games, THQ/GameFX, and Harmonix Music. Patricia moved into massively multiplayer online (MMO) games as Creative Director at Turbine Entertainment before she worked on MMOs at Ubisoft and Disney's VR Studio, the makers of Toontown. After applying her design skills to alternate reality games for 42 Entertainment, she returned to Disney Interactive Studios, where she designed DGamer, a DS online avatar SNA service, and worked on the Club Penguin Elite Penguin Force. Most recently, Patricia was Creative Director for ZeeGee Games. Mostly, though, Patricia just likes to play games.

Michael Pynn is a creative writing instructor at the college level as well as a game designer. He has taught numerous subjects relevant to students of game design, including composition, professional communication, narrative development, analog game design, game documentation, and more. As a designer, Michael has produced multiple alternate reality games and interactive story experiences. A member of the International Game Developers Association, he holds a BA from the University of Central Florida and an MFA in Creative Writing for Entertainment Business from Full Sail University.

Brian Stabile is a game designer/programmer, musician, audio engineer, teacher, and linguist. In addition to teaching online classes at Full Sail University, Brian is the drummer for Yogurt Smoothness, a nationally touring indie rock band, and co-owner of Astro Crow, an independent game developer best known for their iOS title The Last Ace of Space. He worked as a programmer and Japanese translator for the PC title Three Dead Zed, and worked with the University of Central Florida's RETRO Lab programming "serious games" for clients such as the Robert Wood Johnson Foundation, the National Science Foundation, and the Defense Acquisition University.

Jason VandenBerghe has been making games for 17 years, and is currently a creative director at Ubisoft (most recently on Far Cry 3 and Ghost Recon: Future Soldier). In his career, he has been a programmer, producer, designer, and director. He has worked on bad games, good games, hard-to-make games, and, er, less hard-to-make games, and he intends to keep on doing that for as long as they will let him.

About the Illustrator

Raymond Yuen is an illustrator and designer from New York and is currently living in San Francisco. His love of comics inspired him to work through the caricature ranks at Six Flags Great Adventure and in the colorful offices of Pokémon USA. Currently, he is the cofounder and art director at Suspended Belief Studios, his game design company. You can find him at www.rayuen.com.

Index