UNIVERSITY OF CALIFORNIA

SANTA CRUZ

MOTIVATIONAL DESIGN PATTERNS

A dissertation submitted in partial satisfaction of the
requirements for the degree of

DOCTOR OF PHILOSOPHY

in

COMPUTER SCIENCE

by

Christopher Francis Lewis

June 2013

The Dissertation of Christopher Francis
Lewis
is approved:

_____

Prof. Jim Whitehead, Chair

_____

Assoc. Prof. Noah Wardrip-Fruin

_____

Prof. Steve Whittaker

_____

Tyrus Miller
Vice Provost and Dean of Graduate Studies

Contents

List of Figures

List of Tables

Abstract

Motivational Design Patterns

by

Christopher Francis Lewis

The rise of mobile platforms and the Web has created a market where software is now funded by advertisements, subscriptions and in-app purchases. This software is designed with retention in mind, *motivating* users to return to the application again and again.

However, developers struggle to understand exactly *how* to motivate users. This lack of understanding has given rise to the easy answers of 'gamification' and 'social,' without providing the foundational psychological knowledge developers need to truly understand their users' needs. New frameworks are required to bridge this knowledge gap.

This dissertation presents one such framework, a library of twenty-seven *motivational design patterns*, under the categories of gameful, social, interface and information. Theories and experimental results from motivational psychology, behavioral psychology and behavioral economics are used to explain the power of different design patterns and suggest optimal implementations. Additionally, a set of eight dark patterns are presented. These patterns promise the developer short-term gains, but at the expense of long-term motivational harm, and strategies to avoid their use are proposed.

To validate the applicability of the pattern library, it is used in case studies that analyze existing software designs, and generate new ones from problem statements.

The existence of the pattern library not only brings us greater understanding about how motivational software works, but provides us with a language with which to communicate motivational design, and a framework to begin the work on improving software's ability to meet users' motivational needs.

For Sara

Acknowledgments

To Jim Whitehead, who started as my mentor, became my colleague, and is now someone I am now proud to call my friend. His ability to listen to my tangled web of thoughts and figure out how to describe it in one startling sentence is unparalleled. His drive to get my work published was not just in service of my academic career, but always as a vote of support for me as well. I'm deeply thankful to have found an advisor who I worked so well with, and worked so hard for me.

To Noah Wardrip-Fruin, whose uncanny capability to make me excited about research—even when I was deep in the academic doldrums—provided momentum whenever it was needed. His willingness to share his time was a great gift.

To the entirety of the Games and Playable Media group at the University of California, Santa Cruz, a team of individuals that have come together to create a place of intelligence and vibrancy that I am privileged to have been part of.

To Marion and Martin Lewis, who have loved and supported me in whatever I do, even when it takes me far away from them. Their selflessness throughout my life has enabled me to be the person I am.

To Sara Lewis, to whom my dissertation is dedicated. She provides me with self-confidence. Ambition. Support. Perspective. Everything I have done in my career is as much a reflection of her effort as it is of mine.

Chapter 1

Introduction

1.1   Motivational design

We live in an era of computing that incorporates large scale user engagement. For example, in nine years, *Facebook* has built an audience of 1.11 billion monthly active users (*Facebook Reports First Quarter 2013 Results* 2013). That's one active user for every seven on the planet. We also live in a time where those users can be acquired extremely rapidly. In just forty-one days, Zynga's *CityVille* social game grew to 100 million active users (Gobry 2011). To put that number in perspective, only eleven countries have populations greater than 100 million people ("List of countries by population").

This explosive growth has been enabled by increasingly easy access to the Internet, combined with payment models that don't require users to pay any money upfront. But what is it about *certain* pieces of software that causes them to be so popular? How are such large and diverse audiences being encouraged to return to software? How can we build software that better motivates audiences? This dissertation analyzes this *motivational design.*[1]

---

[1] Motivational is not the only relevant term here. Others, such as BJ Fogg, use the term "persuasive." The aim of this work is to meet motivational needs and provide something that users consciously or

1.2 The siren song of the smartphone

For a quick example of software that takes advantage of motivational design, take a look at your smartphone's application list. There are probably applications that offer the means to connect with others, like email or *Facebook*. Likely there are some news applications so you can make sure no event has passed you by today. Maybe there are some games that you poke and prod at every once in a while. None of these applications seems terribly *important*—and we did live our lives quite happily before the iPhone came out—but we miss our smartphones when we accidentally leave them at home.

The draw of the smartphone is undeniable. A quick look around airports, supermarket queues, and coffee shops will show a number of people all doing the same thing, mooching around on their phones as time passes by. As Bogost (2012) puts it, "It's not abnormal. It's just what people do. Like smoking in 1965, it's just life." Smartphones are wonderfully *immediate*. When we need to reach out to someone, they're only a couple of taps away. When we feel curious, *Reddit* always has something new. When we want to compete, *Hero Academy* lets us fight our friends.

Smartphones are motivational at their core. They provide us with easy access to many things we fundamentally desire. If you want to find more examples of motivational software when reading this dissertation, just reach into your pocket and pull out your smartphone.

1.3 The zero-sum game

Building for retention is important to software companies as software increasingly moves to the Web and mobile, where users have come to expect immediate satisfac-

---

unconsciously really want. "Persuasion" has connotations with salesmanship and propaganda; trying to convince people to do something that they may not really want to.

tion without large upfront costs. Successful companies like Google, Netflix and Zynga look to advertising, subscription models or in-app purchases. These revenue models hinge on keeping users engaged and happy, so they continue returning to click on advertisements, renew their subscriptions, or purchase more in-app goods. Even software that was once sold only in boxes, such as *Microsoft Office* and *Adobe Photoshop*, that was once sold only in boxes, are moving to models that allow users to subscribe to access.

The maxim of time being money has never been so true for software. And if time is money, then there is money to be made. *Social Media Report 2012* (2012) found that the total minutes dedicated to software across the United States increased 21% between 2011 and 2012. Software is becoming better at holding our collective gaze. However, there is evidence that our *per-person* minutes are inflexible: we only changed from 37 to 39 minutes a day using our smartphones in the same 2011 to 2012 period (*State of the Appnation – A Year of Change and Growth in U.S. Smartphones*). In this attention economy, companies fiercely battle for our time. Once the population transitions fully to a connected existence, that economy becomes a "zero-sum game," and any time we dedicate to one piece of software will be at the expense of another. The importance of understanding motivational design is not just an academic exercise, but at the heart of the software industry as we know it today.

1.4   Research questions

For an issue so important to the software industry, there is little literature about the underpinnings of software retention. Scholars such as Ian Bogost, Sebastian Deturding and Juho Hamari have taken aim from viewpoints of game scholarship, motivation theory and behavioral economics. These diverse fields are all dedicated to finding out what drives people to make certain decisions and participate in certain actions. The

theoretical and experimental findings from these areas can be utilized to begin shaping a new way of developing software; one that changes from a process-oriented model, obsessing over how quickly users can accomplish tasks, to a motivation-oriented model, looking at how software can meet the motivational needs of users and create long-lasting relationships with them.

One means of finding the commonalities between different designs is to use *pattern languages.* No design is an island, and designers borrow and evolve ideas from one another. Pattern languages provide a means of identifying these common ideas, allowing us to name and describe them. For designers and academics alike, such patterns would provide provide a framework for software analysis, and a toolbox with which to design new motivational software.

However, motivational software thus far has been designed experientially; instinct and intuition are used in the hope that the design will eventually prove "sticky." We have no psychological foundation to even identify which patterns are actually motivating, and which are simply common user interface elements. Before we can even begin finding patterns, we must first know what we need to be looking for.

To this end, this dissertation will answer two questions:

1. What theories of motivation help explain why users choose to engage with software?

2. Using these theories, what are the design patterns present in software that help to motivate user behavior?

1.5   Motivational design patterns

The result of investigating these questions is a library of *motivational design patterns*.[2] These patterns describe common aspects of software that fulfill basic desires within all of us, creating an intrinsic motivation—a drive to perform a task because we find it inherently rewarding—to engage with a piece of software. The creation of a library of these patterns provides us with a language to describe the similar designs that exist across different pieces of software, and theories from motivation theory, behavioral psychology, and behavioral economics are used to describe their motivational power. These theories also help explain whether certain pattern usages are effective, and what can be done to improve poor uses of a pattern.

A short description of each pattern in the library follows. Each of these patterns are described in more detail in Chapters 4 to 7.

1.5.1   Gameful patterns

These patterns have the qualities of gaming, and focus on quick feedback loops.

Score Points awarded in response to actions.

Leaderboard Placing the user in a ranked list of others.

Increased Responsibility Trusted users can perform more influential actions.

Collection The ability to collect virtual items.

Specialization: Badge An indicator of reaching a certain goal.

Growth Ownership of something that grows over time.

---

[2]They fall under the umbrella of interaction design, providing "the *why*, as well as the *how* of our daily interactions using computers" (emphasis added, Thackara 2001).

1.5.2 Social patterns

These patterns help users fulfill their *Social Contact* needs.

Contact List A list of contacts.

Identifiable Community An area where a community can come together.

> **Specialization: Meta-Area** A community for discussing a particular product.

Broadcast A means for a user to share information with others.

> **Specialization: Social Feedback** A means for a user to send feedback about a Broadcast.

Activity Stream A series of Broadcasts grouped together.

Identity Shaping A means for users to customize their identity.

Item Sharing A mechanism that allows users to share items with one another.

1.5.3 Interface patterns

Interface patterns pertain to how applications communicate to the user through the interface.

Praise Users are rewarded for performing actions.

Predictable Results Actions taken should have predictable outcomes.

State Preservation Applications can be exited at any time.

Undo Actions can be reverted.

Notifications Alerts of some state change.

1.5.4 Information patterns

The patterns guide users through content, often satisfying their *Curiosity* needs.

Customization Users can customize their virtual space.

Specialization: Filters Content can be highlighted or hidden.

Personalization The system modifies itself to the perceived needs of the user.

Reporting Content users deem unacceptable can be reported.

Search A means for users to search for content.

Organization of Information Information can be organized for later retrieval.

Intriguing Branches Interesting content is linked together.

Task Queue A list of tasks to be performed.

1.6 Motivational dark patterns

In addition to the general pattern library, eight motivational dark patterns are presented. These patterns harm a user's intrinsic motivation, by controlling them and instilling fear in the long-term. These patterns can be countered by users who develop their manipulation literacy, a new idea contributed by this dissertation. By increasing their ability to recognize when they are being manipulated, these users can make informed decisions about whether they want to enter into a manipulative software environment or not. The eight dark patterns patterns are listed below, and are described in more detail in Chapters 9 to 11.

1.6.1 Temporal dark patterns

Temporal dark patterns occur when users are unable to correctly estimate how much time they will interact with an application. This can happen when the application requests *too much* time from the user, or when the application offers *too little*.

Grind Repetition of a skill-less task in order to progress.

Interaction by Demand Forcing users to engage with the application on its schedule.

Hellbroadcast FILTERING a user's BROADCASTS without consent.

1.6.2 Monetary dark patterns

These patterns cause users to either lose track or regret spending money, creating a short-term gain for the company, but resulting in long-term loss of motivation in their audience.

Currency Confusion Substitution of money for an arbitrary currency.

Monetized Rivalries Exploiting user competitiveness to incentivize purchasing.

Pay to Skip Users can pay money to skip onerous issues.

1.6.3 Social capital dark patterns

Social capital dark patterns exploit a user's social network, putting her friendships at risk.

Social Pyramid Schemes A requirement for other people to be brought in to the application before it is interesting.

Impersonation Creating BROADCASTS that appear to be from the user, but are in fact generated by the application.

1.7   Why the library matters

This library provides the foundational work in motivational design. The library creates a language to discuss motivational design, which has three key benefits:

1. Common techniques are stored in an accessible format that can be communicated to others.

2. These patterns are then linked with psychological theories to understand *why* a certain pattern may work.

3. The insight provided by the psychological theories allows designers to evolve and improve patterns over time.

The library allows designers to move from an *experiential* form of development to a more *experimental* approach, mixing and matching different patterns to create different user experiences. Such an approach combines traditional designer intuition with more informed scientific reasoning to help find a more engaging design for users. To show how this approach works, four case studies are presented. Two of them use the pattern library to *analyze* current designs from *Khan Academy* and *Tiny Tower* with a view to improving them. The other two case studies use the pattern library to *generate* new designs from problem statements. The first creates a web site centered around voting, and the second aims to encourage participation with a crowdsourced science video game.

Designers that use the pattern library are empowered to create more meaningful, engaging, and long-term experiences. This is what it means to develop motivational software. Let's see what we can build.

1.8   Disclosures

In this dissertation, judgment will be made as to the quality of implementations of patterns, and ethical issues will also be addressed. While the intention is to be unbiased, it is nonetheless important that potential conflicts of interest are disclosed:

- I was employed as an intern with Google in 2011 and 2012, and at the time of writing had a signed contract to rejoin the company full-time.

- I acted as independent consultant for WayIn between 2012 and 2013.

Chapter 2

Psychology of motivation

2.1 Overview

Before we can begin to identify and analyze motivational design patterns, we must first understand what motivation is at all. This is often the missing link in many discussions surrounding product development, which seem to assume that: "If you build it, they will come." This is understandable. Psychology is a wide and varied space, full of argument and contradiction, without the certainties of input/output that engineers rely upon. Humans are abstract, diverse and irrational. Why should we bother looping in psychology at all? Why not just focus test, see what users like and don't like, and move forward?

The problem with this approach is that it leads to "cargo cult" design. The cargo cult term comes from cults in small, pre-industrial tribes in the Pacific. These tribes were exposed to cargo coming from Western societies, but eventually the cargo would cease to arrive as the Westerners left. As the tribes didn't understand where the cargo came from, they turned to rituals to try and recreate the *conditions* at which the cargo arrived. In the case of World War II, they built faux-airstrips and radio equipment.

We see this exact same behavior when we hear terms like "we need to make it more social" or "let's add a gamification layer." The designers are trying to recreate the conditions that provided success for others, without understanding the core psychological foundations of what drove that success in the first place. We need to understand why people are motivated to engage with a software product so that we can make the right choices about what to add, what to leave out, and be able to identify what is missing. Otherwise, we remain in a cargo cult state, attempting to replicate only what we have seen, without any knowledge that what we are doing will result in the right outcome.

To gain the required understanding, we'll need to go on a whirlwind tour through three key subjects: behavioral psychology, intrinsic motivation theory, and behavioral economics. Concepts from each of them will be used: variable ratio schedules from behavioral psychology, Reiss desires from intrinsic motivation theory, and various experimental studies from behavioral economics. While this dissertation is perhaps somewhat cavalier at mixing and matching these fields, it's important to note that they are distinct with their own theories and experimental results. This dissertation, by and large, only contains theories that have experimental data to support them, minimizing any negative impact from conflating fields together in this way.[1] What we are interested in as software designers is not so much the theory of mind, but more the ability to make an informed guess as to how a user may respond to a certain pattern. Experimental data helps us with this greatly, even if there is disagreement about the *why* a certain experimental result is what it is. This pragmatic approach lets us get at the core ideas of how we might design software, without getting lost in theoretical frameworks that

---

[1]Some readers may wonder why research from game designers, such as Lazzaro (2004); LeBlanc (2009), does not receive attention in this section. The goal in this section is to introduce psychological concepts that help to explain intrinsic motivation at a *fundamental* level, so that such knowledge can be generalized across all kinds of software, without gameful contexts being suggested as required to create intrinsically motivating software. This psychology research then helps to *explain* the insights that game designers have shared. Literature from game designers will be used throughout this dissertation, and their exclusion here should not be taken to imply that their work is not useful.

could contradict one another.

2.2 Behavioral psychology

Behavioral psychology is a perspective that organism behaviors occur as responses to stimuli. Certain *stimuli* (inputs) are introduced to the body, and certain *responses* (outputs) occur. If you are poked with a stick as the stimulus, you'll probably yelp as a response. This is a fundamentally *extrinsic* view of our motivation: we modify our behaviors in reaction to our environment. Behaviorism rejects the idea that we can search inside our minds—a process known as introspection—to try and get at the fundamentals of the human psyche.

Behaviorism has a past that stems through Russian objective psychology (most of us know of Pavlov and his salivating dog) but the behaviorism term is usually attached to John Watson (Hergenhahn 2001, p. 337). In Watson (1919, p. 10) he writes:

> The goal of psychological study is the ascertaining of such data and laws that, given the stimulus, psychology can predict what the response will be; or, on the other hand, given the response, it can specify the nature of the effective stimulus.

The most famous of the behavioral psychologists was a man named B. F. Skinner[2] who was named the most-eminent psychologist of the 20th century by Haggbloom et al. (2002), and it's his work that we'll concern ourselves with here. He held a particularly functional view of behaviorism: that we are simply a product of environmental and behavioral stimuli. He thought that mental events were simply labels we gave to particular bodily processes, and he didn't acknowledge mental events in human conduct. This made him a "radical behaviorist," and he believed that simply manipulating the environment of a subject and noting the result was all that was required for study (Hergenhahn 2001, p. 391–393).

---

[2]The B and F standing for Burrhus and Frederic respectively.

To this end, he's most well-known for the "Operant Conditioning Chamber," which is now often referred to as a "Skinner box" (Chiesa 1994). Using it, he would study *operant conditioning*, looking at how reinforcement or punishment led to the increase or decrease of certain voluntary behaviors. Operant conditioning can be characterized as "learning how to operate effectively in the environment... engaging in those behaviors that produces attractive consequences (e.g. gaining approval, earning money) and also in those behaviors that prevent unattractive consequences (e.g. being rejected, getting fired)" (Reeve 2005, p. 135). This description leads into a simple equation (Reeve 2005, p. 135):

$$Situational\ Cue : Response \rightarrow Consequence$$

The Situational Cue is the environment. It sets the stage for a response, but does not initiate it. The voluntary response from the organism creates the consequence. For example, if playing a game of football (soccer), the game creates the situation. The response to kick the ball into the goal results in the consequence of everyone on the team cheering. Being involved in the game doesn't force a player to kick the ball, it just creates the environment where certain actions result in certain consequences. If we remove the game, kicking the ball into the goal wouldn't result in the cheering consequence. We can see the same thing in computational environments, where a situation is provided in the user interface, and the user presses a button with the hope that she knows what the consequence will be.

Skinner constructed the Skinner box to perform tests on rats and pigeons, to see how they would respond to certain stimuli. The most well-known use is a lever which a rat can pull to get food, leading to the rats to pull the lever more often to get food. The food is called a reinforcer: it's an extrinsic motivation that increases behavior. Skinner then started to play with when food was produced. Sometimes the food would come out on every pull. Sometimes it would come out on every tenth pull.

14

Sometimes it would come out on average one out of ten pulls. Sometimes it would come out only after a certain amount of time had passed.

As the rat realized the more it pulled, the more food came out (the behavior of pulling the lever was reinforced with the food reward), so the rat begins to pull the lever more often. If, instead, a punishment (like a loud noise) occurs when the rat pulled the lever, it would reduce pulling. By affecting the environment that he controlled, Skinner found he could condition the voluntary actions of the rats and pigeons. Controlling voluntary actions is what we are concerned with when we speak about motivational design: we want the user to voluntarily interact with our application.

The exact set up for how and when rewards are offered is known as a schedule, and one schedule in particular will be referenced later on: the variable ratio schedule. The various reward schedules are (examples provided by Wikipedia `http://en.wikipedia.org/wiki/Reinforcement`):

Fixed Ratio Deliver the reinforcement after every nth response. A coffee card that gives a free coffee after nine cups would fit under this heading.

Variable Ratio The reinforcement is delivered *on average*, after n responses. This is the classic "slot machine" schedule used by one-armed bandits the world over.

Fixed Interval Reinforce after n period of time. A washing machine runs on a fixed interval, providing clean clothes after a certain period of time.

Variable Interval Reinforce to provide an average interval of n time. Fishing is a good example of this: a fisherman might get a fish after just one minute, or he might have to wait an hour to get a bite.

The variable ratio schedule creates the most response over time (Chance 1998). If one wants to create an addictive experience, the variable ratio is the one to choose.

Unsurprisingly, the science around this schedule has been honed to a fine level of specificity by the casino industry.[3]

While the experimental observations that Skinner made remain valid, his ideas have been largely supplanted by cognitive psychology, which brings back mental events. This was due to research indicating that the way we learn differs between us and animals, between children and adults, and may even have genetic roots (Hergenhahn 2001, p. 396). Cognitive psychology accepts the brain has information which it organizes and processes, and uses that information to make decisions (Sternberg, Mio, and Mio 2009). However, like behaviorism, it uses the scientific method and observable states, and rejects introspection. It's also important to note that the experimental results of behaviorism aren't refuted, only the explanations change. For example, rats still pull the lever the way Skinner found them to. But, whereas a behaviorist would say the rats had learnt to press the bar (the behavior), cognitive psychologists would say the rats had learnt that bar presses deliver food (Navarick 2002).[4]

To get a better idea as to why behaviorism is no longer as popular, we can look to Kohn (1999). Kohn is particularly critical of Skinner, as the reductionism in Skinner's argument leaves no room for free will. Kohn (1999, p. 8) writes about a visit of Skinner to deliver a lecture to him:

> I couldn't resist a bit of flippant humor. "We certainly want to thank the environmental contingencies responsible for you being here this afternoon," I said. [Skinner] didn't laugh. Smiling courteously, he replied, "I'm very glad they occurred."
> Skinner believed that he had "chosen" to visit my class—that all of us "choose" our actions—about as much as a rock in an avalanche chooses to land.

At this point, you're probably wondering why so much time has been spent on behaviorism when only variable ratio schedules will be used, especially when the concept

---

[3]For example, a 12:1 ratio of near misses is thought of as being the most entertaining (Harrigan et al. 2010).

[4]And in fact, as described in Navarick (2002), studies have shown that, depending on context, rats can learn from a behavioral or cognitivist view, so both schools of thought are valid.

has been largely rejected as helping us understand why humans do what they do. The reason it is brought up is because gamification, and its corresponding touchstone book, *Gamification by Design: Implementing Game Mechanics in Web and Mobile Apps*, has a strong current of behaviorism throughout. Many of the patterns listed there revolve around rewards—like score, leaderboards, badges—and present them as motivating in and of themselves. As Radoff (2011a, p. xxxii) puts it, "these are simply the tools of game design; they don't tell you what makes games work."[5] Radoff (2011b) then expands on this issue:

> The problem with gamification isn't the term, or its objectives, but how it is applied...It's the behaviorist approach to games that channels inquiry away from the harder problems of immersion, cooperation and competition that is so important to creating successful game experiences. Behaviorism was popular in psychology because it seemed to offer some easy answers - some of which do work (such as certain forms of conditioning) yet which is built on an erroneously reductive premise that ultimately failed to be supported empirically.

As practitioners of motivational design, we must be on our guard for easy answers like behaviorism. As appealing as the input/output paradigm that it's built on might be to software designers, it's simply not something that can be supported much farther than the experimental data. It misses the depth of experience that we need to create in order to have the long-lasting, *meaningful* attachment that products need to compete for users' time. Gamification erroneously uses those easy answers from behaviorism and then in turn presents them as the easy answers for how to increase

---

[5]This is also addressed by Kelly (2012): "Gamifiers often reach for levels, badges, experience points and achievements because those are ludemes with which their clients are familiar. The clients are often not gamers, but they may have tried something like *FarmVille* or read a book like *Reality Is Broken*, and encountered them as terms. Their assumption is frequently that these terms hold some universality of truth, but actually they don't.

"Rather, familiarity breeds contempt. Remembering that users are not naive and likely use more services than just yours, it's likely that they will have encountered many of these same ludemes as you. If they see they have a level of 1 and 0 experience points, chances are that this induces a feeling of grind. Having ground their way through one or more games with the same ludemes, they don't want to do that all over again."

user retention.[6]

2.3 Intrinsic motivation theories

2.3.1 Overview

Intrinsic motivation is, in essence, the opposite of extrinsic motivation. Intrinsic motivation comes from within, whereas extrinsic comes from without. It's what motivates us to do things only for the joy of doing it, and we do them even if there are no environmental reasons to do so. It's what pulls us to play another hour of *Halo* instead of write essays for a class, even though we might be paying large amounts of money to attend that class. Reeve (2005, p. 135) provides a good way of spotting intrinsic motivation:

> When people engage in tasks and feel competent and self-determining, they express their intrinsic motivation by saying "That's interesting," "That's fun," or "I enjoy doing that." For instance, interest and feeling free can spark the desire to read a book, and enjoyment and feeling competent can involve a person in a challenging crossword puzzle for hours.

When we think of trigger words like "interesting" or "fun," we're thinking of intrinsic motivation. When we engage in a task even when our environment encourages us not to (such as surfing *Reddit* on work hours, at the risk of losing our job), we're engaging in an intrinsically motivating task. But what happens if there is a person who likes to paint, and is then paid to do so? It's indeed possible for someone to be both intrinsically and extrinsically motivated at the same time, and it's indeed possible that

---

[6]This point comes up again and again from gaming scholars. Schell (2011) used the thought experiment of "chocofication" to aptly prove this point. He shows how chocolate tastes great, and how it makes other food like ice cream taste even better. But it doesn't work all the time; adding chocolate to hot dogs is a disaster. Chocofication, like gamification, is not a silver bullet. Wardrip-Fruin (2012) "I think the heart of games is play. I think a lot of people who talk about gamification think the heart of games is points and levels and badges...I'm interested in gamification if we rethink the core activity, we think about 'playification.'" Bogost (2011) suggests that even the naming is designed to offer an easy answer. "This rhetorical power derives from the '-ification' rather than from the 'game'. -ification involves simple, repeatable, proven techniques or devices: you can purify, beautify, falsify, terrify, and so forth. -ification is always easy and repeatable, and it's usually bullshit."

performing the same task looks exactly the same whether intrinsically or extrinsically motivated (Reeve 2005, p. 135). The important thing is finding the real driver: whether the person is satisfying psychological needs, or whether they're looking for incentives contingent on the behavior.

In this section, three different researchers on intrinsic motivation will be presented. Malone and Deci & Ryan are largely complementary researchers, whereas Reiss has a separate view of intrinsic motivation. The researchers are presented in chronological order: Malone, Deci & Ryan, and finally Reiss.

2.3.2 Malone

Malone was certainly not the first person to start formulating an understanding of intrinsic motivation, as he himself cites work from the 50s and 60s in "Toward a theory of intrinsically motivating instruction." However, he was the first to look at the issue of intrinsic motivation and software. In Malone (1981), he recognizes playing video games as an intrinsically motivating activity, and tries to pick apart what makes games captivating using a version of *Breakout* that he created. It's important to note the year here: *Pong* had come out just eight years before he first presented the results of his experiment in 1980. Eight years sounds like a long time, but to go from the introduction of a new medium, to recognizing its strong motivational potential, and then publishing experimental results about it, is remarkably fast in academia.

Malone defines an activity as intrinsically motivating "if people engage in it *for its own sake*... [using] the words *fun, interesting, captivating, enjoyable* and *intrinsically motivating* all more or less interchangably." (Malone and Lepper 1987)

In Malone (1981), he first surveys a group of elementary school children on a series of different games, such as *Breakout*, *Snake* and a pinball game called *Petball*. He asks which games they prefer, and then correlates the particular features of the games to preferences. He first notes that "it is clear that there are big differences between people

19

FEATURE	CORRELATION WITH AVERAGE PREFERENCE
Goal	.65
Computer keeps a score	.56
Audio effects	.51
Randomness involved in game	.48
Speed of answers counts	.36

Table 2.1: A table showing the top five features that determine game preferences, according to Malone (1981).

in the kinds of games they like." This finding is not that surprising, but helps underline that motivation is different for different people. However, he does find that certain features seem to appeal to the children, and the top five are reproduced in Table 2.1. Here we see that a goal is clearly important, and goals provide a way of calibrating challenge (with no offered or derived goal, there is no challenge by default).

Malone then creates several versions of *Breakout*, where he tried the different combinations of having brick breaking, the ball bouncing from the paddle, and showing the score on screen. Players significantly preferred the version which featured all three (an average rating of 4.8 out of 5), with the second place being the version with brick breaking and ball bouncing, while omitting the score (4.1). The other versions drastically dropped off, scoring between 3.3 and 1.4. Brick breaking is clearly the most important of the features, scoring a .77 correlation, whereas a score was only .32, and the bouncing from the paddle .30. Malone theorizes that the brick breaking offers clear feedback to the player, and so provides "a goal, a visual effect, fantasy [Malone uses fantasy to describe a relatable context for the game to take place in], and scoring all at the same time."

Of interest here is that the highest rated version of the game is the original, which contains all the features. Malone doesn't offer any theory about this finding, but it seems to indicate that the design of *Breakout* is a gestalt artifact, and splitting

things up just makes the game worse. This leads to a possible conclusion that there is something about great games that are more than the sum of their parts. Every feature of *Breakout* contributes to the gameplay, even the score. The three features that Malone tested are all in service of the core gameplay loop of breaking bricks, bouncing off the paddle, and getting feedback that the player is doing well. Placing them all together is what makes *Breakout* a great game.[7]

Malone takes his thinking on intrinsic motivation further in Malone and Lepper (1987), where he presents a taxonomy of "intrinsically motivating instructional environments." The taxonomy is reproduced in Table 2.2. The taxonomy is interesting in multiple ways. Firstly, he focuses only on "learning environments." However, when we read through the list, does it not seem that all of the identified motivations seem applicable to all environments where we require motivation? It is likely not that Malone has defined his taxonomy too narrowly, but that our general understanding of when we are learning is not broad enough. When we think about learning, we imagine classrooms and lecture theaters. In fact, learning seems to be a core part of any motivational environment, be it classroom, workplace or home. When we're not learning, we're bored. As Koster (2004, p. 41–42) puts it in the context of video games: "With games, learning is the drug. . . When a game stops teaching us, we feel bored." He even expands this to situations where it isn't clear that we are learning: "When you feel a piece of music is repetitive or derivative, it grows boring because it presents no cognitive challenge. . . [the brain] craves new *data.*" When we take such a view as Koster, it's much easier to see that Malone's motivations are broadly applicable to any environment.

The findings of Malone gives us a strong first step in understanding the intrinsic appeal of not just video games and software, but motivating environments in general. We'll now look at Deci & Ryan, who collapse intrinsic motivation down to just three

---

[7]With the caveat that recent research has shown that adding secondary objectives can actually decrease motivation to play (Andersen et al. 2011). The key here is that all the features are not secondary to *Breakout*, but part of the essential experience.

Feature	Description
Individual Motivations	
Challenge	Optimal level of difficulty
Goals	Activity should provide goals, or environment for goal-setting
Uncertain Outcomes	Uncertainty from difficulty, levels, hidden information or randomness
Performance Feedback	Frequent, constructive, encouraging feedback
Self-esteem	Promote feelings of competence
Curiosity	Optimal level of informational complexity
Sensory Curiosity	Variability in audio and visual effects
Cognitive Curiosity	Create surprise and intrigue through paradoxes, incompleteness or simplifications
Control	Promote feelings of self-determination
Contingency	Responsive learning environment
Choice	Levels of choice over aspects of environment
Power	Activity should permit creation of powerful effects
Fantasy	Create motivation with fantasy involvement
Emotional Aspects	Should appeal to emotional needs, encourage identification with imagined characters or contexts
Cognitive Aspects	Appropriate metaphors for learning
Endogeneity	Fantasy should have an integral relationship to learning material
Interpersonal Motivations	
Cooperation	Appeal of activity may be enhanced by enlisting others to cooperate with
Competition	Appeal of activity may be enhanced by enlisting others to compete with
Recognition	Appeal of activity may be increased by social recognition

Table 2.2: A table showing the taxonomy of "Heuristics for Designing Intrinsically Motivating Instructional Environments" from Malone and Lepper (1987).

core ideas: Autonomy, Mastery and Relatedness.

2.3.3 Deci & Ryan

2.3.3.1 Self-Determination Theory

Self-determination theory (SDT) is a theory of motivation first introduced by Deci and Ryan (1985), which was subsequently popularized into mainstream thinking by Pink (2011). SDT defines just three core tenets that a task must have in order to be intrinsically motivating, as described by Ryan and Deci (2004, p. 7):

Autonomy The ability to make choices as you see fit; being the perceived origin of your behavior. This does not necessarily mean that you are independent (not relying on the help of others) or that the choice is not forced on you by someone else (you have autonomy if you feel the decision is correct). Autonomy also does not necessarily imply having a wealth of options, as long as the options available present the path you wish to follow. For example, first-person shooters don't offer many options. *Half-Life* doesn't offer you the chance to sit down and have a roundtable discussion about whether the aliens should end their invasion. However, it does offer the chance to dispatch them with a variety of weaponry, and this is the choice many gamers want anyway.[8]

Competence That the task at hand is something where you feel challenged, but is likely achievable. The challenge should be "optimal for [your] capacity," and allows us to grow our abilities and gain mastery of situations. This is essentially the inclusion of flow from flow theory, which is described in Fig. 2.1.

Relatedness That the task creates a feeling of connectedness to others, caring for them and them caring for you. Pink (2011) expands this notion slightly by renaming it

---

[8]For a discussion of games where little autonomy is perceived, see Sullivan (2012).

High

Anxiety　　Arousal　　Flow

Challenge level

Worry　　　　　　　　Control

Apathy　　Boredom　　Relaxation

Low

Low　　　Skill level　　　High
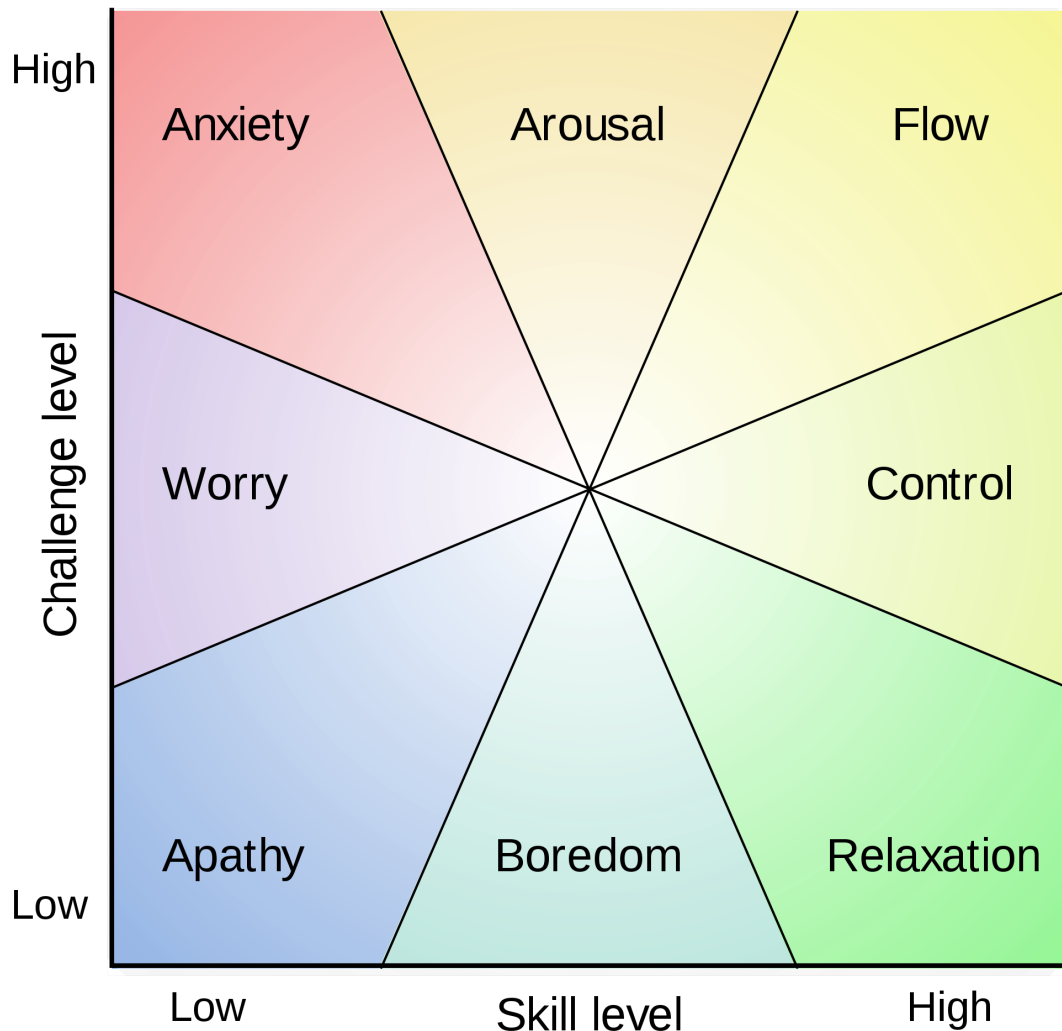
Figure 2.1: A chart showing the model of flow state, defined by Csíkszentmihályi (1997). Flow occurs when we feel "in the zone," where we lose track of time, totally focused on the fun and enjoyment of the challenge. When we take on a challenging task, with the skills to beat it, we feel flow. When the task is too easy, we are bored, when it's too hard we get anxious. Originally, flow was defined by a channel that ran from the bottom-left to the top-right (Csíkszentmihályi 1975). Anxiety or boredom would be felt going outside of the channel. However, this model was adapted to include apathy: there was no feeling of flow when a low challenge task and low skills were combined.

to "Purpose," that the task creates a meaningful change, that it leads to something bigger than just ourselves. Connecting with others is a purposeful task, so relatedness is a subset of purpose.

This, in a nutshell, is the entirety of SDT. It's intuitively believable, and we can imagine times in our lives, particularly in the world of work, where we felt that we had such things and were really motivated. We could do what needed to be done, the work was interesting and challenging, and the results provided something that felt important. But many of us have also had that job where our autonomy was thwarted at every turn, that the challenge was not there, and there was no purpose to what we were doing.

One other benefit of SDT is that its general broadness covers a wide spectrum of applications.[9] Unfortunately, this also makes it more difficult to apply with any granularity.

2.3.3.2 Cognitive Evaluation Theory

Cognitive evaluation theory (CET) is a subset of SDT that focuses on how extrinsic rewards affect intrinsic motivation, focusing just on the autonomy and competence aspects of SDT (Deci, Koestner, and Ryan 1999). A reward doesn't just have to be a trinket or food, it can be something as simple as being verbally praised. CET posits that when a feedback event occurs that we perceive as being *informational* of our mastery of something, we use this to satisfy our intrinsic need for competency. Without information on how we are doing, we have no basis of understanding on whether we're getting better at it. However, if the event is seen as *controlling* us, we lose our feelings of autonomy, and our intrinsic motivation drops.

This theory strikes at the heart of an ongoing and unresolved tension in the

---

[9]`http://selfdeterminationtheory.org/browse-publications` lists applications of SDT to areas such as education, health care, organizations, psychopathology, psychotherapy and sport.

motivational psychology community as to whether extrinsic rewards undermine intrinsic motivation. The classic supporting example given is Lepper, Greene, and Nisbett (1973). The authors performed an experiment with children who enjoyed to draw. The children were split into three groups: one group was told they would get a shiny gold star with a red ribbon if they drew a picture. The second group was given the star for drawing the picture, but were not told ahead of time they would get one. The third group were not made aware of the star, nor given one. The authors found that the group who had been told about the star beforehand drew less pictures independently afterwards. The other two groups showed no change. The theory is that the first group had succumbed to the overjustification effect: the children became focused on the extrinsic reward, and rationalized to themselves that they had drawn the picture for the reward, not for the joy of drawing the picture. They had overjustified the point of the extrinsic reward, and so their intrinsic motivation was hampered. It is worth noting that this doesn't occur when there is no intrinsic motivation to perform the task in the first place; paying a child to take out the trash doesn't undermine his intrinsic motivation to do it, as he had no motivation to take the trash out in the first place.

Once our intrinsic motivation is undermined, it doesn't come back, and we start to look for the extrinsic rewards every time. Even worse, prospect theory from behavioral economics (discussed in more detail in Section 2.4) indicates that we will set a reference point when we get used to those rewards coming in. They're no longer perceived as gains, they're just normal. To get the same motivational boost that the extrinsic reward was supposed to offer, the reward has to climb steadily higher. Analyzing the work of Suvorov (2003), Pink (2011) puts it thusly:

> [A constant reward] will quickly feel less like a bonus and more like the status quo—which then forces the principal to offer larger rewards to achieve the same effect.

Imagine when you first got a job: the pay check was spectacular in comparison

to the lower income you probably lived on before. Heading to work was a big deal, because there was that large check every month. But soon enough, the large check just seems like a normal check, and it's not motivating anymore.

However, voices of dissent argue that there is other experimental evidence showing that the phenomenon is limited to a small set of circumstances, and that other situations show no change or even an increase in interest when extrinsic rewards are used (Cameron 2001). Rather than taking any particular view on this issue, we'll settle for the possibility that the overjustification effect could exist and be important, and is worth considering when evaluating motivational designs.[10]

To better understand how a reward might be perceived, CET breaks down types of rewards into four categories (Ryan, Mims, and Koestner 1983):

Task-noncontingent Given whether the subject engages in the task or not. For example, one might give a reward just for participating in a study, even if the participant doesn't actually do the task that was required.

Task-contingent Given whether when the participant does engage with the task. This can then be broken into whether the reward is just for starting the task (engagement-contingent) or actually completing it (completion-contingent).

Performance-contingent Given when the subject performs well in the given task (such as being the top 80% of scorers in a game).

Task-noncontingent rewards are given regardless of whether the task is engaged in, so are thought to not be controlling. Engagement-contingent rewards are thought to be controlling, and undermine intrinsic motivation. Completion-contingent and performance-contingent are both thought to be even more controlling, but some of this is offset with the benefit of getting feedback about our competency at the task.

---

[10]It is worth mentioning that Cameron's rebuttal paper has 114 citations according to Google Scholar at the time of writing, whereas Deci, Koestner, and Ryan (1999) has 2449, which does give at least some metric as to which may be more widely accepted by psychology scholars.

To maximize the feedback benefit, and minimize the controlling aspects of external rewards, Deci, Koestner, and Ryan suggests that the rewards should be verbally delivered in a style that doesn't feel controlling, with comparisons with a baseline, rather than an arbitrary level. Thus, controlling feedback like "You should have got an A because that's what I expect of you," is more harmful to motivation than a softer "You got a B, and that's better than over half the class." Positive feedback was also found to enhance intrinsic motivation (no-one likes being told they're useless, and it undermines their feeling of competency).

The concern that CET raises is that the extrinsic rewards enshrined in gamification, and games in general[11], could undermine intrinsic motivation. For games, this is a worrying possibility that hasn't yet had enough research to justify any conclusions yet. For gamification, Zichermann and Cunningham (2011) don't really care, as long as it's factored into the design:

> One obvious conclusion of the intrinsic/extrinsic behavioral questions is that once you start giving someone a reward, you have to keep her in that reward loop forever. This consideration informs the total cost of ownership question for gamification and should be part of your calculations.

Another voice on this issue is Kelly (2012) who believes that some software can never be intrinsically motivating, and so even worrying about extrinsic rewards versus intrinsic motivation is moot.

> In the smaller scale of a coupon scheme, a social news site or trying to create a sticky application, [the intrinsic versus extrinsic debate in gamification is] total bullshit. Your service is not their life's work, and most of the time your gamifying efforts are never going to get anywhere close to that level of significance in their lives. They are in large part only motivated by the extrinsic quantity (coupons, prizes, etc) that you offer. So stop kidding yourself. Most of the time motivations are not hard to understand.

In response to Kelly, if you are involved in creating a piece of software that has so few redeeming features that the only motivation people can ever have is extrinsic,

---

[11]Raised by Hecker (2010)

perhaps the better course of action is to reevaluate the entire direction of the product. Software development, especially on the Web, is a wonderful industry that allows for pivoting to new directions quickly, without the sunk costs of things such as manufacturing. The agility afforded by software production should be used to create something people actually want.

2.3.3.3 Player Experience of Need Satisfaction

Importantly for our study of motivational software, SDT has been applied to video games via various experiments by Ryan and Rigby, in publications such as Ryan, Rigby, and Przybylski (2006); Rigby and Ryan (2011), building on the foundations of Malone. To do this, they introduce a new model which they call Player Experience of Need Satisfaction (PENS).

In Ryan, Rigby, and Przybylski (2006), PENS is created by extending SDT with the concept of *presence*; the sense that one is within the game, rather than acting upon it externally. Interestingly, Ryan, Rigby, and Przybylski relate this aspect to flow theory, whereas it has already been noted here that this appears to be part of the competency aspect of SDT. Flow state relies upon challenge and skill, and *Tetris* is a fine example of a game with flow state without any feeling of presence. Additionally, being *present* in the world is not necessarily tied to challenge, which is required for flow. For example, *Half-Life*, *Bioshock* and *Bioshock Infinite* all have phenomenal, enveloping, engrossing first opening hours, but none of them provide any significant challenge to the player during those introductory sections.

Another curious addition to SDT in the PENS model is separating out control schemes for particular study:

> [A] variable of interest to us in assessing need satisfaction in game play is the degree to which game controls are "intuitive;" that is, whether they make sense, are easily mastered, and do not interfere with one's sense of being in the game. We thus develop a measure of intuitive controls (IC) as

a subscale of the PENS that assesses the interface between the player and the action taking place within the game. Intuitive controls can contribute to game motivation because they are associated with a greater sense of freedom and control, and they enhance a sense of competence. Therefore, insofar as IC predicts motivational outcomes of games we expect it to be mediated by perceived autonomy and competence.

Here, Ryan, Rigby, and Przybylski frame controls as a barrier to the satisfaction of autonomy and competence. They are described as "the price of admission."[12] However, there are games where mastery of the control scheme is part and parcel of the experience of gaming competency. Fighting games are a prime example. The ability to pull off special moves by remembering and enacting combinations of inputs is limited to the best players, whereas those starting out will be limited to simple punches and kicks. Mastering the controls is a central part of the fighting game experience, so controls should not be separated out as being barriers to perceived competence, but included as part of the learning experience.[13,14] That said, games with unnecessarily terrible control schemes do indeed hinder players, but these could be classed as games that do poorly at providing the player with a sense of autonomy ("I pressed that button! He should have done x, not y! That wasn't what a chose to do! This game sucks!")

Ryan, Rigby, and Przybylski (2006) presents the results of four different studies that they undertook.

1. A survey of 88 undergraduates playing *Super Mario 64*, which indicates that intuitive controls help players experience high autonomy and competence, and that only competence is associated with presence.[15]

---

[12]This quote is supposedly taken from Rigby (2004), but this publication was not available to the public, so the quote is as used in Ryan, Rigby, and Przybylski (2006).

[13]In what could be seen as a reaction to this, the fighting game *Divekick* offers just two buttons: dive, and kick. The game is a parody of fighting games that include a dive kick mechanic, which is where a character jumps into the air, then dives down kicking the enemy. Of course, this move is completely impossible with real physics.

[14]The same could be said about apps in general. Learning how to effectively use *Microsoft Office* provides mastery feedback. Indeed, the existence of *Ribbon Hero*, a game to help users learn the interface, shows just how strong this connection is.

[15]This provides another data point for the argument that competence already provides the necessary

2. A group of 50 undergraduates were asked to play *The Legend of Zelda: Ocarina of Time* (a critically highly rated game) and *A Bug's Life* (a poorly received game by the critics). It was found that the preferred game, *Zelda*, offered higher feelings of autonomy and competence than *A Bug's Life*.

3. *Super Mario 64*, *Super Smash Brothers*, *Star Fox 64* and *San Francisco Rush*, all highly-rated games, were given to 58 undergraduates to assess their feelings of autonomy and competence. Again, the games that players enjoyed more were the ones that provided better feelings of autonomy and competence. Different players enjoyed different games because their personal feelings of autonomy and competence differed.

4. The final study involved surveying 730 Massively Multiplayer Online gamers, finding that autonomy, competency and the relatedness provided by the online environment all contributed to the enjoyment of the game, and that post-play mood was positively associated with autonomy and competence. They compared their results to Yee (2006), finding only modest correlations between their SDT results, and the theorized motivations from Yee.

All these studies seem to show that SDT and the PENS model broadly apply to video games. The only factor which isn't well researched in Ryan, Rigby, and Przybylski (2006) is the existence of relatedness in games. This is expanded upon by Rigby and Ryan (2011, p. 67):

> Although it is most natural to think about relatedness as dependent on two or more people, our research shows us how computer characters ("non-player characters" or "NPCs") can enhance our relatedness satisfactions when they interact with us in meaningful and supportive ways. For example, when NPCs offer us a well-timed compliment or smile, players feel important and worthwhile. Just as a joke shared with your waiter can change your mood, a warm compliment from a computer character can also make your

framework for the inclusion of flow, and that presence is superfluous to that feeling.

day, especially when it is specific to something we have actually done in the game.

The addition of this final puzzle piece provides all the context required to understand the siren song of video games: they provide the three factors of SDT in large quantities, especially Massively Multiplayer Online games like *World of Warcraft*. One thing that the authors do mention throughout the book is the idea that there are games that only focus on one of the three, such as *competence* in *Space Invaders*, with the given that *Space Invaders* provides all the required autonomy. However, it is possible to go further in the application of SDT to games than Rigby and Ryan do by looking outside of the game artifact. It can be argued that *Space Invaders* also provides relatedness, if you look at how the game was situated. *Space Invaders* was deployed to arcades, and it had leaderboards. The relatedness was part of the arcade culture, gamers talking to one another, trying to beat each others' scores. When relatedness isn't offered by the game, gamers will go and find it themselves. We see this today on *YouTube* and *Twitch* with people broadcasting their gameplay on live video streams, and chatting about games in forums all over the Web. This is not unique to games; there's a reason that we all discussed the previous night's episode of *Lost* around the water cooler. This also provides a clue as to why apps seem to be going to the Web and becoming "social," in order to provide the relatedness they previously lacked on disconnected clients. The relatedness the platform provides, even to applications that traditionally were missing such elements, provides the last powerful tenet of intrinsic motivation.

2.3.4 Reiss

Thus far, intrinsic motivation has been described as a single, unitary value. The theories of Malone and SDT indicate what may move the needle backwards and forwards on how much intrinsic motivation we have to do a task. Reiss rejects the

unitary version of intrinsic motivation, and instead proposes a multifaceted approach that takes into account different peoples' needs at different times (Reiss 2004; Reiss 2008; Reiss 2002).[16] He takes issue with the idea that there are certain tasks that are intrinsically enjoyable to people. Take hiking. He notes that even the most ardent hiker won't want to go out if they are tired, and suggests that the hiking itself is not the goal, but the satiation of the specific need to exercise.

His approach defines a theory of sixteen basic desires, which he links to evolutionary psychology. These are listed in Table 2.3, with possible sources of confusion cleared up in Table 2.4. He presents a number of studies to argue for the specific sixteen he classified, but there are too many to synthesize here, and interested readers should turn to Reiss (2004). He then proposes that these desires help explain different personality profiles (Reiss 2004):

> Although everybody embraces the sixteen basic desires, individuals prioritize them differently. Generally, the most important basic desires for explaining a person's behavior are those that are unusually strong or unusually weak compared with appropriate norms. For example, some people devote much of their time to satiating their desire for curiosity, others seek power, and still others are out for revenge. Those basic desires that are neither strong nor weak compared with appropriate norms are generally less important in explaining a person's behavior. The satiation of a basic desire is always temporary; soon after we satisfy a basic desire, the motive reasserts itself and needs to be satised again. After we socialize, for example, the desire for social contact may reassert itself within hours.

He notes that most people aim for a moderate amount of each desire, and that they are continuums. For example, a person with a low desire for *Social Contact* might be labelled as "private," and have her need for *Social Contact* satiated quickly, so always leaves parties early. In contrast, a socialite who has a high desire for *Social*

---

[16]Reiss was not the first to present a multifaceted approach. For example, Maslow's hierarchy of needs by Maslow takes a similar view, enumerating different levels of needs that we are motivated to fulfill (Maslow 1943). Reiss (2008, p. 97–98) directly addresses the similarity between his framework and that of Maslow's and notes that Maslow's theories were never experimentally tested, with Maslow himself stating that he had "not yet been able to think of a good way to put it to the test in the laboratory" (Maslow 1970, p. xii). Maslow's hierarchy is not considered here due to the lack of supporting data.

NAME	MOTIVE	ANIMAL BEHAVIOR	INTRINSIC FEELING
Power	Desire to influence, be a leader, dominate others (related to mastery)	Dominant animal eats more food	Efficacy
Curiosity	Desire of knowledge	Animal learns to find food more efficiently and avoid predators	Wonder
Independence	Desire to be autonomous	Motivates animal to leave the nest, search for food	Freedom
Status	Desire for social standing (includes attention)	Attention in nest leads to better feedings	Self-importance
Social contact	Desire for peer companionship (includes play)	Safety in numbers	Fun
Vengeance	Desire to get even (includes desire to compete, win)	Animal fights when threatened	Vindication
Honor	Desire to obey a traditional moral code	Animal runs back to herd to warn of predators	Loyalty
Idealism	Desire to improve society (includes altruism, justice)	Unclear	Compassion
Physical exercise	Desire to exercise muscles	Strong animals eat more are are less vulnerable	Vitality
Romance	Desire for sex (includes courting)	Reproduction essential for survival of the species	Lust
Family	Desire to raise own children	Protection of young facilitates survival	Love
Order	Desire to organize (including desire for ritual)	Cleanliness promotes good health	Stability
Eating	Desire to eat	Nutrition essential for survival	Satiation of hunger
Acceptance	Desire for approval	Unclear	Self-confidence
Tranquility	Desire to avoid anxiety, fear	Animal runs away from danger	Safe, relaxed
Saving	Desire to collect, value of frugality	Animal hoards food and other materials	Ownership

Table 2.3: A table showing Reiss' sixteen basic desires, reproduced from Reiss (2004).

First Desire	Second Desire	Difference	Example
Power	Status	People who are powerful might not desire social status, people who display high social status might not have much power.	Mark Zuckerberg is a powerful man, but wears a hoodie and sandals everywhere. Someone who buys an expensive car to show off might not have any power.
Honor	Idealism	People with high honor may do things that don't improve the world.	A soldier involved in a damaging war would have a high honor to his nation, but may not be improving society.
Social contact	Vengeance	Some people play for fun, some people play to win.	Competitive fathers who beat their children at sport play for vengeance motives instead of social contact motives.
Power	Vengeance	Powerful people don't always have to step on the throats of others to get ahead.	A leader of a charity organization is probably someone who enjoys power, but is unlikely to display high vengeance.

Table 2.4: A table showing the differences between similar-sounding Reiss desires.

Contact might always feel that parties end too soon.

One aspect of motivation that is not clearly delineated in Reiss' desires is the appreciation of artistic beauty, such as music or painting. Reiss takes a psychosexual approach to this issue, and places it under the *Romance* classification (Reiss 2002, p. 70):

> The appreciation of beauty—what is called aesthetics—falls under the desire for romance. This includes the desire for physical beauty, art, and music. Although Freud was among the earliest psychologists to call attention to the sexual aspects of art and music, the connection is obvious to anyone who watches MTV videos. The overwhelming majority of popular songs have romantic themes.

While some may find such a classification hard to swallow, this is presented here only as an explanation that beauty is indeed covered by the basic desires, and although the motivational patterns in this dissertation don't reference this desire, it remains available for use with any further expansion of the pattern library.

Throughout this dissertation, the sixteen desires (which will be referred to as

Reiss desires) will be the dominant framework for identifying and explaining motivational patterns. There are a number of reasons for this:

1. Reiss desires broadly subsumes SDT. Independence relates to autonomy, power relates to competence and social contact to relatedness.

2. Reiss desires are specific, which helps more easily pinpoint patterns.

3. Reiss desires cover patterns which are prevalent in motivational software, such as collecting, but were not adequately explained by SDT nor behavioral economics.

4. Defining Reiss desires as a continuum pinpoints why it is important to support as many desires as possible, but also that desires should be opt-in. Desires that are forced, such as the power fantasies that we see in video games, may provide strong mastery satisfaction, but they are not widely enjoyed across the entire population of gamers (girls, in particular). In games, this dovetails neatly with Juul (2009)'s concept of supporting different play styles to appeal to as many different demographics as possible.

This application of Reiss desires to software discards the desires of *Physical Exercise* and *Eating* from consideration. While there are specific applications which help satisfy these needs, say *Nike+* for exercise and *Yelp* for eating, these are specific genres and not generalizable to the genre-neutral patterns presented. Additionally, *Family* is also discarded. *Family* pertains to familial relations, and we cannot be related to anything in software. Reiss specifically does not include the idea of nurturing as part of the *Family* desire. He acknowledges that some are driven by such a need, but he does not believe all humans are (Reiss 2002). This is why he names the motivation *Family* and not *Nurturing*.[17]

---

[17]Interestingly, he doesn't believe pets satisfy *Family* requirements either, only children.

One word of note is that this dissertation will not treat *Acceptance* as being limited to something that can only be offered by other humans. As Rigby and Ryan (2011) showed, we can have relatedness needs met by virtual characters in game worlds. In addition, we've all experienced times when ourselves, or others, anthropomorphize computers, saying things like "it's thinking" (Turkle 1985, p. 110). It thus seems reasonable to extend this to a feeling of whether the *computer* accepts us or not. Does it accept the inputs we give it and show them to be valid, or does it throw errors immediately, telling us that we don't know how to use it correctly? This isn't just a case of mastery, but a case of feeling like the computer accepts us for our current level of mastery, and offer approval to boost our self-confidence. Patterns that help give us self-confidence will be labelled with the *Acceptance* Reiss desire.

2.4 Behavioral economics

Behavioral economics is the study of how various effects factor into the economic decisions we, or institutions, make. Now, an economic decision is not just "Do I buy this or do I buy that?" but any decision where some resource, be it time, money or something else, is gained or lost. For many decisions you make, there's an economic cost somewhere in it, so broadly speaking it's easier to think of behavioral economics as the study of decision-making (Ariely 2010).

The interesting thing about behavioral economics studies is that they often focus on human decisions that appear *irrational*. *Rationality* is the idea that agents (people, companies, sunflowers) work to maximize a particular utility function (happiness, money, sunlight). Humans do things that are sometimes not rational, and hence make *irrational* choices, leaving them with a net loss of resources (happiness, time and money being the obvious resources people value) (Lowenstein and Ubel 2010). If you've sat down to watch a movie you pre-ordered tickets for, even after you find all the reviews

are terrible, and then wondered why you still went, you've spotted yourself making an irrational decision. You lost time in order to mentally justify the lost money on the pre-ordered tickets. This particular phenomenon is known as the "sunk cost fallacy." Once we've locked a resource in that we can no longer extract, we want to make sure that that resource is not wasted. We see this when investors hold onto declining stock for too long, carnival-goers continuing to play expensive carnival games to win an inexpensive toy, or injured people going to the gym just to utilize the membership.

We'll see a number of these behavioral economics effects throughout the patterns, and they will be described when necessary. One important finding of behavioral economics, called prospect theory, underpins a number of theories, so is worth mentioning here.

In a nutshell, prospect theory is the idea that losses hurt more than gains feel good. Imagine there is a unit of happiness you can have. If you are given $10, let's say you might feel 5 units happier. If $10 is taken away from you, you'll be more upset, and maybe feel 8-10 units less happy. This has important implications for the way we make decisions about things we value. In general, we're *risk-averse*, as losses hurt. Take an example where a player is asked to gamble $1000, with a 10% chance of coming out with $20000, and a 90% chance of coming out with zero. Most people wouldn't take that bet, even though statistically the math works in favor of betting. The possibility of losing $1000, or going on a losing streak, is too risky. However, because those losses hurt, once people are down in a bet, they turn to *risk-seeking* to try and undo the losses. The classic image of the gambler—down on his luck, out of his money, and then going double or nothing, placing his wedding ring and car keys on the line—comes to mind. While comical, it is based in the reality of prospect theory. This is where the previously mentioned sunk cost fallacy may come from: our desperation to avoid loss leads us to make irrational decisions.

However, just like behaviorism, behavioral economics doesn't necessarily offer
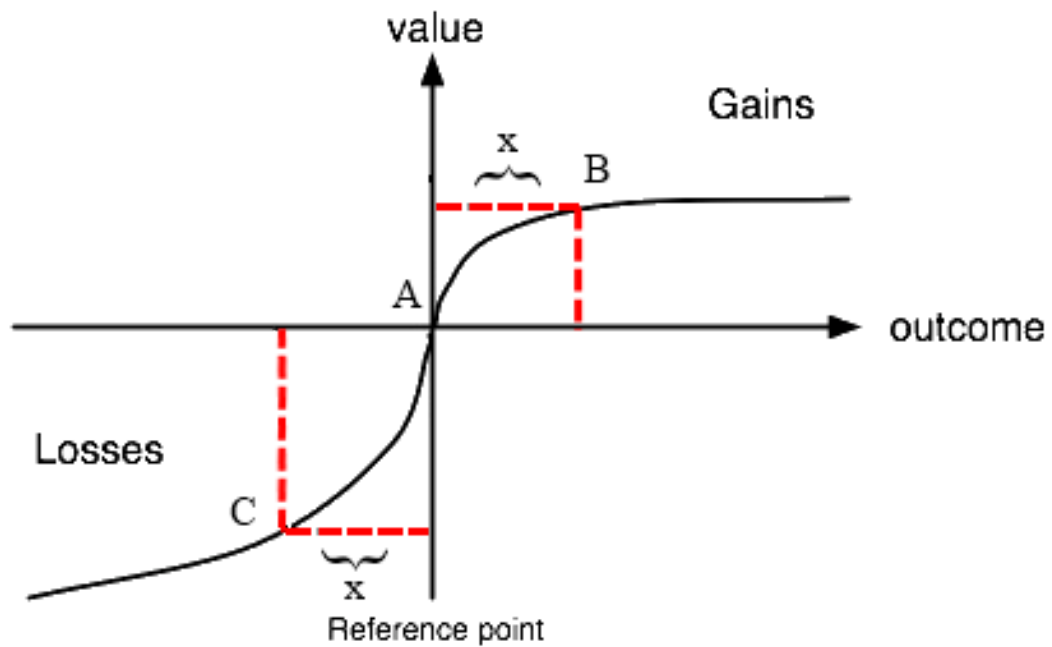
Figure 2.2: A plot of the prospect theory function. Note that the highlighted values for x along the x-axis is the same, but the perceived loss of value in the negative outcome is greater than the positive. *Used under Creative Commons Attribution-Share Alike 3.0 Unported lisence. Created by Marc Oliver Rieger* `http: // commons. wikimedia. org/ wiki/ File: ValunFunProspectTheory2. png`

easy answers. For the experimental data to be valid, it's important to recreate the conditions under which the decisions are taking place. While our rationality and irrationality might be predictable given the right environment, changes to the environment can alter how we think. This was found out first-hand, with an experiment called the "Starcraft Gambling Game."

2.4.1 The Starcraft Gambling Game

2.4.1.1 Hunting for behavioral economics

Behavioral economics appears in a number of gameplay patterns in Zynga's 'Ville-style games (Lewis, Wardrip-Fruin, and Whitehead 2012). This presents one immediate problem. Can it be shown scientifically that behavioral economics even applies to games at all? Behavioral economics usually uses money to validate theories, as money is inherently valuable to most people. Games don't use money, so how would players change? This wouldn't just affect games in isolation, but also general software. Later in this dissertation, we'll see gameful patterns that create a feeling of playing a game, and we'll see other patterns, such as customization, that have a strong undercurrent of prospect theory. Having an answer as to whether behavioral economics really applies, given the change of resource and environment, would be very useful.

2.4.1.2 Setting up the Starcraft Gambling Game

To validate whether behavioral economics could be applied to games, an experiment was devised to present prospect theory as a video game. A gambling game was set up, where players were given a certain number of turns. Each turn, they could choose to either gamble for a big payout and maybe take a loss, or use their turn for a smaller, guaranteed payout.[18] The probability for the payout vs loss wouldn't be

---

[18]This problem is actually a formulation of the *multiarmed bandit* problem. Imagine you have two slot machines. The first one you know will offer a certain amount of cumulative rewards over the pulls.

disclosed. Prospect theory hints that while people will be risk-averse to protect wins, while becoming *risk-seeking* to undo losses.

The population was split into those who self-identified as gamers, and those who didn't. It was hypothesized that one of three things would happen:

1. Gamers and non-gamers would both act the same way as prospect theory expects, being generally risk-averse unless they accumulated losses, in which case they would be risk-seeking.

2. Gamers would prove to be more risk-seeking than non-gamers, as they're conditioned by games to take risks in order to succeed.

3. Neither gamers nor non-gamers would follow prospect theory.

It was predicted the answer lay in the second option: gamers are often asked to take risks in games, and so it stands to reason when presented with a game they would be more likely to take that risk.

2.4.1.3 Starcraft Gambling Game implementation

The Starcraft Gambling Game was implemented using Javascript, so it could run in a Web browser. It used sprites from *Starcraft* in order to give the appearance of a real video game, which it was hoped would provoke players into treating the experiment like any other game. Before players were allowed to play, they filled out a brief survey about their background, where it was of most interest whether they self-identified as a "gamer" or not. This survey is listed in Appendix A. They were then shown the rules of the game, which were as follows:

The second one you don't know how much money could pay out. It could be better, it could be worse. There are multiple strategies you could use—stick with one or the other, explore the probability of the unknown machine and so on—and many mathematicians have had a look at this problem (Weber 1992). Gittins and Whittle (1979) notes that during World War II, Allied scientists spent so much time on the problem that it was proposed it be dropped over Germany so that German scientists could also waste their time trying to solve it.

Figure 2.3: The main screen of the Starcraft Gambling Game. The player's base in red is on the left, and the CPU's base is in blue on the right. The player can choose to mine contested crystals, which has the possibility of gaining or losing a tank, while mining the safe crystals confers no change. At the top of the screen is a list of how many tanks the player has, how many she has won or lost, and how many turns she has remaining.
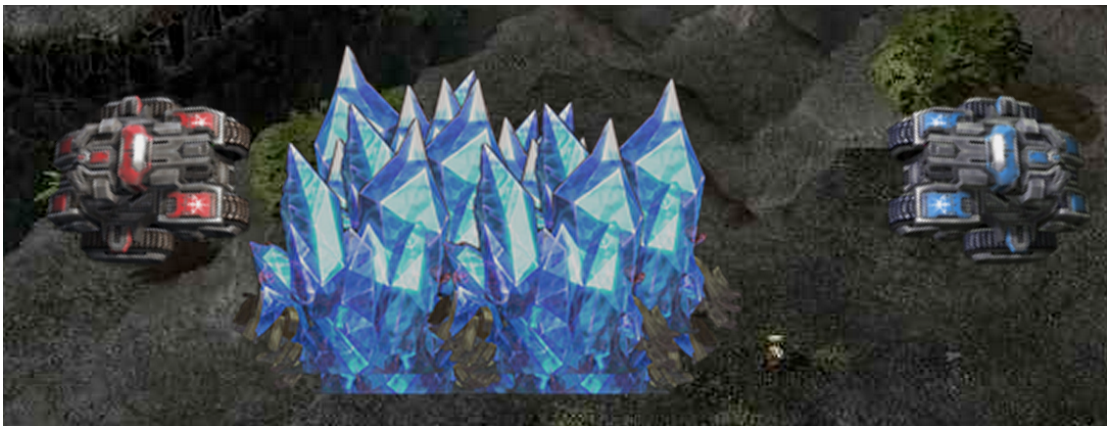
Figure 2.4: If the player chooses to head for the contested resources, the CPU player also sends out a tank to battle with the player.

Figure 2.5: If the player wins the battle (which is decided by probabilities), she wins a tank, and this message appears. If she loses, a message in red saying "Lose (-1 tank)" appears instead, and the player's red tank is destroyed.

- The goal of the game is to have as many tanks as possible by the end of the game. Crystals are gathered each turn. If enough crystals are gathered, a new tank is created.

- The player is given 15 tanks, and 15 turns (this means the player can never have negative tanks, so is always able to collect crystals).

- There are two buttons. The first button next to the large deposit at the top of the screen lets the player mine for enough crystals to create a tank, but the enemy will protect it and could destroy the player tank (shown in Figs. 2.4 to 2.5). The player's chance of winning the battle does not change through the game, and does not take into account how many tanks the player has.

- The second deposit will not create a tank, but is also not contested, so will never result in the player losing a tank.

- The player base requires crystals to remain operational, so the player must attempt to gather resources every turn.

In bold lettering, these rules were summed up as: "You must decide whether you will get a higher score by risking a tank for the bigger payoff, or whether you will get a higher score by keeping your tanks out of danger." This summation is archetypal of many of the decisions made in strategy gaming, and few games reward "turtling."[19] These rules both allowed the investigation of the prospect theory hypothesis, while at the same time presented players with reasonable facsimiles of the decisions that they might make in a real strategy game.

2.4.1.4 Starcraft Gambling Game results

The game was played by 167 people, who were recruited from the student population of University of California, Santa Cruz. 46 players identified themselves as non-gamers, and 121 identified themselves as gamers. Most players played online without supervision. 16 of those players played the game on a laptop with the experiment leader sitting next to them in order to write down feedback, but the instructions given were the same as those given to online players. The in-person playthroughs were offered cookies as an incentive. 88 of the online players were offered a small amount of extra credit in a class they were taking for playing the game. This leaves 63 playthroughs that were not incentivized at all. Players were requested to only play once, but no means of verifying whether players were unique was employed.

Table 2.5 shows the choices made by players throughout the game. We can see that, for the most part, players gambled more often than they would play safe. This means players were not playing as prospect theory predicted. We also see that there is no difference between gamers and non-gamers. Each population chose each choice in similar amounts.

However, prospect theory also predicts when players lose tanks, they'll switch

---

[19]Turtling is the practice of building up a base and focusing entirely on defending it, rather than the more risky play of attacking another player.

Is Gamer	Choice	Times Chosen
No	Gamble	410
No	Safe	280
	Ratio	1.46
Yes	Gamble	1050
Yes	Safe	723
	Ratio	1.45

Table 2.5: The choices taken by players of the Starcraft Gambling Game across all situations. Here we see there is only a negligible difference between the proportion of choices for the gamble and the safe option between gamers and non-gamers. We also see that players were more likely to gamble than they were to play safe.

Is Gamer	Is Winning	Choice	Times Chosen
No	No	Gamble	121
No	No	Safe	76
		Ratio	1.59
No	Yes	Gamble	289
No	Yes	Safe	204
		Ratio	1.42
Yes	No	Gamble	250
Yes	No	Safe	103
		Ratio	2.43
Yes	Yes	Gamble	800
Yes	Yes	Safe	620
		Ratio	1.29

Table 2.6: The choices taken by players of the Starcraft Gambling Game when the current game state is taken into account. "Winning" is defined as having 15 or more tanks.

to more risk-seeking strategies. Table 2.6 shows the same data when the number of tanks is taken into account. Here we see a marked difference in the way gamers played: they would be almost twice as likely to gamble when losing as they were before. Prospect theory also holds out here for the non-gamers, as we see a small bump towards risk-seeking for them as well, although it's possible this is simply a factor of the smaller non-gamer sample size.

However, even with these broken out numbers we see that players didn't broadly conform to prospect theory expectation. They were always more likely to gamble than they were to stick, and this opened up questions about the applicability of behavioral economics to games. The data was further analyzed to see if there was a satisfying explanation of these findings. Quotes taken from the participants were reviewed, and two comments, from two different non-gamer participants, stood out:

1. "What's the point of not taking the risk?"

2. "Go big or go home." (mentioned when the player was winning)

These are from the population expected to be more risk-averse, but instead they were more than happy to gamble. The first quotation, in particular, was odd. It was hoped that players would treat what they were seeing as a video game, and not taking the risk could have meant a higher score, which is something the gamers seemed to be using in their decision-making when deciding to gamble or not. Clearly, this commenter felt that the score was pointless, at least in the case of he/she not winning.

The experiment had run straight into the "peanuts effect."

2.4.1.5 The peanuts effect and creating value

The peanuts effect is a fairly straightforward idea, first put forth by Prelec and Loewenstein (1991) (much more background and experimental results are cited in Weber and Chapman (2005)). It describes how we are risk-seeking when faced with

46

minimal monetary gains (i.e. when playing for peanuts), we don't care about losing, so we may we well "go big or go home." Markowitz (1952) was the first to predict the effect. Given a choice between $0.10 for certain and a 10% chance of winning $1, it seems intuitive that most of us would take the gamble. But what if the stakes are raised to $10 and $100? How about $100 and $1000? The ratio between the guaranteed payout and the gamble are the same, but there is going to be a point for most of us where we flip around, and become risk-averse instead, and take the guaranteed money.

The Starcraft Gambling Game didn't just offer minimal money, it offered no money at all. Nor was there any prize or reward on offer for winning, which was a deliberate choice in order to keep the experiment as close to normal video game play as possible.[20] In hindsight, it seems obvious that players would simply gamble their way to victory, because a loss means nothing.

Once we understand how the peanuts effect is always lurking in the background, it's possible to hypothesize about why some social games or other freemium games make a lot of money, and some don't, even when they seem to use the same motivational design patterns. Perhaps the difference is in the game designer's ability to make the player *care* about what is at risk and what the player stands to gain. It's not enough to just have crops wither. The crops themselves must be of importance to the player for the withering mechanic to have any effect. Making players *care* is something that requires an artistic, creative hand to create something emotionally-resonant. Only then can the motivational patterns be introduced with a chance of working. This is the great failure of many a freemium game or gamification effort made with the cynical view of players as mentally malleable. While there are indeed ways to motivate certain behaviors, designers have to prove to their players that they should care at all, and that often is impossible if you are not interested or capable in creating a compelling experience.

---

[20]For most of us, no-one comes in with a comically large check and confetti when we win a round of *Halo.*

Chapter 3

Patterns

3.1 Pattern languages

To describe motivational design patterns, it is instructive to describe what patterns actually are. Some readers may already be familiar with design patterns from object-oriented programming, as popularized by Gamma et al. (1994).[1] Others may be entirely new to the idea, so it's worth a little discussion here.

Pattern languages were first introduced by Alexander (1977), who used them to describe architectural solutions at city, building, and construction levels. He described them thusly (Alexander 1977, p. x):

> Each pattern describes a problem which occurs over and over again in our environment, and then describes the core of the solution of that problem, in such a way that you can use this solution a million times over, without ever doing it the same way twice.

Each of Alexander patterns have a particular *problem*, and pair that immediately with a solution. Here's one picked at random from the book:

> *Pattern:* Stair seats
> *Problem:* Wherever there is action in a place, the spots which are the most

---

[1] A book which has become so influential in software engineering circles that the authors are referred to as the "Gang of Four," although they have never been seen pictured together as *Wild Bunch*-esque cowboys, which is disappointing.

inviting, are those high enough to give people a vantage point, and low enough to put them in action.

Solution: In any public place where people loiter, add a few steps at the edge where stairs come down or where there is a change of level. Make these raised areas immediately accessible from below, so that people may congregate and sit to watch the goings-on.

Alexander defined 253 of these patterns, but of course advocated that designers only use a subset of these patterns in any one project, the subset being a language in and of itself to tell the story of the artifact being created.

The beauty of pattern languages is not just that they provide an easy-reference cookbook for solutions to all sorts of problems, but that they are *languages* that provide a means of describing things that may have previously had no name. A lack of language means "not enough [can be] done to build on past discoveries, share concepts behind successes, and apply lessons learned in one domain or genre to another" (Church 1999). Pattern languages enable the communication required to share concepts. Instead of an ambiguous conversation where one attempts to allude to an implementation ("You know, it would look a bit like that porch Dave has, but it would have some stairs, and we'd put it in the back garden instead of the front"), one can utilize the language provided ("What we need is a stair seat leading to the back door").

While Alexander defined design patterns to be problem-solution pairs, subsequent authors, in particular Tidwell (2010) and Björk and Holopainen (2004), have strayed from having such well-defined pairs to an area where the patterns are simply described, and their effects documented. Rather than saying "If you have this problem, use this pattern," the patterns are more designed for "While you're experimenting with this, you might find this pattern valuable." Björk and Holopainen (2004) gives three reasons why they expanded the term:

1. Defining patterns as problem-solution pairs could give the impression that they are only useful for removing unwanted problems, rather than supporting creative

49

design work.

2. Patterns overlapped, providing multiple solutions for problems.

3. Game design patterns affect many different areas of gameplay, making them imprecise.

As motivational design patterns are intended to be used creatively, this looser usage of design pattern is the definition intended when this dissertation makes mention of "design patterns."

3.2 Using motivational user stories to discover patterns

3.2.1 Motivational user stories

User stories are an approach in software engineering to ensure that a piece of software meets the requirements of its users. Ambler ("Introduction to User Stories") (adapted from Cohn 2004) gives them the form of:

As a <role>, I want <something> so that <benefit>.

User stories help focus design planning, ensuring teams meet actual user needs, rather than building unnecessary features.[2] For this reason, user stories are generally specific, with specifically defined features and outcomes.

While user stories are very useful tools at the design stage, the reasons are highly utilitarian, such as this example from Ambler ("Introduction to User Stories"):

As a student, I want to purchase a parking pass for my car so that I can drive to school.

---

[2]To external observers, it may sound odd that software teams would spend time implementing unnecessary features, but it's startlingly easy to do, even in small teams. This is partly a function of software developers not necessarily being the target audience of the software, and partly a function of "feature creep": the momentum that builds to continue adding features to software without stopping to think about why the feature is important.

However, a user story like this also has a number of hidden motivational desires that it meets. Perhaps the student wants to get to school on time because he wishes to feel honorable and not interrupt lectures by coming in late. Perhaps he values the status that owning a car can bring. Maybe he uses the car as a means of showing off to girls, increasing the chances he'll have a romantic encounter. Studying user stories in this manner can often yield connections to these deeper desires. What is required is a better way of surfacing these desires in the user stories themselves. Thus, the first step taken in analyzing patterns through the lens of Reiss desires was to modify user stories to create *motivational user stories*. These stories constrain the reason aspect of user stories to Reiss desires, bringing the level of abstraction down to a motivational level. These reasons describe how the goal satisfies a desire, not how certain functionality is provided.

For a user story to be motivational the benefit must be the direct satisfaction of a desire. If there is no direct satisfaction, then the benefit forms a general user story. Motivational user stories take a similar form to general user stories:

As a <role>, I want <something> so that <Reiss desire is met>.

As different people have different requirements for how much of a certain desire they need to be fulfilled, the "<role>" template was kept. One can imagine "child," "adult," "seller," and "buyer" as all reasonable demographics for these stories. In the absence of knowing the particular audience any piece of software that may use a pattern is targeting, this dissertation will use the broad "user" term.

3.2.2   Level of abstraction

The next part of creating motivational user stories is to identify what the "<something>" should be. What's an appropriate level of abstraction to use? Should it be at the user interface level? The code level? Somewhere in between?

51

Reiss Desire	As a user, I want <something>, so that
Power	I can feel powerful and meet my goals
Curiosity	I can gain understanding of the world around me
Independence	I can make choices that are meaningful to me and explore possibilities about myself
Status	I feel like I am an important person
Social contact	I can connect with others
Vengeance	I can compete against others
Honor	I can feel reliable
Idealism	I can help others and improve their situation
Romance	I can court sexual partners
Order	I can create an environment that feels stable and ordered
Acceptance	I feel others feel highly of me, giving me confidence
Tranquility	I am not scared
Saving	I have things I own and that are mine

Table 3.1: A table showing how to create motivational user stories from each Reiss desire. The family, eating and physical exercise desires are excluded from this list, as they are difficult to apply in a software setting. While it is possible to collapse some desires that appear together often into more general categories (such as power and status, power and vengeance), the full specificity of the Reiss desires framework is kept in order to maintain consistency with other work that uses Reiss desires, as well as provide the option of using the desires separately if required.

The user interface level is the most natural fit. This level is where users perceive whether a certain widget provides the affordances necessary to meet their motivational needs, so it's an appropriate level to define the "<something>" for motivational user stories.

Let's take a look at a model of the different layers that a user moves through to satisfy their desires. We'll first start from a top-down perspective, going from the software level into the user:[3]

1. User interface widget.

2. Affordances.

3. Desire.

More specifically, a user interface widget provides affordances, which allows some desire to be met. This model goes from higher to lower level, but what drives us comes from that lower level and bubbles up. Moving backwards up the stack allows us to write a step-by-step example of how a motivational user story is enacted upon, understanding the mental process that users go through. For example:

3. User desires feedback from their friends to feel social approval (desire).

2. User perceives that comments on *Facebook* have "Like," "Comment" and "Share" buttons below, and so understands that posting their own comment will also have such interactions allowed (affordances).

1. User enters a comment in the status update box, and clicks "Post" (user interface widget).

In terms of motivational user stories, the user starts with "As a user, I want <something> so that I can feel social approval", and then perceives what the "<some-

---

[3]Inspired by Antin (2012), with help from Peter Mawhorter at UC Santa Cruz.

thing>" can be, instantiating the story as "As a user, I want to post a status update that people can 'Like' so that I can feel social approval."

3.2.3 Motivational design pattern definition

With the appropriate level of abstraction decided, it is now possible to create a more formal definition of what a motivational design pattern is. Björk and Holopainen (2004) provides a good starting point for this. The game design patterns they present are very similar in scope to motivational design patterns, being concerned with creative experimentation that work towards creating some emotional experience for the player. It thus makes sense to base the definition of motivational design patterns from that of game design patterns. Björk and Holopainen (2004, p. 34) defines game design patterns as:

> Semiformal interdependent descriptions of commonly recurring parts of the design of a game that concern gameplay.

Adapting this definition, motivational design patterns are:

> Semiformal descriptions of commonly recurring parts of the design of an application that concern motivating user behavior.

As with game design patterns, motivational design patterns are imprecise tools, thus the "semiformal" aspect is kept in this definition. The main difference is the loss of "interdependent." Motivational design patterns are not as entwined as game design patterns. Game design patterns have very strong relationships. For example, the inclusion of one pattern in a game may enable or disable the inclusion of another. Motivational design patterns are more independent, and can be mixed more freely to create different applications.

3.2.4 Pattern discovery

Using the motivational user story templates generated in Table 3.1 as skeletons, patterns were generated by filling in widgets that could satisfy the stories. This approach allowed for focus on the motivation rather than the design, and helped lead to some unexpected patterns. For example, the UNDO pattern was included after thought about how applications can offer *Tranquility*. Without using Reiss desires as the initial starting, it is unlikely UNDO's motivational power would have been clear, and it may have not been included in the library.

Widgets were found using a "brute force" approach (Björk and Holopainen 2004, p. 52) through extensive interaction with motivational software—on the Web, on mobile platforms and on game consoles. They were also found by investigating the following pre-existing pattern libraries:

- *Patterns in Game Design* by Björk and Holopainen (2004), a library of gameplay patterns that can be combined together to make different game designs.

- *Designing Social Interfaces: Principles, Patterns, and Practices for Improving the User Experience* by Crumlish and Malone (2009), a pattern library that focuses on the patterns used by social media products to help users communicate with one another.

- "Game On: 16 Design Patterns for User Engagement" by Direkova (2011), a set of patterns designed to increase user engagement, largely through gameful means.

- *Designing Interfaces* by Tidwell (2010), a pattern library that identifies common elements from interfaces across all genres of desktop software.

- *Persuasive Patterns* by Toxboe (2011), a library of patterns that are focused on motivating users, much like this one.[4]

---

[4]The key difference between the library presented in this dissertation and Toxboe's is that this

- *Gamification by Design: Implementing Game Mechanics in Web and Mobile Apps* by Zichermann and Cunningham (2011), a set of patterns that appear often in applications that are part of the gamification movement.

Patterns were tested by using them both as tools to analyze existing software, and as a means of creating new software designs from problem statements. As these tasks were completed, various refinements of the pattern library were made. The final output of the analysis and generative tasks is presented in Chapters 12 to 13.

3.3   Prototype theory

While patterns can be thought of as fairly rigid things,[5] this was never a part of Alexander's original intent. He describes implementing solutions in a variety of different ways, and advocates modifying patterns to suit (Alexander 1977, p. xl). A similarly loose view is taken here. While it is attempted to describe the core of the pattern, that core can manifest itself in many different ways.

One way of better understanding this looseness is to comprehend patterns as part of a prototype theory approach. Prototype theory is a cognitive approach to how we categorize things in the world. How we define categories is far less rigid than we might think.[6] Think of a bird. You might think of a robin or a sparrow, or perhaps even an eagle. You probably didn't think of a flightless bird like the ostrich, kiwi or penguin. We know they are birds, but they seem less "birdy" than the others. Rather than making a mental Venn diagram of "bird" and "not bird," we really think of things

---

library uses Reiss desires to provide a consistent framework to justify the existence of patterns. The pattern library in this dissertation also generally goes into more detail.

[5]This probably comes from *Design Patterns: Elements of Reusable Object-Oriented Software*, as their patterns were so cohesive that it made little sense to deviate from the pattern as it was written.

[6]Lakoff (1987) describes this: "From the time of Aristotle to the later work of Wittgenstein, categories were thought be well understood and unproblematic. They were assumed to be abstract containers, with things either inside or outside the category. Things were assumed to be in the same category if and only if they had certain properties in common. And the properties they had in common were taken as defining the category."

with fuzzy boundaries, and prototype theory helps us understand these *cognitive models*. Here are a subset of the things that Lakoff (1987, p. 12–13) says goes into a cognitive model:

Reference-point reasoning The member of a category can stand as the whole category in certain reasoning.

Membership gradience Some categories have degrees of membership and no clear boundaries.

Centrality gradience Members which are clearly within the boundaries are more or less central, "central" meaning that they are "better" examples of the category than others.

Family resemblences Members of a category may be related to one another without all members having any properties in common that define the category.[7]

Conceptual Embodiment Properties of certain categories are a consequence of the nature of human biological capacities and of the experience of functioning in our environment.

In prototype theory terminology, the patterns presented in this dissertation are *categories*, where *central* members are used as examples to illustrate the category.

3.4   Epistemic issues

Prototype theory raises a particularly important epistemic issue about the categorization of patterns and *conceptual embodiment*. Conceptual embodiment argues that the objectivist view, that "There is a correct, God's eye view of the world—a single

---

[7]Games is a category that Wittgenstein (1953) argued has no common property; some we play for fun, some to win, some involve luck, some skill, some have win conditions, some don't.

correct way of understanding what is and is not true," does not exist (Lakoff 1987, p. 9). The follow-on implication is that the patterns presented are part of a personal cognitive model, and what was found useful for categorization at the time of writing this dissertation. There is no means of ascertaining whether they are "truly" correct, because there is no such truth.[8],[9] Instead, their value is from how well they provide the user with cognitive economy: "categories must be both specific enough to reflect all essential information and general enough not to overwhelm with irrelevancies" (Zagal et al. 2005). Taking the bird example again, the "bird" category contains things like "has feathers" and "can fly," and then inside the hierarchy of birds the specialization for an ostrich "can't fly" is added (Sternberg 2009, p. 314). The bird category provides us with sufficient cognitive economy to be valuable, even though there are exceptions that we need to take into account.

The quality of the motivational design patterns in this dissertation is measured by how much cognitive economy they can provide. To do that, the patterns will be used in Chapters 12 to 13 as tools both for analysis and critique of current designs, as well as tools for generating new designs. If you, the reader, believe the usage of the patterns to be sound and reasonable, then the argument of cognitive economy is made.

In addition to the subjective nature of the patterns themselves, the inferred Reiss desires that each pattern is tagged with are also entirely subjective. Patterns are tagged with the desires they appear to fulfill, but some readers may feel that a pattern is incorrect through its inclusion or exclusion of a desire. One means of validating the desire tagging would be to perform user studies to see which desires are agreed or

---

[8]Lakoff even argues against the idea of there being a one and only transcendental mathematics, and that "no single 'correct' meaning of *mathematics* can be fixed" (Lakoff 1987, p. 360). There is far too much detail to his argument to present here, but the result is that mathematics is itself like human conceptual systems, and that "mathematics grows out of the structures of everyday experience and is used to understand other experiences" (Lakoff 1987, p. 365).

[9]From a less theoretical perspective, Beizer (1990) attempted to categorize software engineering bugs, and came to a similar conclusion, writing "there is no universally correct way to categorize bugs," and that there are a potentially infinite number of ways to form a categorization.

disagreed with. Such studies were not undertaken due to time constraints, but would be useful future work.

There are some things that the patterns do not take into account. Firstly, patterns are always part of a wider software design. The reductionist approach taken— extracting these patterns and analyzing them individually—is useful for the toolbox being created here, but limits the interrogation that can be performed. For example, it is not possible to ascertain which pattern is "best" at motivating behavior. How the pattern is used in a particular design, what other patterns are used in conjunction, the quality of the software as a whole and the audience the software caters to, all affect the impact of any given pattern.

A particular limitation in regards to audience is that this dissertation only looks at Western software, reviewed from the perspective of Western society; it's highly likely there are significant differences between how much a particular pattern appeals to other cultures. While Reiss desires were chosen because they are universally applicable, how much any given demographic requires a certain desire to be fulfilled can change.

The lack of context about the quality of the software as a whole is also important. Just as using one of the patterns from Gamma et al. (1994) doesn't necessarily create maintainable software, using a motivational pattern doesn't necessarily create engaging software. Motivational patterns should be viewed as design strategies that *amplify* motivation to use software; they do not add engagement. If the software doesn't contain any engaging features, adding motivational patterns won't help.

3.5   Pattern description and organization

3.5.1   Pattern template

The motivational design patterns presented all follow the same template, reproduced below.

Pattern: The name of the pattern

Description: A short description of the pattern.

Reiss desires: A list of Reiss desires that this pattern appeals to.

Reduces Reiss desires: [optional] A list of Reiss desires that this pattern inhibits fulfillment of.

Also known as: [optional] Other names this pattern is given by other pattern libraries.

Related to: [optional] Patterns that have some relationship with this pattern, such as being commonly used together or having similar goals.

Examples: A list of applications which include this pattern.

Use: A long form discussion on how this pattern is commonly used, and which behavioral psychology, behavioral economics and Reiss desires apply to this pattern.

Watch for: [optional] A discussion of poor implementations of this pattern, and other warnings that need to be taken into account to avoid pitfalls.

3.5.2 Short-hand notations

In order to quickly identify applications, Reiss desires and references to other patterns, the following short-hand notations are used.

- Applications are written *Like This*. For example, *Facebook*, *Google Search*, *FarmVille*.

- When a company is being referred to, no formatting is used to disambiguate when referring to the company and when referring to the application. This prevents confusion when discussing companies that are named after their main application. For example, Facebook, LinkedIn, Reddit.

- Reiss desires are written *Like This*. For example, *Power*, *Tranquility*, *Social Contact*.

- Patterns are referenced LIKE THIS. For example, SCORE, CONTACT LIST, BROADCAST.

- Dark patterns are referenced DARK PATTERN: LIKE THIS. For example, DARK PATTERN: IMPERSONATION, DARK PATTERN: GRIND, DARK PATTERN: CURRENCY CONFUSION.

In addition to the motivational design patterns, patterns from other pattern libraries are referenced, often in the "Related to:" sections of the new patterns. Patterns from other libraries are referenced as follows:

- *Patterns in Game Design* by Björk and Holopainen, referenced as *Bjork & Holopainen*: PATTERN NAME.

- *Gamification by Design: Implementing Game Mechanics in Web and Mobile Apps* by Zichermann and Cunningham, referenced as *Zichermann & Cunningham*: PATTERN NAME.

- *Designing Interfaces* by Tidwell, referenced as *Tidwell*: PATTERN NAME.

- *Designing Social Interfaces: Principles, Patterns, and Practices for Improving the User Experience* by Crumlish and Malone, referenced as *Crumlish & Malone*: PATTERN NAME.

- "Game On: 16 Design Patterns for User Engagement" by Direkova, referenced as *Direkova*: PATTERN NAME.

- *Persuasive Patterns* by Toxboe, referenced as *Toxboe*: PATTERN NAME.

3.5.3 Organization of patterns

Patterns are organized under four general headings:

61
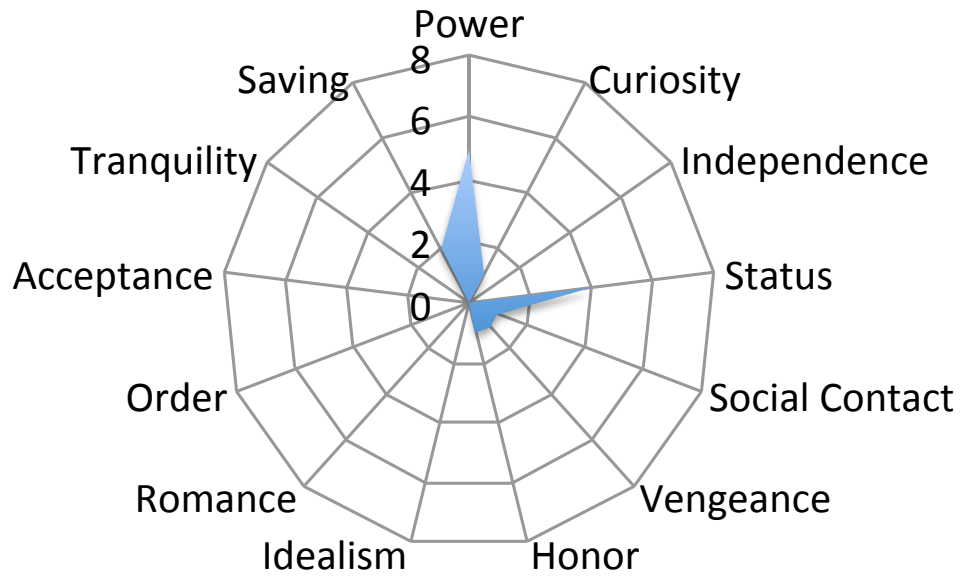
Figure 3.1: A spider chart showing the Reiss desires fulfilled by patterns inside the gameful category. Gameful patterns steer towards *Power* needs, where users are given the feedback to feel self-efficacy based on their growth of mastery over the game. They also confer *Status* to those who are doing well.

Gameful Patterns that exhibit a *gameful* nature, appealing to our desire to play.

Social Patterns that help us connect with others.

Interface Patterns related to how we interact with the interface.

Information Patterns that help us manage information that we require.

Patterns are ordered so that patterns which reference others appear after those which do not. For example, SCORE appears before LEADERBOARDS, as LEADERBOARDS heavily depends on knowledge gained from SCORE.

Figs. 3.1 to 3.5 comment on the distribution of Reiss' basic desires in the various categories.
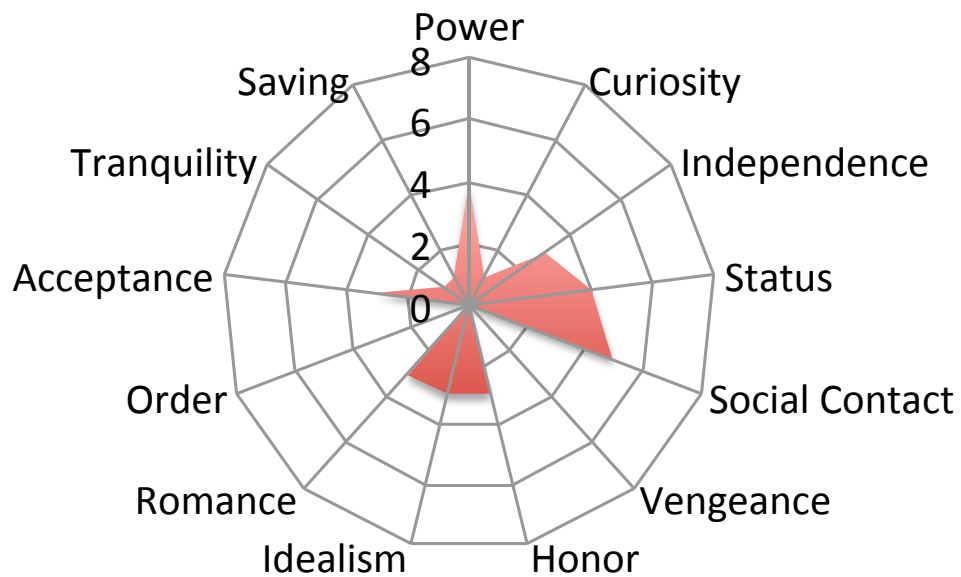
Figure 3.2: A spider chart showing the Reiss desires fulfilled by patterns inside the social category. Unsurprisingly, *Social Contact* is the most represented of the Reiss desires. Social patterns are also the only means with which to fulfill the *Romance* desire as users look for love, and the *Idealism* desire, where they work to create a better community.
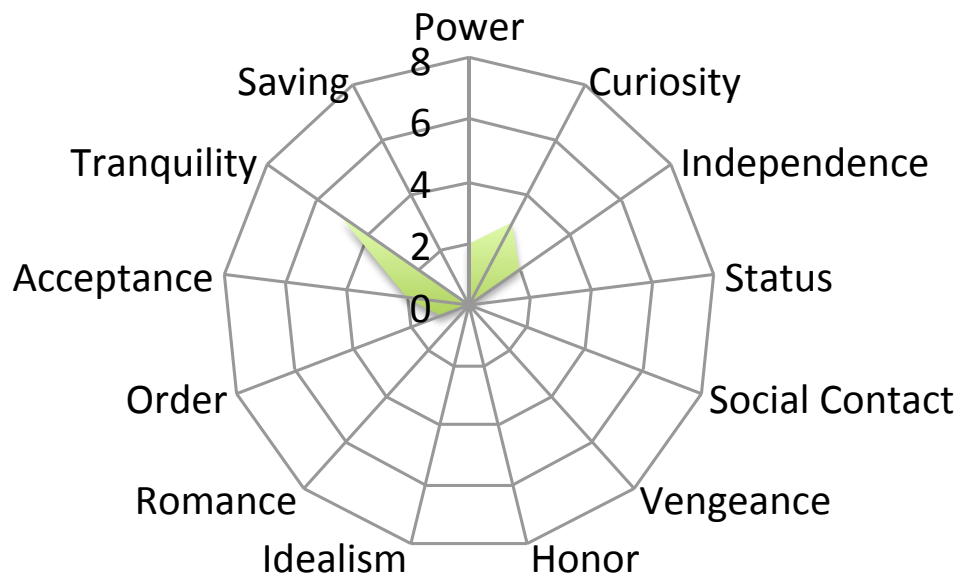
Figure 3.3: A spider chart showing the Reiss desires fulfilled by patterns inside the interface category. Patterns in the interface category largely cater to making users feel at ease with the application, increasing their *Tranquility*. They also allow for exploration, satisfying *Curiosity* and *Independence* needs.
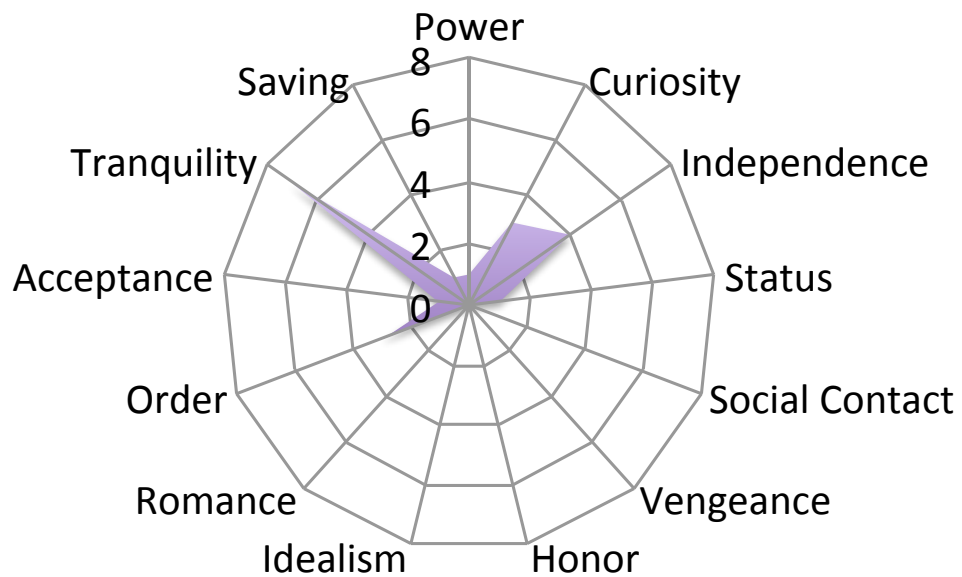
Figure 3.4: A spider chart showing the Reiss desires fulfilled by patterns inside the information category. Much like the interface category, information patterns are geared towards making users feel like their information is safe and recoverable, increasing their *Tranquility* and *Independence*.
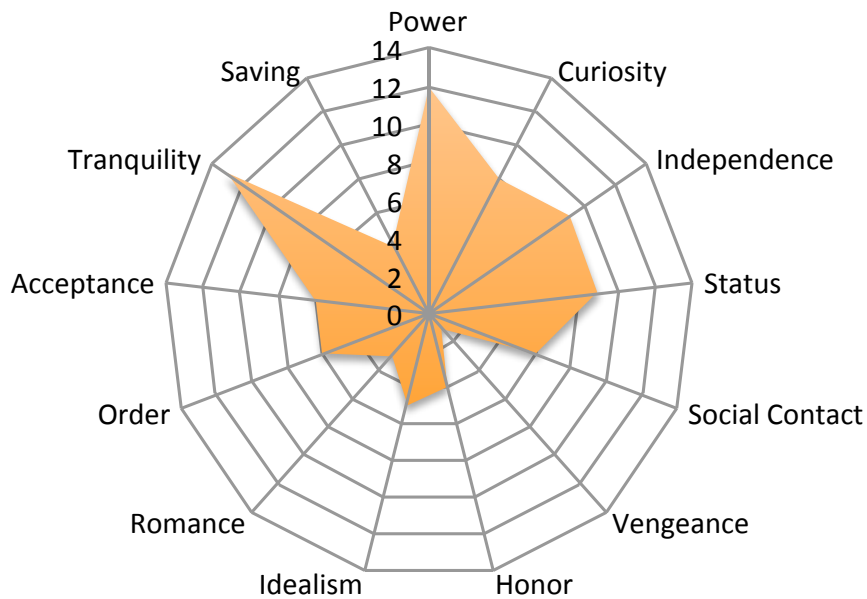
Figure 3.5: A spider chart showing the Reiss desires fulfilled by the entire pattern library. We can see that motivational design patterns skew towards satisfying *Tranquility* and *Power*. These two desires reflect how comfortable the user is with the application, and are likely required before the fulfillment of other desires can be adequately met. However, these are also the desires that computers have traditionally struggled to fulfill. Difficult desktop interfaces made many users feel uneasy and ineffective, and it is no surprise that motivational design patterns focus on improving this situation. It is worth noting that the desires in the bottom half of the chart, such as *Romance*, *Idealism*, *Honor* and *Vengeance*, are largely unrepresented by the patterns in the library. These desires usually require at least one other participant (real or virtual) in order to fulfill them, and are thus harder to incorporate in most software.

Additionally, there are three dark pattern categories that *reduce* users' ability to fulfill their Reiss desires:

Temporal dark patterns Patterns that cause users to incorrectly estimate how much time they will spend with an application.

Monetary dark patterns Patterns designed to encourage users to part with money in a way they did not expect, either by being confused into spending more money than expected, or feeling regret at the amount of money spent.

Social capital dark patterns Patterns that will result in users harming their social relationships.

Chapter 4

Gameful patterns

4.1 Overview

Gameful patterns are those which have the "qualities of gaming" (Deterding et al. 2011; McGonigal 2011). Creating interactions to support gamefulness is the process of "gameful design," and will typically use game design elements. The patterns in this chapter are all heavily inspired by games, such as SCORE, LEADERBOARDS and BADGES. Because many games also rely on social interactions, many of these patterns could have appeared under the Social patterns categorization, but they appear here if their gamefulness is more central than their social aspects.

The term "gameful design" is broadly aligned with that of gamification, but with a definition of "the use of game design elements in non-game contexts" (Deterding et al. 2011). Deterding et al. proposes using this term over "gamification" as it provides "less baggage, and [is] therefore a preferable term for academic discourse." From this point on, patterns that exhibit game design elements will be described as gameful, whereas gamification will be limited to discussion of work in Zichermann and Cunningham (2011).

When using gameful patterns, it is by no means guaranteed that users will

engage with the gaming aspect. Hamari (2012) argues against framing the discussion around game or non-game contexts, noting that simply providing a gameful context does not indicate that users will play. For example, people may not gain any feeling of gamefulness from chess. Conversely, it is possible to extract a gaming experience from a non-gaming context, such as "playing" the stock market.[1] If this is so, then we can't simply create gameful experiences, as we have no idea whether the users will play, and if they do, whether they will even play in the intended way. There is a strong possibility for gameful patterns to backfire, encouraging unexpected and undesired behavior.

Use gameful patterns when the process of engaging in the game brings a deeper understanding of either the application, or the underlying motivation the gameful pattern seeks to extract. Gameful patterns should encourage users to learn through exploring systems.

4.2   Pattern: Score

Description: A quantified value awarded to a user as a reward to certain actions, representing level of success

Reiss desires: *Power*

Also known as: *Zichermann & Cunningham*: EXPERIENCE POINTS, *Crumlish & Malone*: NAMED/NUMBERED LEVELS, *Crumlish & Malone*: POINTS, *Bjork & Holopainen*: SCORE

Related to: LEADERBOARD

Examples: *Foursquare*, *Pac-Man*, *Space Invaders*

Use: When we think about SCORES, we often think about doing well, or times when we have achieved a high score in a game. *Bjork & Holopainen*: SCORE describes score as "the numerical representation of the player's success in the game, often not only

---

[1]It seems that there is functionally zero difference between the game of the stock market and the game of sports betting.

representing the success but also defining it."

A more general definition is that SCORE is a pattern used when designers wish to give *feedback* to the user that they are participating in desired behavior, and to quantify the extent of satisfaction the designer has with that behavior. When quantification is not required, PRAISE can be used instead. SCORE is used when success is easily quantified, hence was a prevalent pattern within arcade games. SCORE in arcade games provided an easy metric for players to judge their mastery of the game, and remains a means of expressing a user's *Power* over something. As games have grown increasingly richer, SCORE has become less prominent. It cannot easily quantify things such as whether the player in *Bioshock* is a benevolent or malevolent force inside Rapture[2], nor the aesthetic qualities of a room design in *Animal Crossing*.

In this definition, SCORE only applies to a use where an arbitrary value is offered to the user, and that the arbitrary value is handed out by a computer system. For example, while gaining 100 points for completing a sign-up process is a use of the SCORE pattern, receiving five 'Likes' for a comment on *Facebook* is an example of the SOCIAL FEEDBACK pattern. It is tempting to believe that SCORE applies to any discretized and countable quantity, but how the quantity is perceived by the user, and how it matches with Reiss desires, indicates whether it is the quantity itself that should be valued (SCORE) or that the quantity indicates some deeper acceptance or competition (e.g. SOCIAL FEEDBACK, CONTACT LIST, REPUTATION).

Sometimes metrics are offered to users that they perceive implicitly as a SCORE, when the application does not present it as so. Take the board game the *Game of Life*. In the *Game of Life*, players move through various stages of life, from birth to death. The winning player is the one with the most money at the end of the game: money is

---

[2] *Bioshock* does actually try to perform this calculation behind the scenes, by giving you a good ending only if you do not harvest (kill) any Little Sisters. The game thus ignores players who begin to save Little Sisters after they are aggressively humanized at the end of the game's second act, or those who harvest a Little Sister out of necessity to support the greater good of liberating Rapture.

the SCORE. In *Monopoly*, the goal of the game is to be the last man standing, with all the other players bankrupt. The goal is not to amass as much money as possible. Money is an enabler of winning, but does not define success, thus money is an *implicit* SCORE. Taken to a further extreme, *Sim City* does not offer any appreciable goal at all. Players can choose to make large cities or small villages. Some players choose to make their city population as large as possible. Some players may choose to horde cash. In both instances, players treat the metric as an implicit SCORE, even when the game does not codify it as such. One benefit implicit SCORES have over explicit SCORES is that the user has the *Independence* to choose what she values. This degree of autonomy means that implicit SCORES can feel more rewarding than the explicit.

SCORE does not have to pertain only to continuous representations, but also discrete representations such as levels that range from 1–5 (*Crumlish & Malone*: NUMBERED LEVELS) or even words that translate to underlying numeric values, such as 'Beginner,', 'Journeyman,' 'Expert' and 'Master' (*Crumlish & Malone*: NAMED LEVELS). In fact, there is research that shows humans mentally compress the differences between numbers as they get bigger, commonly known as the *numerical distance effect* (discovered by Moyer and Landauer (1967), with a good summarization of work in Longo and Lourenco (2007)). The difference between 6 and 7 seems larger than between 76 and 77. This indicates that the communicative ability of SCORE tails off as the number increases, as we are less able to perceive the change.[3] Thus, discrete implementations, such as levels, may well have more impact than systems which use large numbers in order to try to "wow" their users.

Watch for: It is tempting to use SCORE as a shortcut to provide feedback quickly, but this only provides value when it meets the user's need to be powerful and express her mastery over a particular subject. Applying SCORE as a primary feedback mechanism

---

[3]Ariely (2010) also describes this effect, which he describes in the using *relativity*; that we are sensitive to advantages relative to context. For example, the $15 difference between $10 and $25 seems much more important than the $15 of $425 and $440.
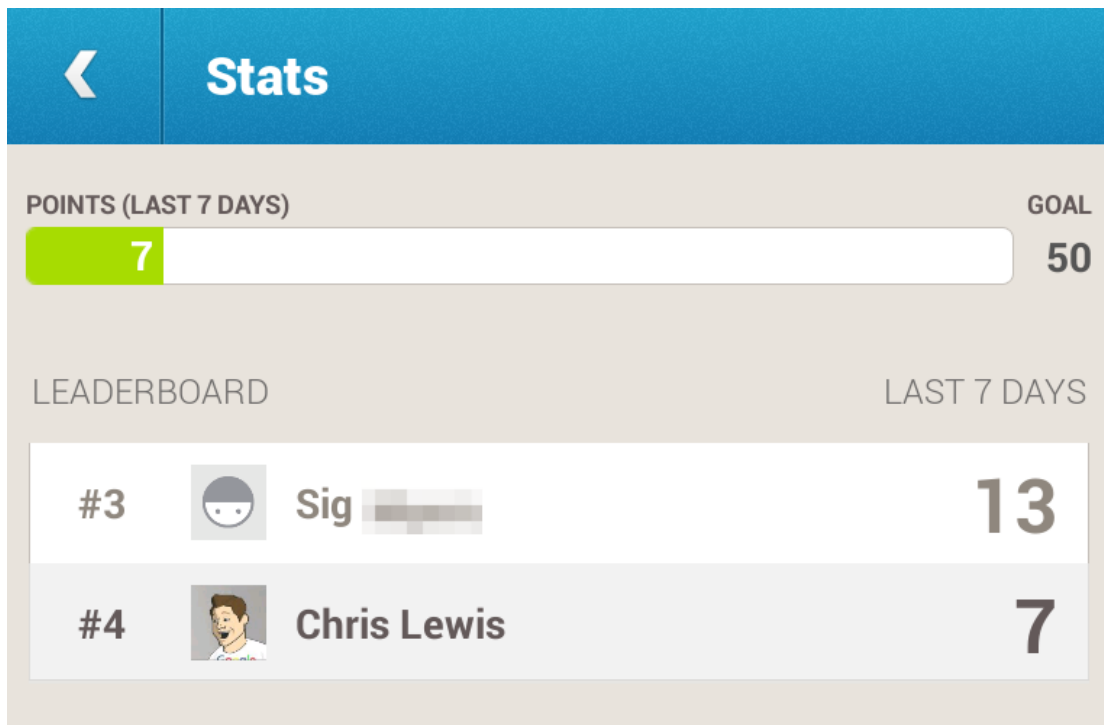
Figure 4.1: A screenshot of *Foursquare* points as seen in the *Android* app. A progress bar indicates a goal of achieving 50 points. If a user has achieved that goal, then the user is then presented with the goal to match her previous peak seven day score in the last seven days (this is a use of a hedonic treadmill, discussed later in Section 4.3). These points are only surfaced in the mobile apps and mobile website, and do not appear on the desktop website. Last names obscured for privacy.

often misses the deeper connections and greater range of motives that could be met, and can be disastrous applied when users are not looking for a satisfaction of *Power* alone. When SCORE is excised from an exciting game design and applied elsewhere, it's easy to spot that striving to achieve a score is a fairly vacuous and meaningless thing in and of itself. This likely explains why there doesn't seem to be any successful use of the pattern outside of games. Even in games, the findings of Malone (1981) indicate that score is not a primary feature of games, and other mechanisms for feedback are just as, if not more, useful.

A classic example of a poor use SCORE is found in the *Foursquare* app.
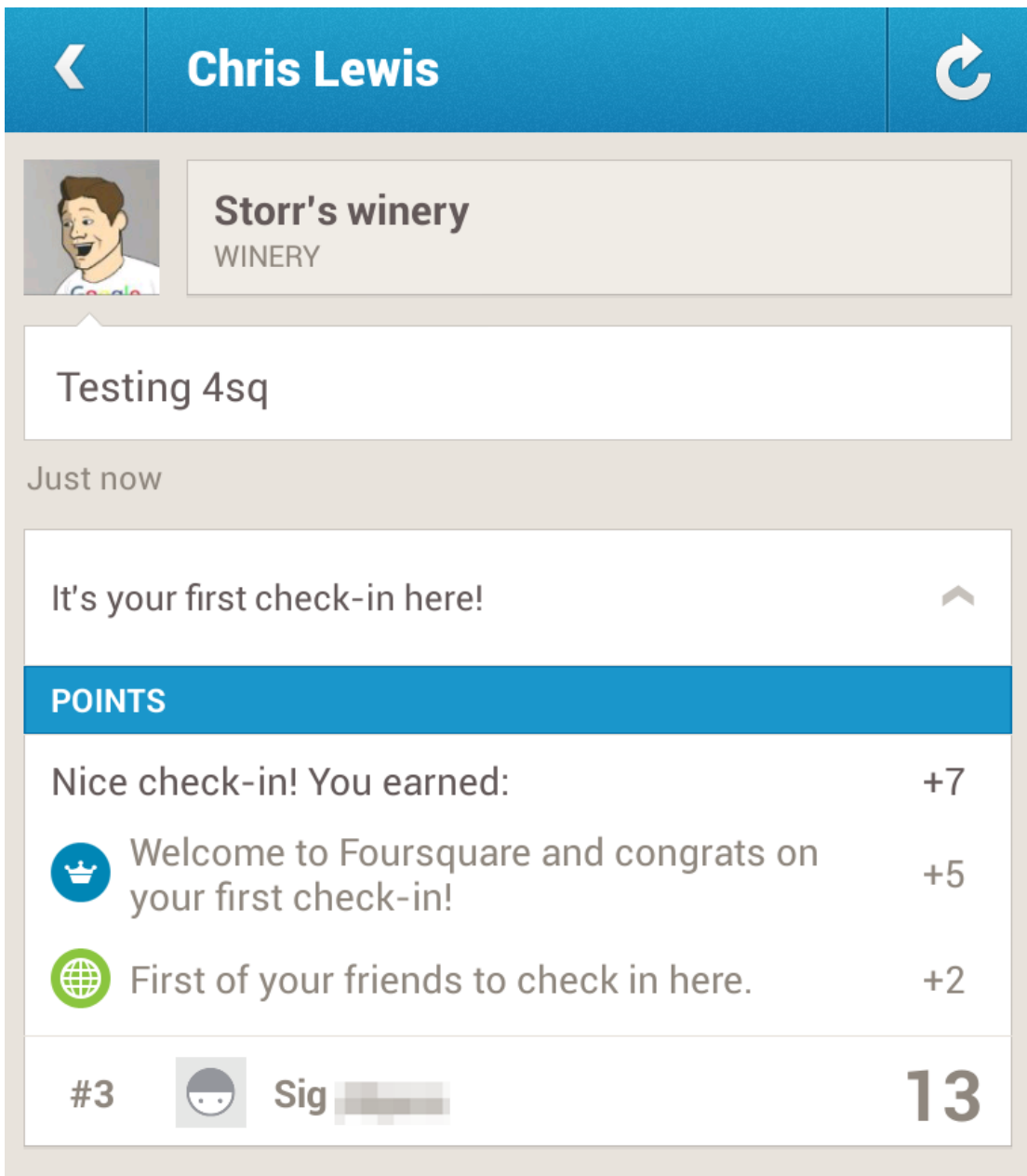
Figure 4.2: A screenshot of *Foursquare* check-in as seen in the *Android* app. A check-in is worth five points, and the user scored an extra seven for his first check-in and for checking-in somewhere that his friends had not yet. Last names obscured for privacy.

Foursquare asks users to "check-in" to places they visit with their smartphone, sharing their location with others, and creating a list of a user's favorite haunts. *Foursquare*'s approach to gamification has been modified over the years, and at one stage even removing points, only to restore them soon afterwards (Knowles 2012). Fig. 4.1 shows how points are surfaced in the app at the time of writing. Points are offered for things such as checking into a venue, adding venues, or various bonuses such as being the first of your friends to check-in somewhere. Fig. 4.2 shows the user checking into his first establishment. Points can then be seen in a user's profile, or as part of a LEADERBOARD with friends.

One glaring question is why users should find *Foursquare*'s score valuable, and Foursquare itself seems unsure: it heavily de-emphasized the gamified aspects of the application in the summer of 2012, more closely aligning with *Yelp* (Sweeny 2012; Smith 2012). *Foursquare*'s tag line in one promotional video ran "Keep up with friends. Discover what's nearby. Save money and unlock rewards" (Foursquare 2012). These goals tap into Reiss motives of *Social Contact*, *Discovery*, and *Saving*. SCORE, on the other hand, is a pattern that pertains to *Power*. It doesn't provide useful feedback to support any of the motivations that lead to *Foursquare*'s offered goals, although it can be combined as a system to dictate a position on a LEADERBOARD, which mixes-in a motive of *Social Contact*. *Foursquare* changed to better match these motivations through systems such as an ACTIVITY STREAM that shows nearby locations that may be of interest and using check-ins as a means of IDENTITY SHAPING. The feeling of *Foursquare* mastery that SCORE offered pales in comparison to the feelings of mastery of surroundings by discovering a wonderful new coffee shop. It is notable that *Yelp*, a review site for businesses that offers much the same functionality as *Foursquare*, utilizes a number of feedback systems, but does not use a SCORE in the same way as *Foursquare* does.

Another use of SCORE misaligning with user motives can be seen in the *Star*
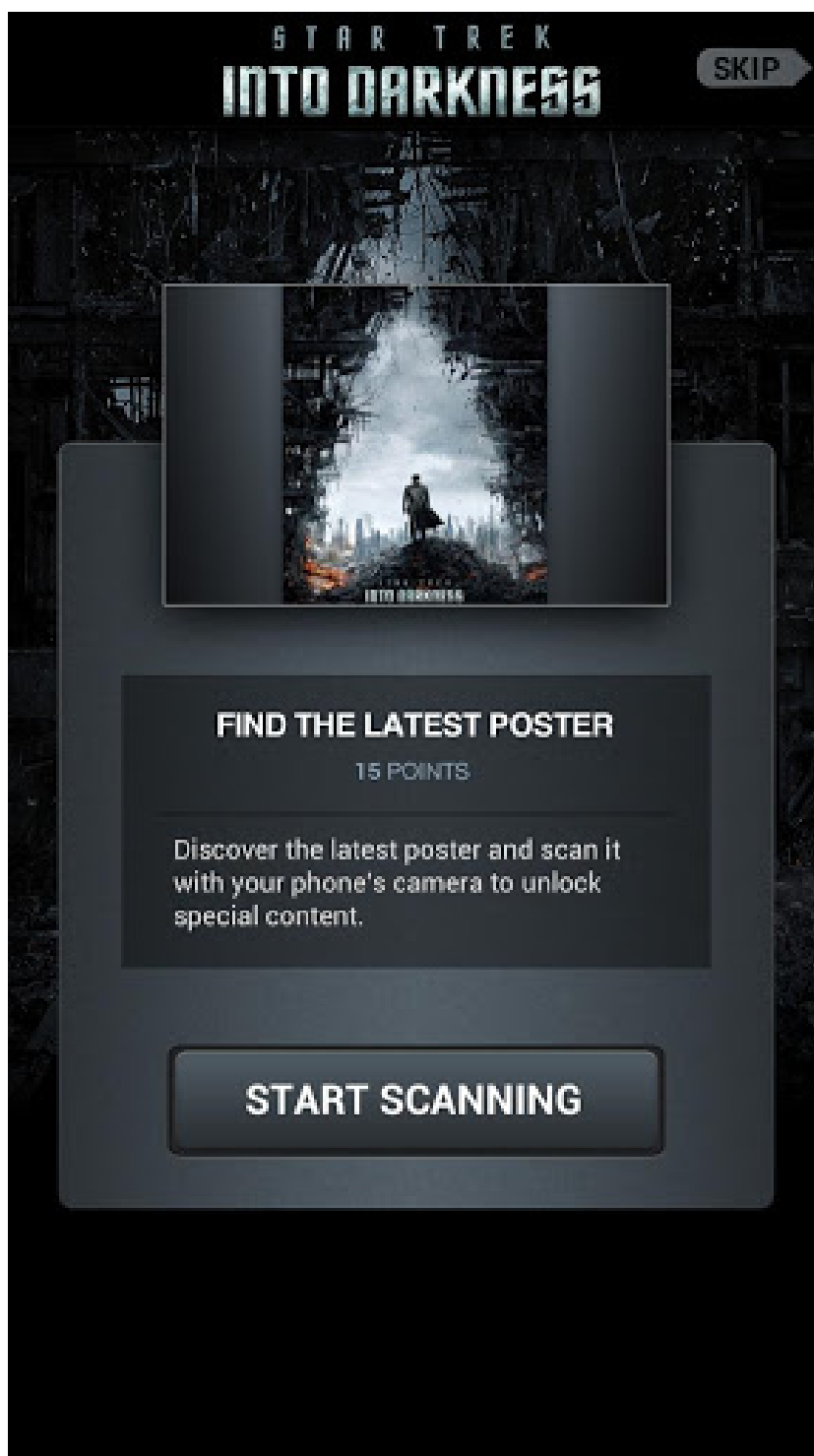
Figure 4.3: A screenshot of a *Star Trek: Into Darkness* mission, which offers 15 points for scanning a movie poster with your phone.

Trek: Into Darkness app for mobile. This app allows users to scan television or billboard advertising of the new *Star Trek: Into Darkness* movie, which confers points, as seen in Fig. 4.3. Points can also be gained by entering sweepstakes or checking-in from San Francisco where the Starfleet Academy will be built in the *Star Trek* fiction.[4] One may download the *Star Trek: Into Darkness* app with the expectation of finding out more about the movie, or perhaps getting a nice new phone wallpaper. It instead presents "Missions" that must be performed in order to unlock content. The designers do not say what content will be unlocked, perhaps worried that users will not play if the rewards are not what they want. It's likely the unlockable content will be more advertising for the movie, and so the app asks users to perform non-trivial interactions in order to be given the opportunity to have more advertisements shown to them! As if by way of apology, the app's FAQ contains the following:

> **What are points good for?**
> Points allow you to advance in rank which will give you more opportunity
> to accomplish missions and unlock content. Plus it's great to out rank your
> friends.

This starkly illustrates the fundamental mismatch between the user's motivation for interacting with the app, and the motivations that the app tries to support. The app's promise is that it will fulfill the user's *Curiosity*, but instead it offers *Power* and *Vengeance*. This shows a profound misunderstanding of the audience.

4.3 Pattern: Leaderboard

Description: Placing a user in a list of others, ranked by a chosen metric

Reiss desires: *Social Contact*, *Vengeance*, *Status*

Also known as: *Bjork & Holopainen*: HIGH SCORE LISTS, *Crumlish & Malone*: LEADERBOARDS, *Zichermann & Cunningham*: LEADERBOARDS, Mayorship, Neighbor

---

[4]For the 342 million people in the US & Canada that do not live in the Bay Area, a vague promise of more location-based missions every week was made.

bar

Related to: CONTACT LIST, INCREASED RESPONSIBILITY, SCORE

Examples: *Doodle Jump*, *FarmVille*, *Foursquare*

Use: *Bjork & Holopainen*: HIGH SCORE LISTS defines LEADERBOARDS as a opportunity for "players to rank themselves against other players who have previously played the game." The term LEADERBOARDS is used rather than rather than HIGH SCORE LISTS in order to remove the implication of SCORE as being the only possible ranking method. You could rank players entirely arbitrarily, perhaps in order to ensure that no player is ever left disheartened at the bottom of the leaderboard.

Zichermann & Cunningham: LEADERBOARDS describes several ways a LEADERBOARD can be displayed:

The no-disincentive leaderboard A leaderboard where the player is always displayed in the middle, so she can see those just above her and those just below her.

The infinite leaderboard A leaderboard that can be sliced into various different views, such as one that displays the global list, one that only displays friends, one that only shows users in the same geographic area...

It is unclear exactly what differentiates the infinite leaderboard from the no-disincentive leaderboard, as the infinite leaderboard should allow the player to view her position in the leaderboard regardless. The no-disincentive leaderboard is an instantiation of the infinite leaderboard to a particular default. What makes a sane default is different depending on the context, but it is increasingly rare to see a leaderboard that does not default to using friends lists. Both Web and mobile apps often connect with *Facebook* to bring in a user's social graph, and utilize this knowledge to automatically find friends that are also using the app.[5] Playstation Network, Xbox Live and *iOS*

---

[5] *Direkova*: GATED TRIAL refers to this trend, but she expressly phrases this pattern as a means of

Gamecenter also provide such functionality at an operating system level.

Madrigan (2012) theorizes that social comparison theory (Festinger 1954) helps us understand how LEADERBOARDS work, especially in the context of using friends:

> We prefer, for example, comparisons to people who are like us in certain respects, and tend not to compare ourselves to people whose ability is way above or far below our own... Many researchers think that the reason we seek out comparisons with people we're familiar with is that we use them as proxies for guessing if we're capable of some feat or achievement. If another person achieves something tricky, our confidence in our ability to do it as well is linked to our opinion of how we're similar in relevant attributes. This is obviously easier to do with people we know.

This theory also neatly aligns with the view of self-determination theory (SDT), which also notes that challenges which we have no hope of attaining can't provide any feeling of mastery.[6] Tapping into friendship circles helps us better calibrate our expectation of success. Using friends also increases meaningful *Social Contact*, and we increase our *Status* among our peer group when we perform well. *Vengeance* is more attainable when we are matched with people of similar skill. All these factors largely explain why global leaderboards are seldom the default view, as friend leaderboards are much more powerful at amplifying and fulfilling our needs for *Social Contact*, *Status* and *Vengeance*.

In social games, the '*Ville* series in particular, "neighbor bars" are presented as CONTACT LISTS, but also serve as LEADERBOARDS. These bars allow the user to visit their friends' game spaces, and they reorder themselves based on players' experience points. For Zynga, ranking by experience points makes sense. Their business model is predicated on as many people playing the game as possible, as they operate on slim percentages of players that will actually pay. In order to make the slim percentage a

---

blocking access unless the social graph is brought in. Few apps actually prevent use if social networking credentials are not offered, the major exception being *Spotify* which requires a *Facebook* account.

[6]Rigby and Ryan (2011) uses the example of dunking a basketball: while dunking a basketball at three feet poses no mastery challenge, neither does dunking a basketball at regulation height for the majority of the populace. It's simply not achievable, and thus reduces our feelings of competency.

Figure 4.4: A screenshot of a neighbor bar from *Empires & Allies*. Avatars obscured for privacy.

large enough *absolute* number, they need as many people active in the game as possible, which will encourage others to do the same. When we think of LEADERBOARDS in games, we often think of them showing the most skillful players, which may have some correlation to how much time they have invested in improving their skills. Here, the time a player has played is the *primary factor* dictating a player's position in the neighbor bar, and thus is an instance of DARK PATTERN: GRINDING.

Using a behavioral economics lens, LEADERBOARDS are instantiations of *hedonic treadmills* (Brickman and Campbell 1971). Hedonic treadmills promise us goals that will make us happy, only to find that achieving that goal presents a new one, so our happiness returns to a baseline until that goal is achieved. We continue marching up a hill expecting to finally feel happy, but the new situation we find ourselves in leads us to feeling no better. One example might be the graduate student that works hard at university to land a tenure-track job. Finally, she has the academic job she worked so hard for! But now, everyone around her has tenure. She's moved up in the world, but her world view has now changed so that she's below everyone else again. Thus she continues to work hard to try and get tenure, cycling on the hedonic treadmill anew. LEADERBOARDS offer a similar sense of fleeting satisfaction. Finally beating a friend, or even reaching the top of the leaderboard might feel like a huge achievement, but being

at the head of the leaderboard does not last for long. This is especially true when apps such as *Foursquare* or *Robot Unicorn Attack: Evolution* clear their leaderboards every seven days, forcing users to return if they wish to place on the Leaderboard once more.

Watch for: Given what we know about attainable challenges being a core part of experiencing mastery, it is surprising to see the popularity of the *Foursquare* mayorship. Becoming a mayor in *Foursquare* requires the user to check-in to a venue more times than any other, and this person is crowned mayor. Being the mayor means the user is at the top of the Leaderboard. The problem with the mayorship system is that only one person can be the mayor at any one time. This means that the competition doesn't scale. Those who are not within the top competitive band have no hope of becoming the mayor, and so cannot meet any mastery need.[7] A particularly egregious example of this was used by *WayIn*, who implemented a mayorship which they called "Owner." How to become the owner is not listed on the site, which means that even those who achieve it cannot experience mastery, as described in Rigby and Ryan (2011):

> ...we need to have clarity about the goal at hand ("jump high enough to stuff this ball through that hoop"). If we are uncertain about what we are undertaking, it is harder to act meaningfully or to feel a sense of mastery, even if we succeed.

The second issue is that of population. Fig. 4.5 shows those who subscribe, and are thus eligible, to become the owner of the "Technology" community, which users are automatically subscribed to.[8] There can be only one owner, but we find 72,840 users in the community. With such a daunting number of people, and only one person at the top, it's highly likely the majority of users don't see being the owner as an achievable task, and thus don't engage with the Leaderboard at all. Population, and limiting that population, is a key part of any Leaderboard use.

---

[7]Paraphrasing Zichermann and Cunningham, one might call this the "all-disincentive leaderboard."
[8]Users are also subscribed to Movies, Music, Politics and "SportsBall."
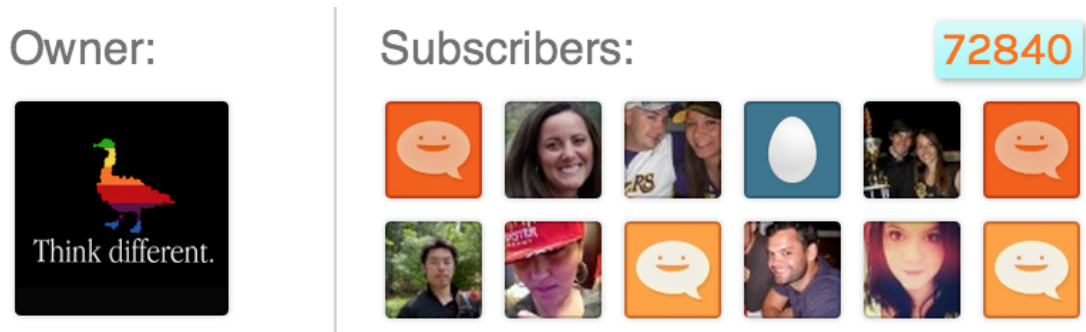
Figure 4.5: A screenshot from *WayIn* showing the owner of, and the subscribers to, the "Technology" community. Taken on February 5, 2013.

4.4 Pattern: Increased Responsibility

Description: As a user becomes more trusted by the designer or the community, that user can take more influential actions

Reiss desires: *Honor*, *Idealism*, *Power*, *Status*

Also known as: *Toxboe*: Unlock Features

Related to: Meta-Area

Examples: *Hacker News*, *Stack Overflow*, *Yelp*

Use: Increased Responsibility is a reward given to users when they achieve a certain amount of notability. There are often different levels of responsibility that can be conferred, and these are usually handed out piece-meal (such as with *Hacker News*). Users who have been given Increased Responsibility are conferred *Power* and *Status*, and can use their newfound power to improve the community, increasing their *Idealism*. Those who are new to the community can look towards these members for pointers to their expected behavior, and so this helps feed their need for *Honor* (although this can often communicated more directly via Badges). This is done via the concept of "social proof." Social proof is the theory that when we aren't sure of our expected behavior, we'll turn to others to decide what to do, and the popular set

Karma	Reward
10	Upvote comments
200	Flag comments
500	Downvote comments
?	Make polls
?	Change top bar color

Table 4.1: A table showing the "karma" required to unlock increased responsibility on *Hacker News*, with figures taken from Mattheij (2011). The figures change frequently, as the founder, Paul Graham, ups the limits in order to cancel out "karma inflation." "Karma inflation" occurs as users stay longer, even untrusted users could gain enough karma by just trying to avoid being obnoxious and downvoted. Increasing the limits frequently ensures only trusted users gain access to the features.

of decisions are correct (Cialdini 2008).[9] Reputable users provide social proof to new users of the norms expected of them in the community.

The pattern is more easily communicated by examples, which can be seen in Tables 4.1 to 4.2. The Yelp Elite Squad, on the other hand, cannot be so easily summarized into a table. Members of the Yelp Elite Squad "reveal hot spots for fellow locals, act as city ambassadors, and are the true heart of the Yelp community, both on and offline"(*Yelp Elite Squad*).[10] Elite Squad members must apply to be chosen by "The Elite Council", who grade yelpers based on their frequency and quality of reviewing and commenting, and submitting Social Feedback on others' reviews.

Increased Responsibility is a tangible extrinsic reward. In the case of *Stack Overflow*, the reward is tangible, expected and contingent reward, which could harm intrinsic motivation for a task. One would hope that *Stack Overflow*'s users find the task of answering tricky questions interesting and rewarding, so the inclusion of Increased Responsibility appears dangerous. We might expect a high number of

---

[9]Cialdini gives the example of laugh tracks in television comedies. Audiences laugh more when a laugh track is present, even though many people (myself included) find laugh tracks abhorrent.

[10]A more questionable "benefit" is that Elite Squad members also receive "first dibs on everything from Yelp sunglasses and lip balm to sweatbands and temporary tattoos."

REPUTATION	PRIVILEGES
1	Create posts
5	Participate in meta [a META-AREA for Stack Overflow]
10	Remove new user restrictions [removal of spam guards]
10	Create wiki posts [posts that others can edit]
15	Vote up
15	Flag posts
20	Talk in chat
50	Comment everywhere
75	Set bounties [bounties are used to offer extra reputation for answering a question]
100	Edit wiki posts
100	Create chat rooms
125	Vote down
200	Reduce ads
250	View close votes
500	Retag questions [change the tags on a question to better organize questions]
1000	Established user [can view upvotes/downvotes on posts, have an extended user representation, helping with IDENTITY SHAPING]
1000	Create gallery chat rooms [rooms where only certain people can talk]
1500	Create tags
2000	Edit questions and answers
2500	Create tag synonyms [automatic retagging of a given tag]
3000	Cast close and reopen votes [shut down a question]
5000	Approve tag wiki edits [approve submitted changes to the description of a tag]
10,000	Access to moderator tools [close/delete posts, view deleted posts, access to special lists of questions that indicate potential problems, such as posts with extreme votes]
15,000	Protect questions [prevent anonymous and new users answering a question]
20,000	Trusted user [more powers to vote for deletions, can edit wikis without approval]

Table 4.2: A table showing the reputation required for given privileges on *Stack Overflow*, taken from http://stackoverflow.com/privileges.
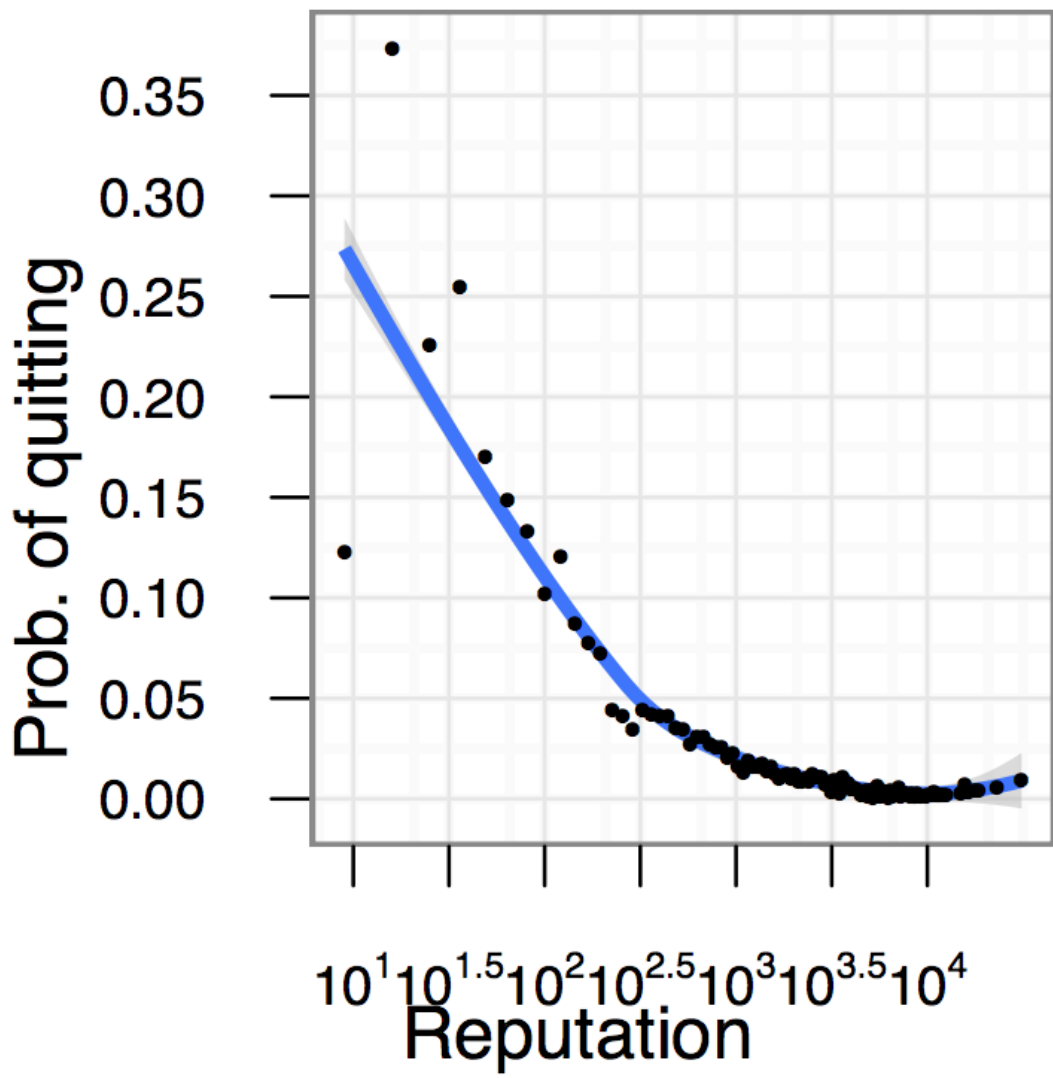
Figure 4.6: A chart showing the probability of users quitting, presented in Lotufo, Passos, and Czarnecki (2012).

users quitting after they attain their intended reward, likely the moderator tools at 10,000 reputation.

Fortunately for researchers, all of *Stack Overflow*'s data is freely available in a database dump to XML. In Lotufo, Passos, and Czarnecki (2012), quit rates of users were analyzed, and presented in Fig. 4.6. Here, quitting means that the user has not interacted with the site within 60 days of their data collection. The chart shows the probability of quitting decreases to below 5% after attaining 300 reputation points, with a small rise in quitting after 10,000 points. The authors suggest that this means there is "a decrease in user motivation after achieving such high reputation and privileges," and with our understanding of extrinsic motivators, this seems like a reasonable conclusion. However, the increase in quitting is barely visible in the data, moving about 1% higher than before the moderator tools are granted.

In a separate study, Mamykina et al. (2011), found four different types of *Stack Overflow* users:

Community Activists Registered users who are highly active on the site for multiple months. (1% of the user base, provides 27.8% of the answers)

Shooting Stars Registered users who have a single, short period of high activity followed by low activity. (4.2% of the user base, provides 21.9% of the answers)

Low-profile Users Registered users who have intermittent activity, but who never become highly active. (94.4% of users, supply 34.4% of the answers)

Lurkers and Visitors Users who have not been asking or answering questions; visitors without user accounts.

The authors believe that the gameful elements of *Stack Overflow* draws in both community activists and shooting stars, but that the latter "moves on after a short infatuation period." They go on to suggest that the moderation tools threshold "[leads] to a subsequent reduction in participation, creating the shooting star pattern."

Unfortunately, the tiny quit numbers found by Lotufo, Passos, and Czarnecki (2012) don't obviously correlate with the higher number of shooting stars users that Mamykina et al. (2011) found. While these results are not directly at odds with each other, they do not help us find a compelling story.

Without strong qualitative data, we will be unlikely to tell exactly why user drop-off was not much higher than expected.[11] *Stack Overflow*, and the INCREASED RESPONSIBILITY pattern in general, utilize endogenous goals that offer rewards that acknowledge and increase the motivations of *Power* and *Status*. Perhaps the endogenous aspect of the reward is effective at reducing the erosion of intrinsic motivation in this case, as the reward serves to enhance the experience that committed members have. Perhaps users focus on the introduction of *new* activities, rather than perceive INCREASED RESPONSIBILITY as a reward for old ones. Or perhaps those who engage with the product for so long become driven by community-minded desires such as *Idealism* and *Honor*, and INCREASED RESPONSIBILITY is introduced just at the right time to satiate them. Certainly, this indicates more research needs to be performed in this area if we are to ever find a resolution to the question of extrinsic rewards in software.

4.5   Pattern: Collection

Description: A means of collecting virtual items

Reiss desires: *Order*, *Power*, *Saving*, *Status*

Related to: CUSTOMIZATION, TRADING, BADGE

Examples: *CastleVille*, *Forza Horizon*, *iTunes*, *Pokémon*

Use: The motivation to collect is directly represented by *Saving*. Applications that offer *Collection* allow users to collect virtual items, either just for *Saving* alone, to meet

---

[11]Mamykina et al. (2011) does perform a small qualitative expertiment with six users, but these users aren't classified into the given classes, and more subjects would likely be needed to get any significant findings.

a *Power* need when building, completing or utilizing the COLLECTION, or to achieve *Status* by showing off the collection to others.

COLLECTIONS are most prevalent and easily identified in games.[12] *Pokémon* is the most obvious canonical example, where players "gotta catch 'em all." The joy of *Pokémon*'s collection mechanic is that it meets a number of Reiss desires all at once.

Saving Most obviously, collecting Pokémon taps into our need to *Save* and collect things.

Power By collecting Pokémon, players exert their *Power* over the game. Players travel around, locating areas where specific Pokémon can be found, then hunt for them in the grasses. Once discovered, the Pokémon's health needs to be whittled down until it can be captured. The effort expended in gathering Pokémon creates a feeling of accomplishment that would not be reflected if players could just buy them. Separately from the collection mechanic itself, collecting more powerful Pokémon allows players to become stronger in-game and progress through, leading to another satisfaction of a player's *Power* needs.

Status Collecting exciting Pokémon allows players to show off their collections to their friends in school playgrounds the world over, meeting the need for *Status*.

Order As a player's Pokémon collection grows, the spots in the list where Pokémon are missing become more and more prominent. Methodologically filling in the missing Pokémon fulfills her need for *Order*.

Curiosity Because Pokémon offer tangible in-game benefits, players get to explore their powers, how they match up in battles with other Pokémon, or seeing what they can evolve into by leveling them up.

---

[12]Loot-driven games, such as *Diablo 3*, are not included here as the goal isn't collecting, but rather the *Power* needs; items that aren't useful are sold by players. Games which offer greater scope for collecting and keeping sentimental items, such as *World of Warcraft*, do offer the COLLECTION pattern.

Games that utilize collection mechanics well take advantage of all of these needs, and broadly follow the same framework as Pokémon. Perhaps the biggest mistake that some collection mechanic implementations make is to remove the challenge of growing the collection. Too often items just unlock as part of a DARK PATTERN: GRIND, where players have to amass enough time with the game (or, perhaps, pay real money to unlock something) just to press a button in the UI that magically makes the item appear in the player's collection. Collecting Pokémon is challenging and even heart-stopping when a desired Pokémon appears. Every Pokémon is hard-earned, and the process of finding and capturing that Pokémon makes for a wonderful player-led narrative. *Forza Horizon* goes someway towards this by allowing players to race for an NPC's car, but that car is not special, and can be purchased by players in the game's marketplace at any time. Further, players need to have a competitive car to beat the NPC in the first place, providing no boost in *Power* once the rival car is aquired. Pokémon avoids this issue by using a rock-paper-scissors setup for how Pokémon battle each other (e.g. water beats fire, electricity beats water), and so beating a Pokémon might provide a boost to the *Power* of the entire team by slotting into a missing role, rather than just providing something the player already had.

While COLLECTIONS are most obvious in games, COLLECTIONS also appear with digital goods just as they do with real world goods. *iTunes* music libraries, for example, represent our music collection, and amassing a large library feeds the same motivational needs of *Saving*, *Status* and *Curiosity*. Music itself also fills the need for *Romance*. Appealing to our need to *Save*, *iTunes* offers a system called "Complete My Album," which lets users fill in the album gaps in their music collection. Similarly, *Plex* is an application that shows users video collections, and shows which gaps they have in TV seasons.

It is unclear whether applications that offer "jukebox" style subscriptions, like *Spotify* or *Netflix*, are part of the COLLECTION pattern as the entirety of the collection

is available at all times. *Spotify* lets users create playlists, and so one could argue that each playlist is the act of collecting certain songs together. Users could derive the same fulfillment from creating a collection of playlists they enjoy, although this seems tenuous.

Watch for: While we can collect things entirely for the sake of collecting, the act can be combined with a number of other motivations, as we saw in the Pokémon example. Implementations of collections that don't speak to other motivations are missing out on a big motivational draw they could offer. For example, take the COLLECTION system in *The Pioneer Trail*, shown in Fig. 4.7. The items that are collected appear as random drops from the result of taking actions such as chopping down a tree. These actions have a primary purpose, such as the collection of wood or removal of a tree that was in the way. Taking these actions costs energy, and it seems unlikely players will use up energy just to complete a collection. The collection items themselves are neither functional nor decorative and appear nowhere else in the user interface but the Collections window. When a collection is completed, a small token reward is given that is much more easily collected in other ways.

This collection mechanic only utilizes the *Saving* motivation, and offers nothing else. Players aren't even able to enjoy the beauty of the things collected by putting them down in the game world.[13] The COLLECTIONS can't be shared, and the completion of the COLLECTION doesn't offer anything unique, so no *Status* is gained, and they offer no in-game benefits either. Such a usage will only encourage those who are highly driven by *Saving*, but others will not find any intrinsic motivation to collect. Simple changes that provide a more meaningful extrinsic reward, be it just a decorative item or something that confers a greater gameplay bonus, would be more useful.

---

[13]Probably because this would impinge on the monetization strategy of the game.

Figure 4.7: A screenshot from *The Pioneer Trail* on *Facebook*, showing the collection screen. Here we see that if the "Family Collection" is completed, the player receive a small reputation boost, and a cosmetic Washboard item to decorate his yard. The items in the collection themselves have no intrinsic value. The Washboard at the time of writing costs 800 coins. The player character has 12,000 coins, and 7,500 coins costs $10.

4.5.1 Specialization of Collection: Badge

Description: Indicator that a user has performed a certain set of actions, or achieved a certain goal

Reiss desires: *Power*, *Saving*, *Status*

Also known as: Achievements, *Zichermann & Cunningham*: BADGES, *Crumlish & Malone*: COLLECTIBLE ACHIEVEMENTS, Trophies

Related to: COLLECTION, SCORE, TASK QUEUE, DARK PATTERN: GRINDING

Examples: *Foursquare*, *Khan Academy*, Xbox 360

Use: BADGES are small tokens awarded to a user for completing certain tasks, and are a specialization of COLLECTION. They gained widespread appeal at the introduction of the Xbox 360 which implemented achievements[14] and they have taken root as a symbol of gamification from both proponents and opponents. The pattern is termed "Badges" instead of "Achievements" to divorce the idea that such systems necessarily require any skill. "Achievements," as defined by Hamari and Eranti (2011) are "A challenge consisting of a signifying element, rewards and completion logics whose fulfillment conditions are defined through events in other systems (usually games)." However, when there is no obvious challenge, "achievement" is too strong a term. For example, *Reddit* provides a badge based on the time the user account has been open. This doesn't require overcoming any particular challenge, so "badges" is a more apt-name for this pattern.

The pattern itself is simple: one or more events occur, which are then rewarded via the feedback of the badge, which a user can display in her profile, perhaps as part of IDENTITY SHAPING. BADGE systems are employed to either reward certain behavior to encourage it, or as a means of validating that certain behavior is correct. The first case is most prevalent, and requires little explanation. In the latter case, an example can

---

[14]However, this was not the first gaming system to offer BADGES. (Jakobsson 2011) lists at least two other predecssors. The first was the Atari 2600, which had a BADGE system where a decorative fabric patch would be sent to players of some Activision games if they completed challenges listed in the manual and sent in a photo of their TV screen. The second was the MSN Games web portal (acquired by Microsoft in 1996), that offered BADGES to players.

be found on *Stack Overflow*, which uses badges such as "Self-Learner: Answered your own question with a score of 3 or more" to show that such actions are acceptable.[15] *Half-Life 2* does something similar, where the game communicates to players that they can complete the game by using the Gravity Gun alone by including an achievement for doing so.[16]

The big issue surrounding BADGES, from a motivational standpoint, is how they relate to our motivational needs, and what user behavior might result.

Zichermann & Cunningham: BADGES notes a number of motivations that BADGES could have:

> Although it's easy to forget, *Foursquare* did not invent badges. They've been around for a long time... the automotive industry [uses badges] to signal... what kind of driver is behind the wheel... In addition to signaling status, people desire badges for all kinds of reasons. For many people, collecting is a powerful drive. Other players enjoy the sudden rush for surprise or pleasure when an unexpected badge shows up in a gamified system. A well-designed, visually valuable badge can also be compelling for purely aesthetic reasons.

In this piece, we see *Status*, *Saving*, *Curiosity* and *Romance* motivations named. However, Zichermann and Cunningham (2011) is particularly obsessed with *Status* throughout, and Deturding (2011) rails against this *Status* primacy:

> Points, badges, and leaderboards are all feedback mechanisms games use to signal to a player how well she has done in overcoming challenges on the way to her goals. The joy comes from the realization that she overcame an interesting challenge, not from any extrinsic "reward value" of the point/badge/whatever.

This pushes the *Power* and *Curiosity* aspects of motivation, discounting the *Status* and *Saving* motivations that Zichermann and Cunningham propose.

---

[15]This was relayed by Joel Spolsky in a talk at Google's Mountain View campus.

[16]The Gravity Gun lets players pick up objects in the game and shoot them at enemies. The game is carefully designed to include enough objects in each encounter to allow the Gravity Gun to be useful for the entire game.

The truth probably lies somewhere in the middle: that badges mean different things to different people, and one of the strengths of the Reiss motivational framework is that it doesn't attempt to ascribe One True Motivation for everyone, but recognizes different people are driven by different things. One of the strengths of the BADGE pattern is that it does meet numerous different motivations, and so provides wide appeal.[17] A successful BADGES implementation would thus acknowledge this broad array of motivational needs that can be met, and aim to satisfy each of them as fully as possible.

The issue of achievements sapping intrinsic motivation via the overjustification effect caught traction from the presentation of Hecker (2010). Hecker wondered if this meant that BADGES were a harmful pattern, as "the intrinsic reward for knifing dudes is knifing dudes." Hecker synthesizes research into ways of making BADGES less susceptible to the overjustification effect:

Don't make a big fuss about them Placing emphasis on the BADGES diverts attention from the action to the reward.

Use unexpected rewards Lepper, Greene, and Nisbett (1973) found that intrinsic motivation appeared to be diminished only when the reward was expected. Unexpected rewards did not have this effect. Hecker notes that unexpected rewards are very difficult to generate when users seek *Social Contact* around their games. Xbox achievements can be secret until they are unlocked, but players will immediately share with others what these achievements are on sites like Xbox 360 Achievements (`http://www.xbox360achievements.org`). It seems reasonable to hypothesize that most Xbox 360 players don't visit such sites, so such rewards

---

[17]Deturding (2011) uses a reductio ad absurdum argument that if all we cared about were rewards, that a button which offered 1,000,000,000,000 points for clicking it would be the most engaging game ever. This unfairly frames the argument as only being based on the motivations with SCORE, and doesn't address the other, valid motivations that Zichermann and Cunningham (2011) brings up, like the aesthetic beauty of a badge. It is important to look at BADGES as a holistic whole, rather than poking at any one particular facet of motivation as not being enough.

work at intended for the majority of the player population. However, as noted below, unexpected rewards can not be part of goal-setting, so they don't help aid with *Power* needs.

Use absolute, not relative measures Avoid "grading on a curve."[18]

Use endogenous rewards Give rewards as part of the system. For example, if a player had done well using a sniper rifle, offer up a better sniper rifle as the reward. Most BADGE systems don't follow this rule, offering up BADGES and other rewards externally from the game.

Make them informational, not controlling Informational feedback is required for people to ascertain their position on their road to achieving *mastery* of a subject. Controlling feedback reduces our sense of *autonomy*, and thus our motivation to perform a task.

A survey of the badge systems used by Xbox 360 achievements, *Stack Overflow* badges and *Khan Academy* badges, shows that none of these meet all of these guidelines. All of them use exogenous rewards, offering points to add to a SCORE. Some *Khan Academy* badges are written in a controlling manner, *Stack Overflow* does not.[19]. None of them solely use unexpected rewards, although *Khan Academy* and the Xbox 360 Achievements system have rewards for which the actions required are hidden.[20] Each system tends to make "a big fuss" about badges, notifying the user when they are unlocked. To their credit, none use relative measures.

---

[18]It appears that Hecker took this from Kohn (1999), who suggests that grading is a system of rewards, and thus should be abolished. One aspect brought up in the book is that grading on a curve is, according to Milton, Pollio, and Eison (1986) a "failure to teach well." It's not clear that this opinion has any basis in the motivational psychology literature, but reserving achievements for a limited section of the populace may well violate SDT constraint that a challenge has to feel achievable. The worst offender in this case is likely the "World Champion (Multiplayer)" achievement from *Ghost Recon: Advanced Warfighter*, that is only awarded to players who reach number 1 on the global leaderboard.

[19]There are too many Xbox 360 achievements to perform an analysis of controlling language.

[20]Games on the Xbox 360 often use this functionality to hide plot twists.

While Hecker's guidelines are useful, it is important to note that Reiss rejects the argument for the overjustification effect as circular (Reiss 2005)—if the subject gets an extrinsic reward then stops the activity, intrinsic motivation is judged to have been lost; if the activity continues the subject is judged to be hooked on the extrinsic motivator. Reiss' argument, as well as other contradictory publications from multiple parties, led Medler (2011) to conclude that "There seems to be no clear cut argument for or against presenting extrinsic rewards as positive motivators for individuals, especially players."

It is possible, once users are fully engaged with a badge system, for them to transfer to stop caring about the aims of the application or game itself, and instead only interact with the badge system. Jakobsson (2011) describes how some Xbox players move from playing the individual game, to the meta-game of collecting achievements, which he dubs "Xbox Live Massively Multiplayer Online Game." Interestingly, he notes that the achievements then form a TASK QUEUE for these players, signifying quests they must fulfill to play the game.

Watch for: An important issue with the design of BADGES is that they encourage certain behavior, which can result in unintended consequences. In a talk at the University of California, Santa Cruz, game designer Nicole Lazzaro noted "Badges are systems and systems, hey, they're games." BADGES are systems, and each badge has an interplay with the others. She gave an example of a badge offered by *The Huffington Post* for posting comments on news stories, which could have led to an increase in spam.[21] In order to counteract this behavior, a badge was also offered for moderating posts, which brought the spam down again. It is important to ascertain all consequences, intended and unintended, once users begin to interact with BADGE systems. Readers who are

---

[21]One example of the frustration this can create can be found at *Giant Bomb*, that offered a badge for getting the first post on a news story. One commenter noted that "When people write shitty one word comments to try and get the first post 'achievement,' a rubber glove filled with sand should shoot out of their monitor and punch them in the face" (Scotto 2012).

interested in better understanding how rule sets and player actions work together would be served well by looking into game theory. Min-max game players show us that there are players who act economically rationally and extract the maximum reward, even when it's against the *intent* of the system (and in games, even if it's not any fun to do). Viewing BADGES with a game-theoretic eye will help to understand how these users may exploit the system, and what can be done to prevent (or at least balance) it out.

A particular danger with BADGE systems is they ascribe value to the tasks that they measure. The problem is that people are affected by "medium maximization." Hsee et al. (2003) found that when a medium (an arbitrary and valueless number, such as Gamerscore) is attached to certain tasks, an "illusion of advantage" is created. This leads to people attempting to perform a harder task with a greater value attached, even if the actual outcome is the same as a task with a lower value and that required a lower effort level.[22] The medium itself is worthless, but people become fixated on it. The sheer act of including BADGES is likely to induce some aspect of medium maximization.

Aside from the theory of medium maximization, BADGE systems are a way for a designer to communicate that certain behaviors are desired, in the same way SCORE is used.[23] This can be at odds with a particular user's goals, and even encourage negative experiences. Take, for example, *Mass Effect 2*, that includes an achievement for saving every team member in the final mission, which was described as a "suicide mission" throughout the game. Consider a first playthrough, where the player loses her robot companion Legion, to a rocket that hit him as he secured a doorway. His death acts as a turning point for the player and her courageous party, for whom success must

---

[22]They perform an experiment where users are offered vanilla ice cream for 6 minutes of effort, or pistachio for 7, and see what people do. When they assign points as well (60 for vanilla, 100 for pistachio), more people choose the pistachio task than they did previously. What's worse is that most users preferred vanilla to pistachio ice cream, so they actively sought a more negative outcome in order to maximize the medium.

[23]Or, in some cases, that certain behaviors are not desired. *Dead or Alive 4* offered zero point achievements for taking losses in the online mode, permanently attaching to a player's Xbox Live profile as a symbol of shame ("Dead or Alive 4 Achievements").

be achieved in order that his loss may not be in vain. On a second playthrough, all team members are saved, and the emotional arc of loss is no longer part of the game. The second playthrough is bereft of poignancy, and the ending potentially emotionally bankrupt. Assigning an achievement to this playthrough, even though it is less narratively satisfying, communicates to players that this is the ending they should desire. Assigning badges to experiences is always a poor choice. As Carvalho (2009) puts it:

> I don't like being reminded that I could have been "better" in successfully completing a story rich game. Which to me, is somewhat of an oxymoron. Most of my favorite gaming moments came from enjoying the experience, whether I did poorly or not.

Unfortunately, games on the Playstation 3 and the Xbox 360 require trophies and achievements for certification. One sidestep that some have chosen, such as *Heavy Rain* and *Spec Ops: The Line*, is to assign an achievement for each possible decision, so no implied value is given to either choice. This helps in some way, but this encourages players to replay the game to get the other badges. In the case of *Heavy Rain*, the lead designer expressly wanted players to only play once, so they own their experience (Sterling 2009). To replay would "kill the magic of it."

Assassin's Creed 3 has a particularly poorly implemented BADGE system. In the multiplayer mode, badges are awarded for performing certain actions or feats, such as "Use disruption to confuse and stun a pursuer" or "Escaped the most." These often fall under Hecker's "unexpected rewards," as *Assassin's Creed 3* features a dizzying array of achievable badges and items, listed in Table 4.3. Of particular note are the 128 different challenges and accolades that a player could be awarded after any given game, shown in Fig. 4.8. The player must dig through various menus to find what these unlocks actually are. There are two key issues with this system:

1. Rewards are offered so frequently that they become meaningless.

2. There are so many rewards that no clear goal is available.

Figure 4.8: A screenshot from *Assassin's Creed 3* on the Xbox 360. This screen appears at the end of a play session where the player leveled up. Along the top row are accolades that she received. Along the second and third rows are unlocked weapons, as well as an unlocked emblem and an unlocked title.

UNLOCK	DESCRIPTION	COUNT
CUSTOMIZATION		
Titles	Placed next to player name	246
Emblems	Icons placable on characters	190
Relics	Small items attachable to characters	9
Patron pictures	Avatars	81
ABILITY SETS		
Abilities	Active abilities players can use	14
Ranged Weapons	Usable weapons	4
Perks/Kill Streaks/Loss Bonus	Passive player abilities	26
PROGRESSION		
Levels	*Crumlish & Malone*: NUMBERED LEVELS rewarded for experience gained	49
Prestige	Awarded for completing 50 levels	99
Challenges	Perform certain actions	90
Accolades	Perform actions, judged relatively to other players in single game	38

Table 4.3: A table showing the number of unlockables in *Assassin's Creed 3*'s multiplayer mode. This table only lists the character-nonspecific unlocks. Each character that the player can choose has his or her own customizations, such as different costumes. Technically, the 49 available levels (players start at level 1, and move through to level 50, so only go through *49* levels) are iterated through on each Prestige, so there are 4851 levels a player can go through.

This is another instance of the prospect theory application provided in the discussion of cognitive evaluation theory (CET), which notes that our evaluations of the worth of something is reference-dependent. As we are rewarded, the reference point shifts upwards, and we evaluate the next rewards based on that reference point.[24] A constant deluge of rewards means we simply won't value them at all. The second issue is addressed by Rigby and Ryan (2011), who write "We need to have clarity about the goal at hand. If we are uncertain about what we are undertaking, it is harder to act meaningfully or to feel a sense of mastery, even if we succeed." The sheer number of possible challenges offered by *Assassin's Creed 3* makes it very unlikely that any player keeps more than a couple in her head at a time, if indeed she tries at all. This is compounded by the first issue: if the player is always rewarded anyway, why would she bother taking note of any particular goal?

BADGE systems run the risk of becoming DARK PATTERN: GRINDS, and so should be carefully designed to reward interesting, meaningful actions, instead of "Find n of x." Particularly good achievements in video games help point out exciting new ways to play, such as climbing the Agency Tower in *Crackdown*, or completing *Half-Life 2* with only the Gravity Gun.

4.6 Pattern: Growth

Description: Provide ownership of something that is to be tended to, which grows and transforms over time

Reiss desires: *Curiosity, Order, Power, Saving*

Related to: BROADCAST, COLLECTION, IDENTITY SHAPING, CONSTRUCTION (Lundgren and Björk 2012), NURTURING (Lundgren and Björk 2012), *Zichermann & Cunningham*: NURTURING, GROWING

---

[24]This issue is analyzed in the context of social games by Hamari (2011).

Examples: *Animal Crossing, Nintendogs, Sim City*

Use: The act of building or cultivating something comes under the heading of Growth. Examples include building cities in *Sim City*, tending to orchards in *Animal Crossing*, or caring for a virtual pet in *Nintendogs*.

Lundgren and Björk (2012) previously described two separate gameplay patterns:

Construction The action of introducing or rearranging game elements to create structures in game worlds.

Nurturing Taking care of game elements in order to see them evolve or develop.

From a Reiss perspective, both of these relate to *Power* and *Order*. These patterns combined into this single motivational pattern, termed Growth.

Growth differs from Customization by including one or more of these three elements, as defined by Jose Zagal in a personal communication.

Delayed time effects Do something now, see the effects later.

Liveliness The simulation continues to run when the user is gone.

Non-determinism Some element of non-determinism is involved so that the results of Growth is not fully known.

Certain mechanics in *Animal Crossing* include the *Delayed time effects* and *Liveliness*, but not the *Non-determinism*. For example, when the player buries an apple, the apple tree will not grow until later (delayed time effect), but the player does not need the system to be on for the tree to have grown the next day (likeliness). Thus, planting fruit is a use of the Growth pattern. However, when the player is changing the wallpaper in her house, none of the three elements exist, and thus is a form of Customization.

Figure 4.9: A screenshot from *The Pioneer Trail* on *Facebook*, showing a particularly messy play space. These weeds and skulls are part of a GROWTH system that generates them over time. Clearing the weeds offers coins, experience and other drops from the "Clearing" COLLECTION.

GROWTH feeds into our need for *Power*, as we show our ability to influence our environment around us.[25] Tending to something and cleaning it up fulfills our need for *Order*. *The Pioneer Trail*, for example, specifically addresses this motivation by steadily generating weeds and trash in a player's play space, that the player must go around and clean up, as seen in Fig. 4.9. While it's frustrating to see the trash when you first log in, it feels much better once it's cleaned.

The delayed time effects and liveliness aspects of GROWTH set up implicit reinforcement schedules. This primes users with times when they should return to the simulation, so that they can satisfy their *Curiosity* about the results of their efforts. It's little wonder that so many social games, which rely upon small amounts of re-engagement over a long period of time, almost always have some GROWTH pattern within them.

---

[25] A real world example would be a Lego set, which Reiss specifically notes as being part of the *Power* motivation (Reiss 2008).

Chapter 5

Social patterns

5.1 Overview

Social patterns are those that offer us the chance to interact with others, providing *Social Contact* needs. While the rush to add social elements to applications seems unabated, the fact is that *Social Contact* is but one of the sixteen desires that Reiss believes drive us. We aren't as wrapped up in *Social Contact* as it seems. The telephone is a pure *Social Contact* offering, but it seems more and more like its the least used part of our smartphones.

Instead, we dip in and out of contact, leaving BROADCASTS to others via text message and *Facebook*, returning to consume SOCIAL FEEDBACK when our ego requires it. All of the patterns here are primarily used in asynchronous communication, freeing us from immediate obligation to others; obligations we keep so that we feel *Honorable*, but don't bring us enjoyment when we have no need for *Social Contact*.

Use social patterns when the value of your application is as a platform for users to BROADCAST to one another, or when utilizing the social graph allows the application to grow through word-of-mouth.

5.2 Pattern: Contact List

Description: A list of contacts

Reiss desires: *Acceptance*, *Power*, *Social Contact*, *Status*

Also known as: *Crumlish & Malone*: BUDDY LISTS

Related to: BROADCAST, DARK PATTERN: IMPERSONATION

Examples: *Address Book*, *Facebook*, *Twitter*, Xbox 360

Use: The CONTACT LIST is the linchpin of all social elements in an application, and is always coupled with BROADCAST.[1] Figs. 5.1 to 5.2 show typical CONTACT LISTS.

CONTACT LISTS can be interpreted by users as showing their *Power* ("how many people can I influence?"), their *Status* and *Acceptance* ("how many people like me?") and provides a means for *Social Contact* via BROADCAST. While it is likely some users find a larger number of contacts to indicate a greater satisfaction of these motivations, it is not a SCORE, as the number of contacts is not an arbitrary number.

In order to avoid bootstrapping new social networks, many apps choose to import CONTACT LISTS from other services. Some applications or platforms, like the Playstation 3, contain a native CONTACT LIST for displaying other Playstation 3 contacts that they can play with, but also allow BROADCAST to other networks, such as *Facebook*.

Watch for: This pattern doesn't use the term 'friend,' as often contacts made on networks such as *LinkedIn* are merely acquaintances. Notably, *Twitter* dropped the use of the word 'friend' (Kalucki 2011b) in favor of the clumsier, but more accurate, 'who you follow' (Kalucki 2011a). In fact, the use of the word 'friend' creates confusion in users, reducing their *Tranquility* and quality of *Social Contact*, as found by Tokunaga (2011):

---

[1]It is difficult to think of a FRIEND LIST implementation where BROADCAST was forbidden. The closest may be a telephone directory, but even this is designed around BROADCAST by telephoning people, it's just that the medium of the list is different to the medium of the broadcast.
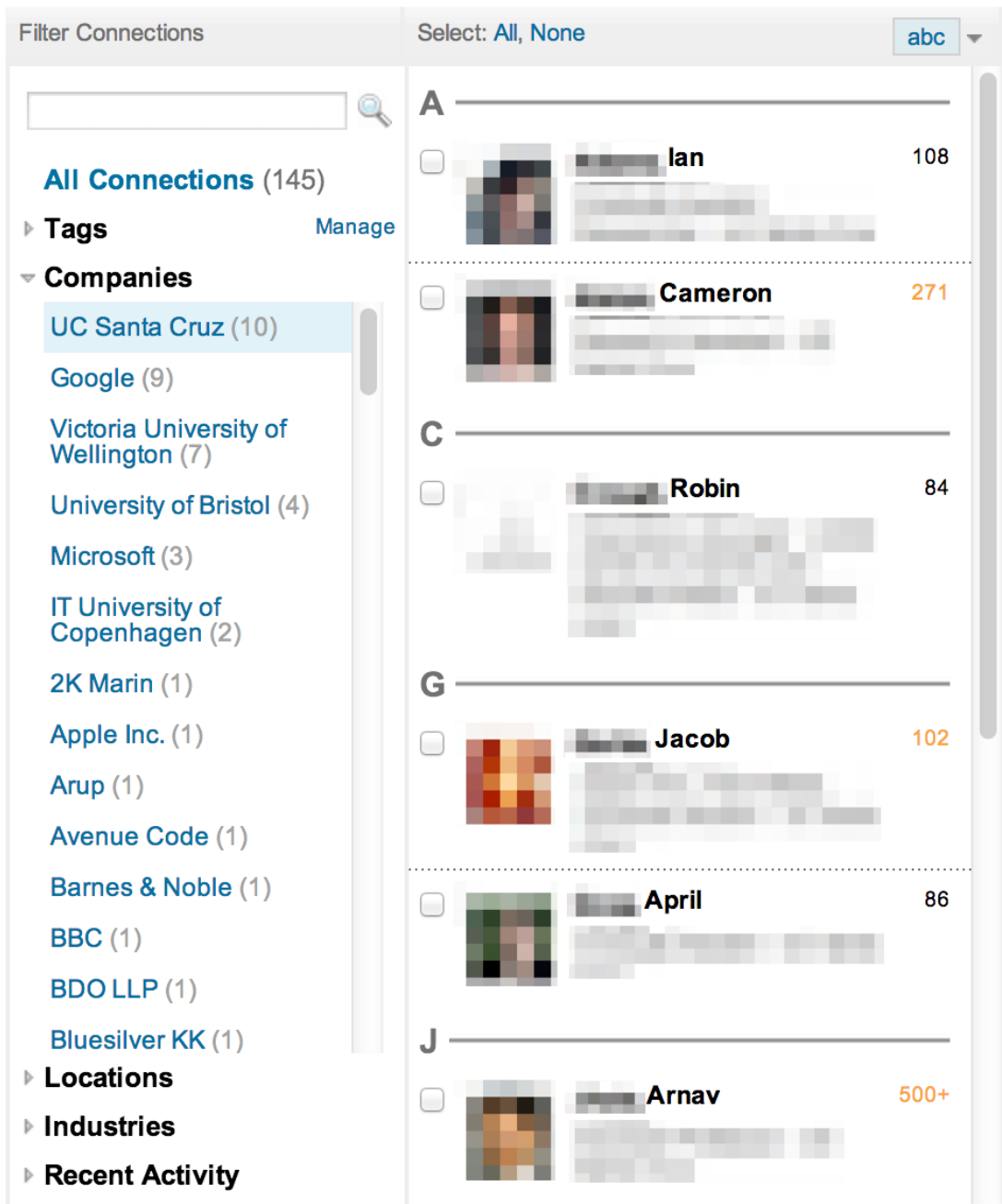
Figure 5.1: A screenshot from *LinkedIn* showing a CONTACT LIST. The numbers to the right of a contact indicate the number of connections he or she has. This is part of *LinkedIn*'s core use case, which is to find and explore your social network to create new connections. Those with many connections are more useful in this endeavor. Avatars and last names obscured for privacy. Taken on February 7, 2013.

105
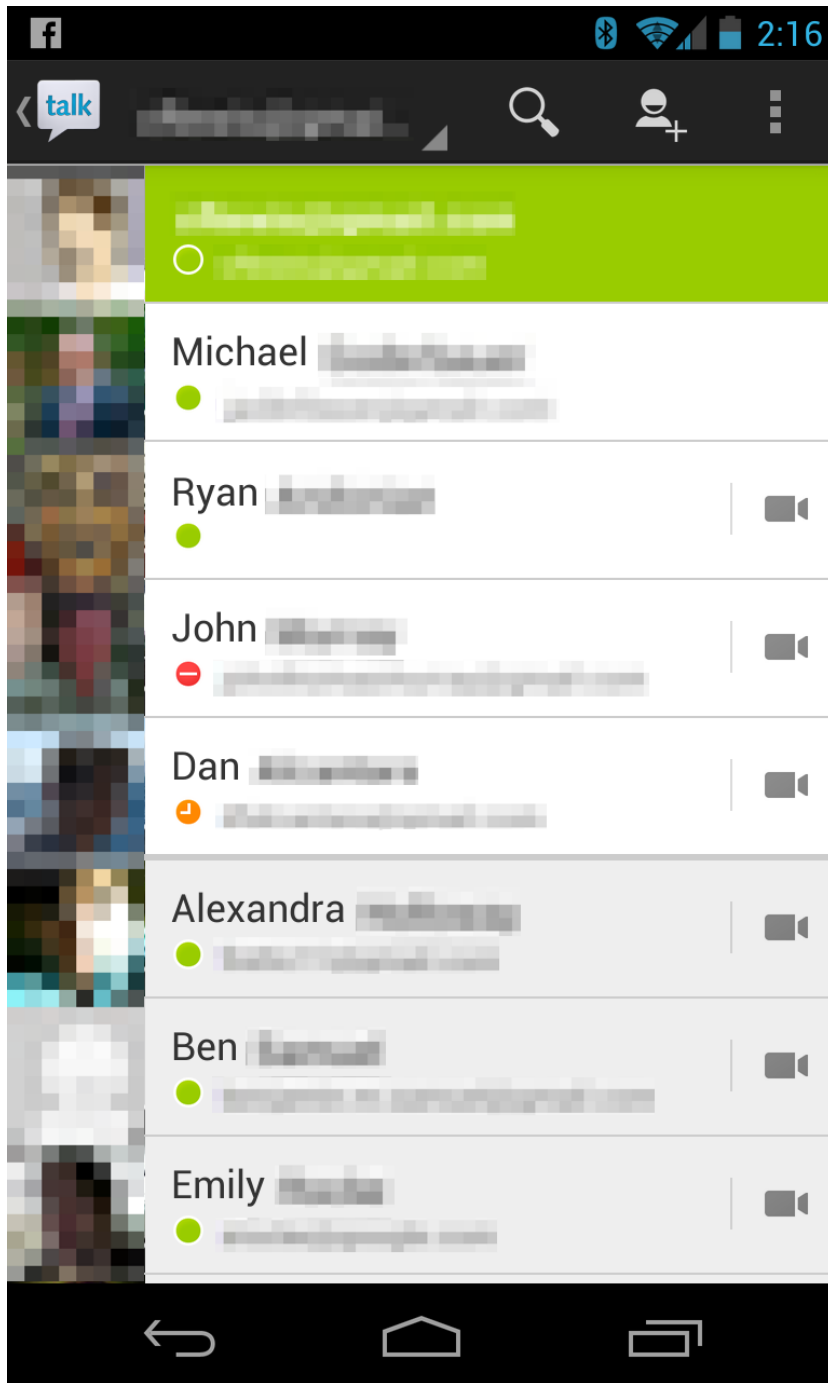
Figure 5.2: A screenshot from *Google Talk* on *Android*, showing a CONTACT LIST. Avatars and last names obscured for privacy. Taken on February 7, 2013.

The confusion surrounding the definition of friends on Social Network Sites (SNS) complicates matters further in the friend negotiation and ranking processes. Because the equivocal term "friend" is used on SNSs, there are assumptions carried with the label, which may escape some users.

Individuals diverge in how they interpret the meaning of friends on SNSs; some use it to mean mere contacts, others only use friends to refer to people they have met offline, and there are those who apply the term to only close friends. The way in which people construe the notion of friends on SNSs determines their actions in friend negotiations and rankings.

Interpersonal strain may result when two people use and act on discrepant meanings of friends.

Further, Dunbar (2011) claims Dunbar's number—the maximum number of people you can actually be friends with—to be 150, due to a direct limitation of the neocortex size in humans. However, in a survey of 269 *Facebook* users, Hampton et al. (2012) found that the average user has 245 contacts, likely further feeding the confusion of users. All this means the terminology applied in the deployment of a CONTACT LIST is important. If the CONTACT LIST is being imported from another social network, the same term for contacts should be used (eg. 'friend' for a *Facebook* list, 'contact' for a *LinkedIn* list, 'who you follow' for *Twitter*) for consistency. Otherwise, in order to ensure users are able to properly understand the context they are making relationships and not undermine their motivations, careful consideration between the use of the word 'friend,' 'contact,' or even less, should be taken.

CONTACT LISTS, while being sources of useful information, can be easily abused (accidentally or not), and combined with BROADCAST, enables DARK PATTERN: IMPERSONATION.

5.3   Pattern: Identifiable Community

Description: Discussion-supporting features where one or more communities of people can come together, with recognizable social norms

Reiss desires: *Honor*, *Social Contact*

Related to: *Crumlish & Malone*: GROUP CONVERSATION

Examples: *Reddit, Mailing Lists, Usenet*

Use: IDENTIFIABLE COMMUNITIES are areas where users can see that there are one or more social groupings, and derive the social norms expected in those groups, either via social proof or more explicit documents, such as FAQs. Many apps have a community around them or within them (as discussed in Section 5.1). IDENTIFIABLE COMMUNITIES are implemented using commenting, forums and newsgroups. This pattern only includes implementations where a community is clear to visitors. For example, *Amazon* reviews allow for discussion via SOCIAL FEEDBACK, but there is no identifiable *Amazon* community. In another example, *Facebook* groups are IDENTIFIABLE COMMUNITIES, but the *Facebook* news feed is not.

IDENTIFIABLE COMMUNITIES develop their own social norms, in-jokes and memes.[2] It is common for large community areas to divide up into smaller groups to discuss topics: Usenet was setup in 1980 and used newsgroups for this purpose, and *The WELL*, one of the oldest communities that started as a BBS in 1985, utilized "conferences" to split up users into discussion areas such as music or business (Hafner 1997).

Research has shown that successful IDENTIFIABLE COMMUNITIES have a maximum of about 500 members before they must be subdivided. Arguello et al. (2006) summarizes research into online group sizes:

> People tend to be less committed to larger groups and to contribute less to them (Karau and Williams 1993; Oliver and Marwell 1988). In addition to the effects of size per se, the size of a group influences the amount of communication in them. Overall communication volume seems to have paradoxical effects on reading and participation in online communities. On one hand, empirical research has shown that higher communication volume lowers return rates in online groups (Butler 2001; Jones, Ravid, and Rafaeli 2004),

---

[2]Memes were proposed by Dawkins (1976) and originally referred to ideas and behaviors spreading throughout a culture. Now, the Oxford American Dictionary lists alongside this definition the more common use: "an image, video, phrase, etc. that is passed electronically from one Internet user to another."

consistent with the information overload argument. On the other hand, network externality and critical mass theories imply that online groups need a minimum volume of message traffic to draw and retain members ("Systems Competition and Network Effects"; Markus 1987). Too many messages, and people may not return to participate; too few, and it will be difficult to maintain the community responsiveness needed for successful interaction.

Schlack (2011) grounds this out to actual numbers. Schlack found that in communities of 300–500 people, 64% contribute each month. This contrasts to large public communities, where "90% of users are lurkers who never contribute, 9% of users contribute a little, and 1% of users account for almost all the action" (Nielsen 2006).[3] This means that 200,000 visitors would be required to participate as much as a 400 person community (A. 2011).

Given this evidence, it can be concluded that *Social Contact* is higher in smaller communities, where relationships between users can grow, provided they are busy enough to maintain momentum.

Watch for: When FILTERING is used to allow users to find suitable IDENTIFIABLE COMMUNITIES, there is a risk that the user population will be overly divided. If the overall user community is too small, FILTERING may put users inside tiny filter bubbles with a lack of activity where the critical mass to initiate long-term social interactions has not been achieved (Markus 1987) (for more on filter bubbles, see the FILTER pattern in 7.1).

Ludford et al. (2004) found that users in their experiment were more likely to contribute to a movie forum when they were joined together with dissimilar people[4]; but they note that Preece (1998) found that some communities had members that sought support rather than friction. While some users seek out conflict and debate, the findings of Preece, and fears of Pariser's filter bubbles, lends credence to the possibility

---

[3]This is known as the 90-9-1 rule, and dates back to the early 90s (Nielsen 2006). What's fascinating about the 90-9-1 rule is how durable it's been over the last twenty years. As Nielsen puts it: "How to overcome Participation Inequality: You can't."

[4]They theorized this was because difference of opinion spurred conversation, as illustrated in the classic XKCD comic: "I can't [come to bed], this is important. Someone is *wrong* on the Internet."

that many won't. These self-made filter bubbles, where users filter their surroundings in order to feel *Tranquility* may be at the cost of users experiencing deeper *Curiosity*. Injecting content from other communities at some point in a user's flow may help break down bubbles, and allow for greater exploration and DISCOVERY.

5.3.1 Specialization of Identifiable Community: Meta-Area

Description: A place for community members to guide the product, which contains one or more IDENTIFIABLE COMMUNITIES

Reiss desires: *Idealism*, *Independence*, *Power*, *Status*

Related to: *Crumlish & Malone*: FORUMS, IDENTIFIABLE COMMUNITIES, INCREASED RESPONSIBILITY

Examples: *Stack Exchange Meta*, *Wikipedia:Village pump*, *World of Warcraft Forums*

Use: META-AREA is a specialized IDENTIFIABLE COMMUNITY where users offer guidance and discussion of a product. One of the conclusions drawn by Mamykina et al. (2011) on the popularity of *Stack Overflow* was the inclusion of a META-AREA, but they analyze the feature from the perspective of the developers:

> The design team established a continuous feedback loop with their users. A forum for discussion about the site, but external to it (`http://meta.stackoverflow.com`) helped the founders understand challenges and concerns of their users, and prioritize feature requests.

While engaging with the community clearly helped the development team work more effectively, this interaction is also deeply rewarding to the users that participate in it. They feel their *Power* by influencing the direction of the platform, some who continually make good recommendations might get a *Status* boost. Users feel like they have more choices in what the software can provide giving them *Independence*, and, finally, those who are community-minded feel like they are improving the community around the software, feeding their need for *Idealism*. The META-AREA is a powerful,

and yet overlooked, pattern for creating long-term engagement across order months and years, rather than minutes or hours.

Jeff Attwood, one of two co-founders of *Stack Overflow*, found META-AREAS so important that he left to create a new application, *Discourse*, that provides forum software for developers. Attwood (2013) uses a fictional example conversation to justify the venture:

Developer: We'd love to get your expert advice on our thing.
Attwood: I probably don't use your thing. Even if I tried your thing out and I gave you my so-called expert advice, how would it matter? Anyway, why are you asking me? Why don't you ask your community what they think of your thing? And if you don't have a community of users and customers around your thing, well, there's your problem right there. Go fix that.

META-AREAS are places where IDENTIFIABLE COMMUNITIES are built, regardless of whether the application itself supports any social features. META-AREAS have been common in games for some time, particularly for Massively Multiplayer Online games, that are constantly evolving. The *World of Warcraft Forums* [5] are the archetypal video game forum, and heavily trafficked.[6] Smaller games often have forums as part of their publisher's website. For example, one can find a place to discuss *The Misadventures of PB Winterbottom*, a small game for the PC and Xbox Live Arcade, on the *2K Forums*.

When users are not offered a META-AREA, they may take to other applications in order to try and make contact with leadership to express their views about the product and to fulfill their *Power* needs. GOOGLE+, for example, has a figurehead in Vic Gundotra, the Senior Vice President who oversees its development. He regularly uses *Google+* to post photos and news articles, and engages with the community at large.[7] Users look to Gundotra whenever they have issues with Google products,

---

[5] http://us.battle.net/wow/en/forum/

[6] At the time of writing, the first page of the "General Discussion" shows around 40 threads, but the oldest touched of these is only 30 minutes ago.

[7] This contrasts with Mark Zuckerberg, whose *Facebook* profile does not contain any public posts at the time of writing.

regardless of whether they are under his jurisdiction, even adding hashtags and looping in other people using the + signifier: "I'm really disappointed that Google pushed Chrome to be stock when it's clearly not ready so shame on you #Android #N4 #N10 #N7 #Google #StillloveGoogle #GooglePlay +Android +Vic Gundotra +Google Play +Hugo Barra,"[8] "Google + mobile version (Android) still doesn't recognize the markup tags like strikethrough . . . hope a fix comes for this one of these days. +Vic Gundotra,"[9] "Dear +Vic Gundotra, Is there a technical reason why #GoogleNow can't be a simple app for #Android users running #ICS and earlier to download? Or is this some kind of 'plan' that +Google has?"[10]

Watch for: The danger of the META-AREA is that it only provides the *Toxboe*: ILLUSION OF CONTROL, rather than a way for users to actually change the application. With a large number of users, they will struggle to find a voice and feel like they are being heard, preventing the fulfillment of any of the motivational needs they expected to, leading to frustration. Blizzard, who is one of the most communicative companies with their user base, is often accused of not listening to their community. Greg Street, more commonly known as "Ghostcrawler", is a game designer on *World of Warcraft*, and posted to Twitter ("Ghostcrawler: "Blizzard doesn't listen to. . .""").

> "Blizzard doesn't listen to our feedback," always astounds me. Most of our changes are from feedback. With 10M players, we get a lot!

Replies from users came back including "But you are terrible at communication", "you never listen for forums feedback discussions" and "Could you be more full of shit? YOU guys listen to the changes YOU WANT TO, and then say .. 'yeah, we listen to our fans"'[11]

---

[8] http://goo.gl/AkK5Y

[9] http://goo.gl/6FPWK. This was actually fixed in a subsequent *Google+* release.

[10] http://goo.gl/LZoKh

[11] This last one provoked a response from Street stating "I don't block many people, but the time has come."

The toxicity of game forums from frustrated fans is not unique to Blizzard. In Gaider (2013), the lead writer of Bioware's *Dragon Age* series notes a similar trend.

> The overall tone of the forums has become increasingly toxic. I tend to largely avoid them these days, myself. Why? Because spending too much time there starts to make me feel negative—not just about the games we make, but about myself and life in general. That's not a good feeling to have. I'm sure there are folks there who would bristle at that comment, suggesting that all negative feedback is justifiable and that ignoring it is the equivalent of us sticking our heads in the sand. How will we ever improve unless we listen to their scolding and take our lumps like good little developers? That is, of course, ignoring the idea that we haven't already digested a mountain of feedback—both positive and negative—and there's really only so much of it you can take. Eventually you make decisions (informed by that feedback, though only in part—it can only ever be in part) and move on.

This indicates that discussion forums only scale to a certain level, at which point users feel lost in the shuffle. Angry posts lead to the broken windows effect, where visible violation of behavioral norms leads to further violation by others (Kelling and Wilson 1982; "Can the can" 2008). Kottke (2008) theorizes that in online communities, "Undeleted hateful or ad hominem comments are an indication that that sort of thing is allowable behavior and encourages more of the same." However, heavy-handed moderation reduces participants feelings of autonomy. How does one resolve the quandary?

The *Stack Exchange* model has already solved this problem, at least in the context of gaining user feedback.[12] Fig. 5.3 shows a screenshot of the most upvoted questions tagged `[world-of-warcraft]` on the *Gaming Stack Exchange* site. We see questions from users about how to best traverse the world, what is acceptable play, and what do to if an account is hacked. One can easily imagine this model transposed to gamer requests for changes to cooldown timers or addition of flight paths, and Fig. 5.4 shows a mockup of what that might look like. The most desired features are voted to the top, *Stack Exchange* purposefully collapses duplicate questions together (negating

---

[12]In the context of open discussion, the *Stack Exchange* model would be a poor choice. The model purposefully suppresses discussion in order to float up important questions and important answers (Attwood 2013).
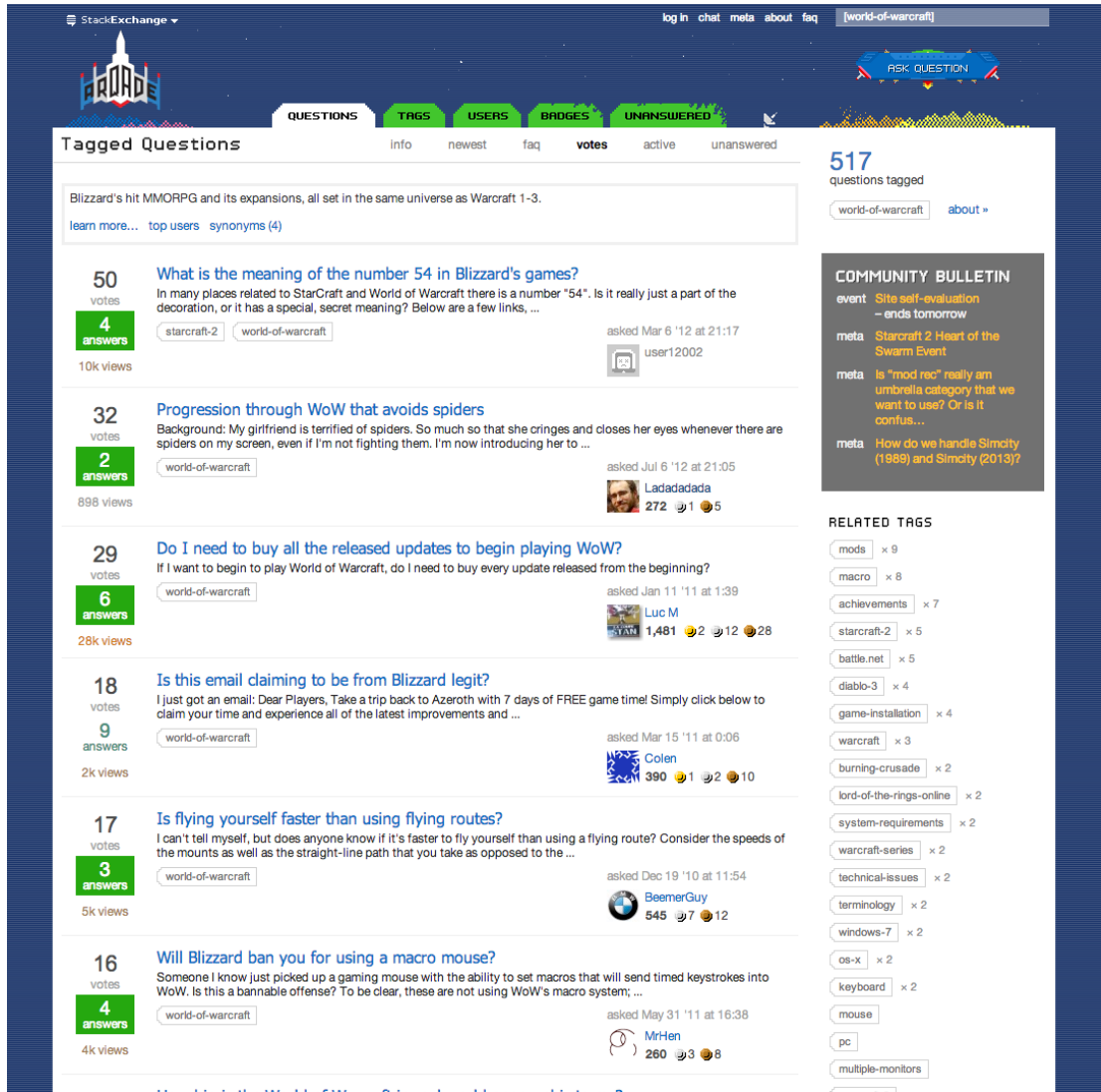
Figure 5.3: A screenshot from *Gaming Stack Exchange*, that shows the most upvoted questions tagged [world-of-warcraft]. Taken on February 7, 2013.
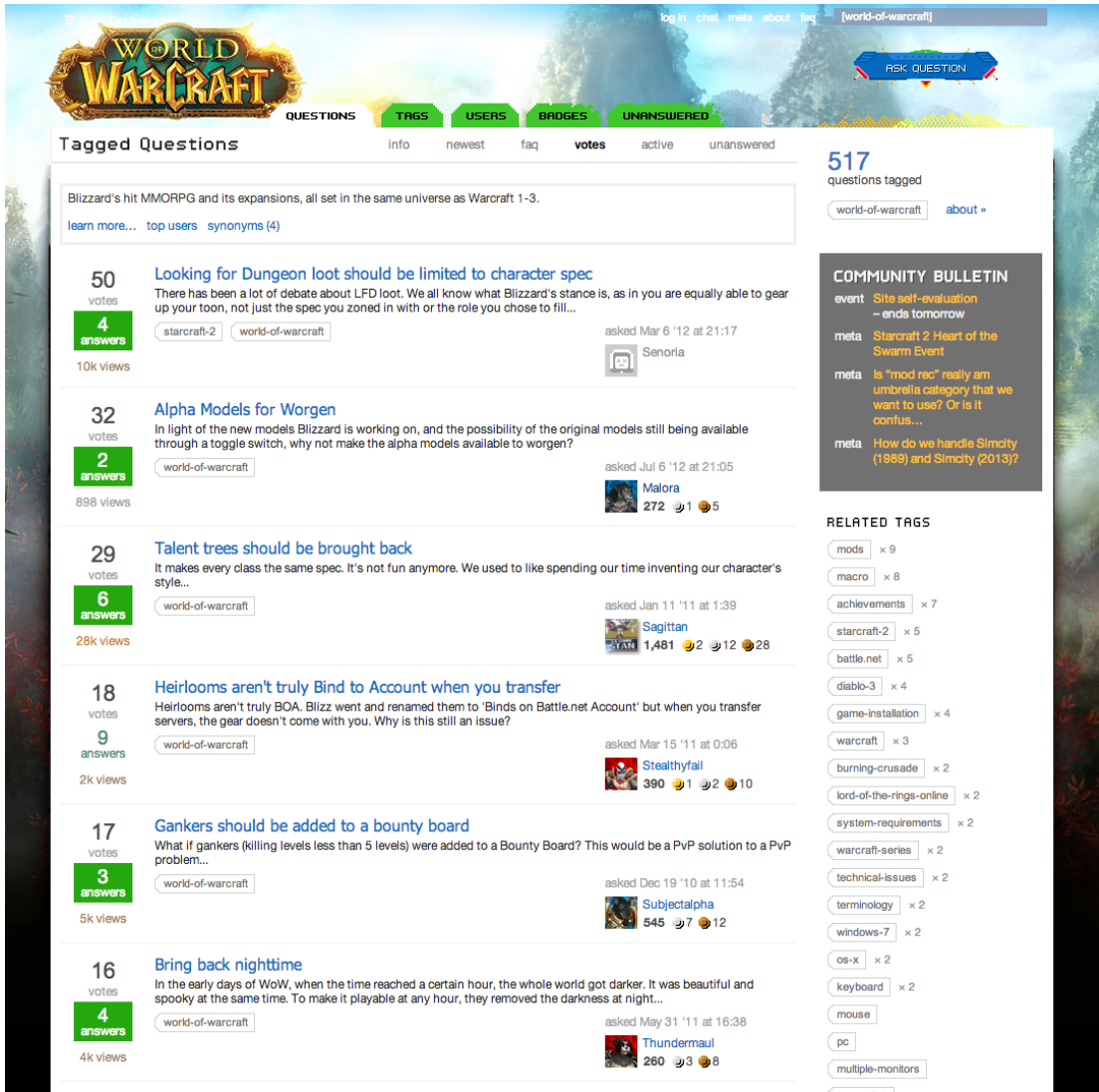
Figure 5.4: This is a mockup of what a screenshot of *Stack Exchange* deployment might look like using questions taken from the General Discussion forums of *World of Warcraft*.

developers having to dive into multiple threads to give similar answers), and questions with significant upvotes could be answered by developers and then closed.[13] *Stack Exchange* utilizes a very heavy hand when moderating, but does so without removing the *intent* and the *volume* of what users are trying to communicate to developers. Their needs of *Power*, *Status* and *Independence* are met, without the broken windows effect.

META-AREAS are one means of interacting with users, and, as such, should be treated as areas where normal maxims of customer service apply. This is usually adhered to, but when combined with INCREASED RESPONSIBILITY, which gives privileges to ordinary users, it is easy for the company messaging to get out of hand. Dropbox found this out to their cost when users complained on their forums that email addresses that they only provided to Dropbox were receiving spam, leading them to believe Dropbox had either sold the email address or had its security breached (F. 2013). Dropbox had given users external to their company "Moderator" privileges, which gave these users the impression of authority. Some particularly curt responses from these moderators, such as "Your email likely wasn't leaked," and "Just the fact that you listed your emails says it all," forced Dropbox into issuing an apology (B. 2013):

> We want to apologize for some of the dismissive responses from our volunteer moderators—since they aren't employed by Dropbox, they don't have visibility into issues like this. We want you to know that we've taken these reports seriously.

5.4 Pattern: Broadcast

Description: Users are able to share information with others

Reiss desires: *Independence*, *Power*, *Romance*, *Social Contact*, *Status*

---

[13]Perhaps a guarantee of the threshold of upvotes required for an official response could be utilized, such as with the White House Petitions site. Even with questions being upvoted, without guarantee of response, the community could still feel unheard, as with the infamous Woody Harrelson Ask Me Anything on Reddit. Harrelson only answered questions about his upcoming movie, and failed to answer the top questions, enraging Redditors ("Woody Harrelson Reddit AMA" 2012).

Also known as: *Crumlish & Malone*: Broadcasts

Related to: Activity Stream, Contact List, Identity Shaping, *Crumlish & Malone*: Statuscasting

Examples: *Facebook*, IRC, *Twitter*

Use: When users transmit a message that can be seen by one or more people, they are creating a Broadcast. While messages are often textual, such as email, newsgroups or internet chat, they can contain any media, such as pictures or sound. Posting a photo to *Flickr* is a Broadcast, a new recording uploaded to *SoundCloud* is a Broadcast, posting a trophy from *Uncharted* to *Facebook* is a Broadcast. While the Broadcast pattern is quite general, it does have specific boundaries.

- The content must have the capacity to be visible by one or more people, not including the original sender. If no-one else can see it, then no Broadcast has been made.

- Even if no-one actually sees the Broadcast, it is still one if the capability to see it is there. Sending a tweet, even if no-one follows the user, is still a Broadcast, as long as one or more users *could* see it if they tried to access that content.

- A Broadcast only applies to the act of sharing, not the creation of the content. Taking a photo is not a Broadcast. Posting that photo to *Facebook* is.

- A Broadcast doesn't imply permanence. Taking a photo with *SnapChat*, which is automatically deleted once its recipient sees it, is a Broadcast. Similarly, Broadcasts can be edited at will.

- Users don't need to specifically authorize or initiate any individual Broadcast. For example, *Spotify* automatically Broadcasts each song that a user has heard to *Facebook* if given ongoing permission to do so.

BROADCAST is a pattern that is so prevalent that it is almost invisible to us now.[14] BROADCAST, of course, has been baked into the network communications since the beginning: bulletin board systems, moving on to email and newsgroups. However, the World Wide Web used to be a largely static place, where only those with the knowledge of HTML were able to BROADCAST by putting up Web pages.[15] The growth of blogs and forums democratized the ability to BROADCAST to much wider audiences than email or chat clients, and the explosive growth of social networking cemented this pattern as a core element of interaction design.

BROADCAST is, as might be expected, a key component of any *Social Contact* need. Without a communication means, no contact can be made. Less obvious is that BROADCAST is also a means of influencing others, and so can satisfy *Power* issues. BROADCAST allows users to express themselves and their *Independence*, and maybe even find *Romance*.

Watch for: BROADCAST is, in and of itself, likely not as powerful as it first appears. BROADCASTING is an important part of communication, but so is SOCIAL FEEDBACK. Many of the BROADCASTS users perform are with the implicit assumption that some SOCIAL FEEDBACK will be returned to them, otherwise they can feel like they are just "shouting down a well," reducing their sense of *Power* and *Social Contact*. At worst, they may feel stuck in a DARK PATTERN: SOCIAL PYRAMID SCHEME.

One option is to have BROADCASTS transmitted to somewhere where the user already has a social graph set up, increasing the likelihood of some SOCIAL FEEDBACK from their contacts. However, while this may help the user who is doing the BROAD-CASTING, those who receive the messages may not be so pleased. The possibility of

---

[14]In Tidwell (2010), Tidwell retires patterns that she deems obvious. It seems worthwhile to write about as many patterns as possible in this early stage of understanding what motivational design patterns are.

[15]Sites such as *Geocities* and *Fortune City* offered space to host those pages, removing some of the burden of BROADCASTING. The neighborhood metaphor that these sites used was particularly enjoyable, and getting a good house number in the right neighborhood would lead to a nicer URL.

spamming—either from DARK PATTERN: IMPERSONATION or specifically triggered by the user herself appears—and thus strong FILTERING controls are required to ensure users feel like their ACTIVITY STREAMS are as they wish. Without FILTERING, users will experience a drop in *Order* and *Tranquility*.

5.4.1   Specialization of Broadcast: Social Feedback

Description: A means for people to receive asynchronous feedback from others

Reiss desires: Receiver: *Acceptance*, *Power*, *Romance*, *Social Contact*, *Status* Giver: *Honor*, *Idealism*, *Romance*

Also known as: *Toxboe*: LIKING, *Direkova*: SOCIAL FEEDBACK, *Crumlish & Malone*: SOLICITING FEEDBACK

Related to: *Bjork & Holopainen*: ALTRUISTIC ACTIONS, *Toxboe*: SOCIAL PROOF

Examples: *Facebook*, *Reddit*, *Yelp*

Use: SOCIAL FEEDBACK is a broad pattern that relates to how people provide BROADCASTS as feedback to others. This could be in the form of a 'Like' (*Facebook*), an upvote (*Reddit*, *Hacker News*), an 'I found this helpful' (*iTunes*, *Amazon*, *Yelp*), endorsement (*LinkedIn*), a comment or other acknowledgment that is handed out by others. SOCIAL FEEDBACK is almost always in response to a specific BROADCAST, but does not have to be. In some cases, the system itself attempts to elicit SOCIAL FEEDBACK without a broadcast, such as in the case of *LinkedIn* endorsements, or requests from *Facebook* or *Google+* to wish a user a happy birthday. This specialization is worth studying as it creates fulfills motivational desires for both the sender and the recipient.

Those who offer SOCIAL FEEDBACK also gain a feeling of *Social Contact*, as well as meet their desire to feel *Honorable* by engaging in expected social behavior, and *Idealism* by extending altruistic gestures. Those who receive SOCIAL FEEDBACK can feel influential (*Power*), raise their feeling of self-importance (*Status*), increase interactions with others (*Social Contact*) and feel accepted by the group (*Acceptance*). SOCIAL

119

FEEDBACK can even be used as a means for satisfying *Romance*, such as *Facebook*'s 'Poke', which is both ambiguous and innuendo-laden (at least in English).[16] *Zichermann & Cunningham*: FLIRTATION AND ROMANCE also describes interactions such as 'winking' or 'flirting' on dating sites.

There are two types of SOCIAL FEEDBACK: structured, and unstructured.

Structured Structured feedback is when the form of the feedback is offered by the system, and usually only takes one click to perform: liking, poking, upvoting, offering compliments (*Yelp)*, marking as helpful, all come under this heading. Structured feedback is not a SCORE, but is quantifiable and can be shown to users in aggregate.

Unstructured Unstructured feedback is a free-form response, usually via a text box. Commenting or replying would come under this heading. Unstructured feedback may itself receive its own SOCIAL FEEDBACK. For example, *Facebook* comments can be liked, and *Reddit* comments can be upvoted. This creates a feedback loop which encourages continued contribution from involved parties.[17]

Structured and unstructured feedback are not mutually exclusive, and can be combined. Fig. 5.5 shows one such instance. Sometimes, both are required, such as with *eBay* feedback. SOCIAL FEEDBACK can become very fine-grained. For example, if a user leaves a one or two star reply for the accuracy of item description on *eBay*, the site pops up a further question asking what was wrong with the item.

SOCIAL FEEDBACK is the general pattern that also includes reputation systems, where a certain classification is given to a user that indicates their overall 'quality'

---

[16]It's worthy of note that at the time of writing, of the seven user-given definitions of 'facebook poke' at *Urban Dictionary* (`http://www.urbandictionary.com/define.php?term=facebook%20poke`), six make reference to flirtation. As one user puts it: "The Facebook poke is especially useful in the process of overanalyzing a potential romantic interest's feelings about you based solely on impersonal online interactions."

[17]"Feedback loop" is a highly ambiguous term that has a semantic payload defined by multiple disciplines. Here the meaning is only that feedback is given, and feedback is given in response, and feedback is given in response to that, creating a loop.
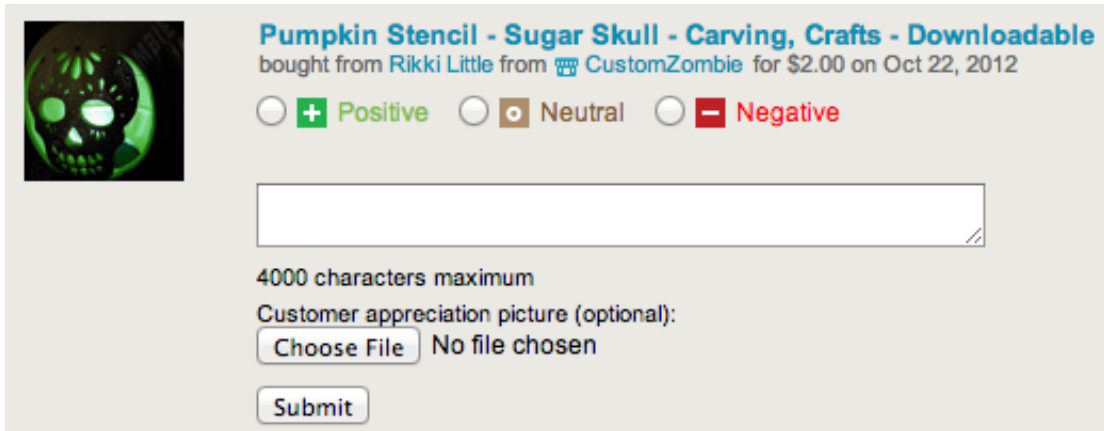
Figure 5.5: A screenshot from *Etsy*, showing the possible feedback mechanisms. Users can give structured feedback in the form of a positive, neutral or negative response, and then include unstructured feedback, including an appreciation picture! Seller details publicly available. Taken on February 10, 2013.

(for want of a better word). Reputation systems can be *Power-* and *Status-*based when they are contingent on user intelligence and quality, such as with the *Stack Overflow* reputation system, which allows for structured feedback to be given to good answers. They can also be *Honor-*based, such as with *eBay*, where users are rated on their honesty and quality of sales interaction.

SOCIAL FEEDBACK systems tap into our sense of fair play and politeness, creating *social obligation*. Social obligation can be used to elicit responses, such as in the case of *Google* asking users to wish a friend a happy birthday, which meets our needs of *Honor* and *Idealism* (this pervades both the Web and on mobile, as seen in Figs. 5.6 to 5.7).

Behavioral economics views social obligation as a form of reciprocal altruism (Axelrod and Hamilton 1981; summarized in Thaler 1994), where people become engaged in a loop that could be broadly described as, "Do unto others as you would have them do unto you."[18] The act of meeting a social obligation in order to fulfill moti-

---

[18]This is known as The Golden Rule by philosophers (Singer 1963).

Figure 5.6: A screenshot from *Google Search*, showing that it is a *Google+* contact's birthday. Photo and birthday information publicly available.

vational needs is sometimes termed 'impure altruism' (Andreoni 1990) (as opposed to the model of pure altruism, where the act of performing such an act has no physical or emotional benefit at all). Experiments have shown reciprocal altruism to be powerful. In the case of *Farmville*, gifting loops have been shown to create a sense of *Acceptance* in players (Wohn et al. 2011).

Social feedback can be used as a form of *Toxboe*: Social Proof. The public nature of social feedback means we'll use it to make decisions: buy from sellers who have high reputations, wish happy birthday to friends when we see many other people do it, leave comments on *eBay* that say 'A+++++++++' even though we'd never write that anywhere else. This likely feeds into motivations of *Acceptance* and *Honor*. Porter (2010) discusses how apps can shape behavior using Social Feedback to provide social proof.

> *Yelp* takes pains to promote certain profiles whose owners behave as model citizens. They tend to have huge numbers of friends, lots of reviews, and other gaudy numbers that represent success on the site. It's clear that the designers at *Yelp* want to promote desired behavior in the hopes others would see and emulate it.
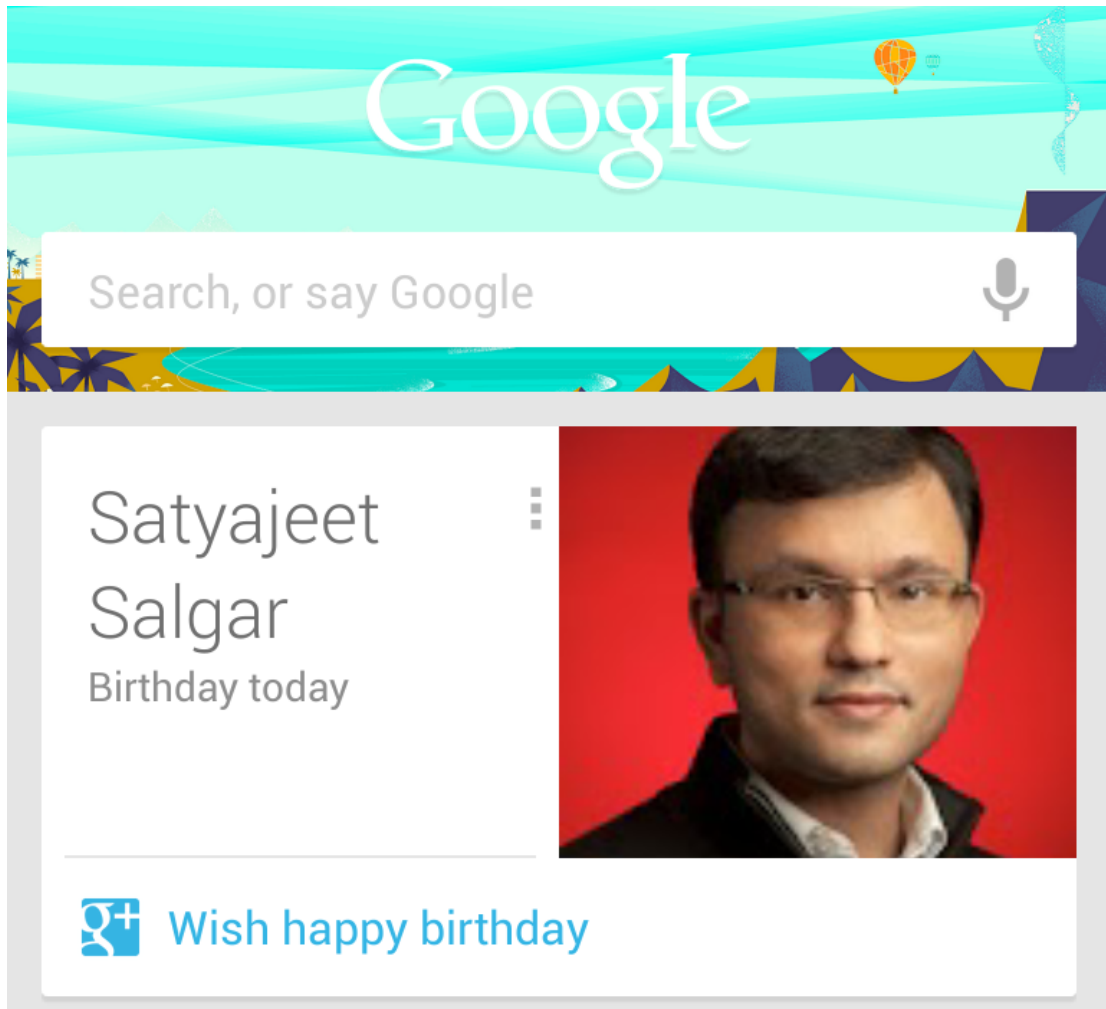
122

Figure 5.7: A screenshot from *Google Now* on *Android*, showing that it is a *Google+* contact's birthday. Photo and birthday information publicly available.

Watch for: When a user is giving structured feedback, some sites, such as *eBay*, require the user to provide unstructured feedback as well. This can be grating, especially to those who do not have as strong a need to fulfill their *Honor* motivations. The intent appears to be for the unstructured feedback to fill in the gaps that structured feedback cannot, but extra signal is certainly not guaranteed. What can result is seen in Fig. 5.8 where a user was involved in multiple transactions, and chose to cut and paste his response instead. Worse, this feedback gives the appearance of being rote in a way that structured feedback does not. If the behavior reaches a critical mass, social proof indicates that this will permeate throughout interactions on the site.

When deciding between structured and unstructured feedback, it is worth considering the amount of friction that the inclusion of unstructured feedback offers. When interactions with just structured feedback can be dramatically increased by removing page loads to an AJAX asynchronous load[19], forcing unstructured feedback may not result in any useful response.

Social obligation is a powerful tool, and enables Dark Pattern: Social Pyramid Scheme.

5.5 Pattern: Activity Stream

Description: A series of Broadcasts or Notifications that illustrate recent events

Reiss desires: *Curiosity, Social Contact*

Also known as: *Crumlish & Malone*: Activity Streams, News Feed, *Tidwell*: News Stream

Related to: Broadcast, *Tidwell*: Dashboard, Identity Shaping, Intriguing Branches, *Crumlish & Malone*: Personal Dashboard, Social Feedback

Examples: *about.me, Facebook, Flickr*

---

[19]According to Kevin Rose in regards to the design for the now defunct *Digg* (as reported by Wroblewski 2008; via Porter 2010).

● Multiple transactions with this 5* favorite. Thanks for the positive experience.		Buyer: whjepson (260 ☆)
CHIP AND DALE SODA FOUNTAIN JERK 3D Limited Edition Disney Pin ON CARD!!! (#200877597225)		US $11.69
● Multiple transactions with this 5* favorite. Thanks for the positive experience.		Buyer: whjepson (260 ☆)
TINKER BELL MICKEY PHILHARMAGICAL DINING Disney LE Pin (#380541076462)		US $10.99
● Multiple transactions with this 5* favorite. Thanks for the positive experience.		Buyer: whjepson (260 ☆)
CINDERELLA AND PRINCE CHARMING SWIRL 3D Disney LE Pin (#200828708564)		US $7.19
● Multiple transactions with this 5* favorite. Thanks for the positive experience.		Buyer: whjepson (260 ☆)
LIGHTNING MCQUEEN & THE KING CARS LOGO FAMILY Disney Pin (#130840912131)		US $3.49
● Multiple transactions with this 5* favorite. Thanks for the positive experience.		Buyer: whjepson (260 ☆)
MINNIE ST. PATRICKS DAY 2010 Disney LE 3D Pin (#130840915461)		US $3.75

Figure 5.8: A screenshot from *eBay*, showing feedback given from the same user buying Disney pins from a seller. In order to avoid typing unstructured feedback for each review, the buyer simply cut and pasted the same response for each pin bought. Reviews publicly available. Taken on February 10, 2013.

Use: Activity Streams consolidate Broadcasts or Notifications into a single area, allowing users to quickly see all the activity that has taken place within the application. *Facebook*'s news feed is the most popular example of an Activity Stream, and other examples include the *Twitter* home page, *Flickr*'s Friends PhotoStream, and email inboxes. They help users find new content, and often offer a chance of providing Social Feedback and discovering Intriguing Branches, while Filtering signal from noise. Activity Streams can either aggregate the activities of a single user (such as a *Facebook* profile page) or many users (such as the *Facebook* news feed).

Activity Streams consolidate information in one place, so users don't have to hunt for updates, increasing the *Tranquility* of knowing nothing important has passed them by. They also serve as a means of answering the question "Who is here?" providing a proof of liveness, making users feel confident that their *Social Contact* needs can be met. Applications that rely on social networking in particular benefit from Activity Streams in this manner, turning the perception of a ghost town into a bustling, active

Figure 5.9: A screenshot from *Twitter*, showing an ACTIVITY STREAM of tweets from various users. Some of them are discussing the resignation of John Riccitiello from Electronic Arts. Tweets publicly available. Taken on March 18, 2013.
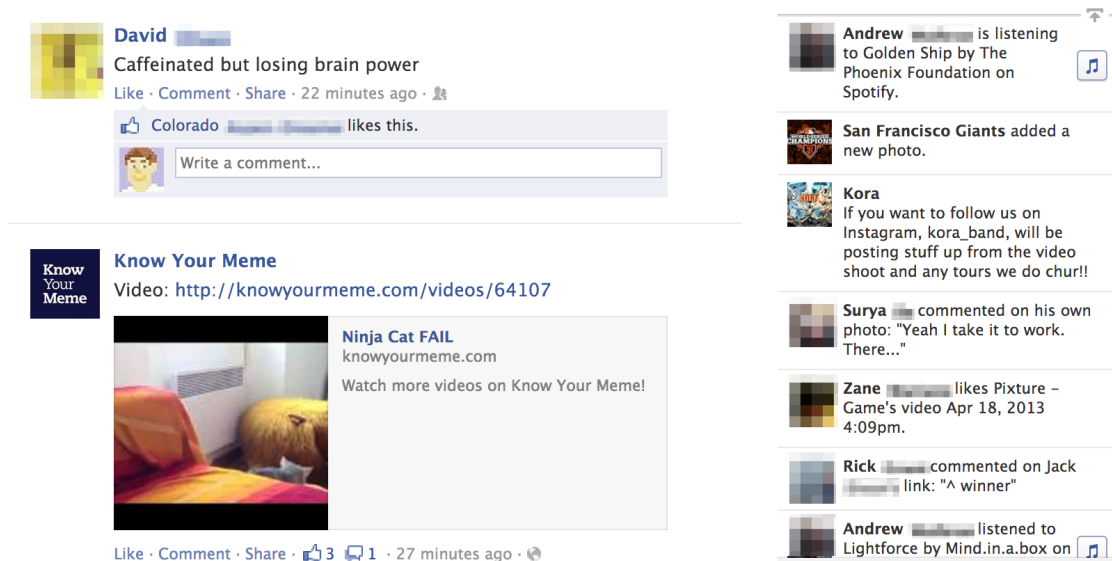
Figure 5.10: A screenshot from *Facebook*, showing two different ACTIVITY STREAMS. On the left is the main news feed, where most user interaction happens. The news feed is semi-curated to attempt to show important events. On the right is the "ticker", which shows a raw feed of activity from contacts. Avatars and last names obscured for privacy. Taken on April 18, 2013.

meeting place. Users can casually check in on either their global CONTACT LIST, or drill down into individual contacts to see how someone in particular is doing, satisfying their *Curiosity* about what might be going on.

ACTIVITY STREAMS are one method that users can engage in IDENTITY SHAPING, and services like *about.me* take advantage of this. *about.me* is a service that lets users attach ACTIVITY STREAMS from other applications, consolidating the streams of *Facebook*, *LinkedIn*, *Twitter* and more. *about.me* thus creates a separate ACTIVITY STREAM that merges these all together, showing how ACTIVITY STREAMS can be many layers deep.

5.6 Pattern: Identity Shaping

Description: A means for users to customize their identity, and provide controls for how their identity is viewed by different groups

Reiss desires: *Acceptance*, *Independence*, *Romance*, *Tranquility*

Also known as: *Crumlish & Malone*: IDENTITY, *Crumlish & Malone*: PROFILE

Examples: *Google+*, *Gravatar*, *Twitter*

Use: IDENTITY SHAPING allows users to customize their online identity, and control who sees what about them. This allows users to shape their identity as they see fit, and present different identities to different people. This is a computational manifestation of the sociology theory of dramaturgy (Goffman 1959). The sociological 'dramaturgy' borrows from the theatrical concept of 'dramaturgy' which pertains to how drama unfolds on the stage. In the sociological use, the theory of dramaturgy posits that people in real-life are essentially actors on a stage, and that we present ourselves differently to different audiences, maintaining different 'masks' (personas). This is easy to understand: the person who we present to our boss is often different from the person we present to our drinking buddies. According to dramaturgy, we do this to gain *Acceptance* from
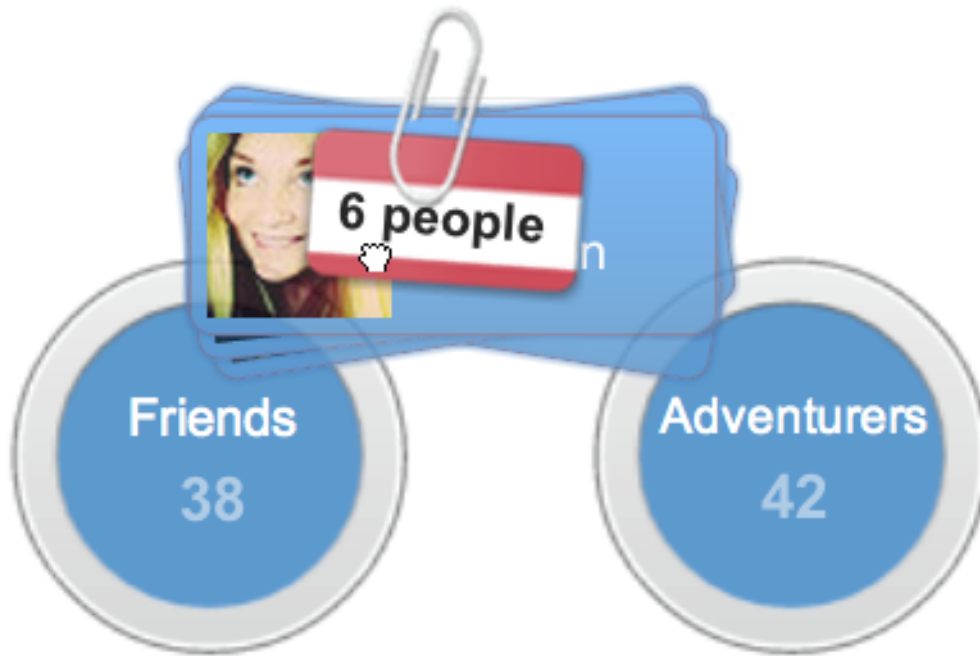
Figure 5.11: A promotional image from *Google+*, showing the circles interface, which allows users to place people in different circles. These circles are then assigned BROADCASTS, allowing users to choose who sees what content.

our audience, and IDENTITY SHAPING is a means of allowing this in software.

A good example of IDENTITY SHAPING are the controls present in *Google+*. *Google+* uses a concept of "circles," a term they apply to CONTACT LISTS that represent different social circles. A user can place any of their contacts into any circle, and a contact can be in more than one circle at a time. For example, one could maintain circles named "Best Friends," "Family," "Academics" and "Following." These can not only be used to separate out what BROADCASTS the user wishes to show—such as only showing photos of boozy nights out to friends— but also those she wishes to hide—such as limiting in-depth discussions on dry academic papers to her Academics circle. This allows users to align their masks with how they would operate in real life. In comparison, *Facebook*'s user interface used to make it very difficult to change

the visibility of BROADCASTS, which was changed after the launch of *Google+*. Even seemingly innocuous data, such as what a user "likes," can be used to accurately piece together a user's sexuality, race, religion and political leanings (Kosinski, Stillwell, and Graepel 2013). This placed users in a social quandary: should they limit their *Facebook* CONTACT LISTS to only those they fully trust (possibly creating a situation when friend requests have to be turned down, which can create friction), or should they accept all who come, and hide possibly controversial details from what they share?

Effective identity shaping controls not only helps us gain *Acceptance*, but a feeling of *Independence* that we have the power to shape ourselves, and also provides *Tranquility* from the fear that the mask we present to a certain audience may be invalidated, leading to negative consequences. When privacy controls are not available, users can meet these needs by setting up multiple profiles, or by using a certain mask for a certain application. For example, *LinkedIn* offers a venue for a professional persona that *Facebook* doesn't, and so we feel more comfortable adding professional colleagues on *LinkedIn* (this can be undermined if the information is public and the users' real names can be cross-referenced, as mentioned in the "Watch for" section below). The need for controls arises when an application wishes to present a "catch-all" single identity that should assume all masks, as *Google+* is designed for: *Google Play* reviews, *YouTube* comments, *Blogger* and *Gmail* are all tightly bound to a single *Google+* account.

IDENTITY SHAPING doesn't just apply to BROADCASTS themselves, but also to how those are collected on profile pages or even small CUSTOMIZATIONS such as avatar pictures, usernames and signatures.

Watch for: One common source of friction between users and application developers is whether users should be forced to use their real name, giving rise to the so-called *Nymwars* (Galperin 2011). The Nymwars were a battle after the launch of *Google+*, where Google had instituted a policy where users were expected to use their real name, not a pseudonym. Google enforced the policy, and suspended accounts which did not

130

appear to have their real name attached. This created an uproar from privacy advocates, going so far as to creating an extensive list of at-risk demographics—based on sexuality, employment, health and others—who could be threatened or discriminated against by such a policy ("Who is harmed by a "Real Names" policy?"). Users were not able to shape their identity as they wished, leading to a perceived loss of both *Independence* and *Tranquility* if they were to register with *Google+*.

boyd (2012) notes how real names disempower people to manage social situations online:

> When people are expected to lead with their names, their power to control a social situation is undermined. Power shifts. The observer, armed with a search engine and identifiable information, has greater control over the social situation than the person presenting information about themselves.

Madrigal (2011) describes this as a "radical departure from the way identity and speech interact in the real world":

> Imagine you're walking down the street and you say out loud, "Down with the government!" For all non-megastars, the vast majority of people within earshot will have no idea who you are. They won't have access to your employment history or your social network or any of the other things that a Google search allows one to find. The only information they really have about you is your physical characteristics and mode of dress, which are data-rich but which cannot be directly or easily connected to your actual identity. In my case, bystanders would know that a 5'9", 165 pound probably Caucasian male with half a beard said, "Down with the government!" Neither my speech or the context in which it occurred is preserved. And as soon as I leave the immediate vicinity, no one can definitively prove that I said, "Down with the government!"...In real life, we expect very few statements to be public, persistent, and attached to your real identity.

Thus, IDENTITY SHAPING, *including pseudonyms*, actually better represents our real-world ability to control our identity.[20] Real names policies should be instituted with care and attention. Sometimes the intention of using real names is to counteract

---

[20]With the rise of wearable computing like Google Glass, it is unclear how long interactions without cross-referencing will last.

the *online disinhibition effect*, which can lead to trolling and uncivil discourse (Suler 2004).[21] Dash (2011) suggests that instead of using real names, persistent identities can be used instead. People are held accountable, but are allowed to choose which persona they wish to use and cultivate on the service, and can prevent their name being leaked.

5.7 Pattern: Item Sharing

Description: A way for users to trade, share or copy items between one another

Reiss desires: *Honor, Idealism, Saving, Social Contact*

Related to: COLLECTION

Examples: BitTorrent, *World of Warcraft*

Use: ITEM SHARING is a mechanism that allows users to fill in parts of their COLLECTION, but also can also create reciprocal gifting bonds between individuals, or between collections of individuals, such as guilds. ITEM SHARING is a much more prevalent pattern in games than in standard applications, although it doesn't engender a sense of gamefulness.

World of Warcraft provides a strong framework for analyzing ITEM SHARING through its trading systems. Trades can happen in the Auction House, where players list items they would like to trade, and others can bid on them using in-game money. Trades can also happen between players using the trading window.[22] Players may gift items to other players, increasing *Social Contact*. When players are in guilds, they have access to a guild bank, which is a shared area where players in the guild can put items in, and take items out (pictured in Fig. 5.12). Players put their items in the bank to help their guildmates, again meeting their *Idealism* or *Honor* needs.

---

[21]Many gamers, myself included, find this effect more simply stated by a Penny Arcade comic, which characterizes this behavior with "John Gabriel's Greater Internet Fuckwad Theory" that stipulates: Normal Person + Anonymity + Audience = Total Fuckwad (Holkins and Krahulik 2004).

[22]The trading window makes both parties confirm the exact details of the trade before it is approved in order to prevent items being stolen.

Figure 5.12: A promotional screenshot from *World of Warcraft* depicting the guild bank interface.

One-to-one trading like this doesn't seem to occur much in general applications, as the supply of virtual goods is not constrained in the same way as virtual worlds. If it can be copied, it can be shared freely to everyone. The most common version of this sort of sharing is file-sharing via protocols such as BitTorrent.[23] In some BitTorrent communities, a sharing ratio is used. *UKNova*, for example, allows users with high ratios of sharing vs downloading to access files earlier, whereas users with very low ratios have a long wait, and risk being banned from the community.[24] These file sharing communities often have users who go to the trouble of recording television shows or ripping DVDs, and they do so out of their needs for *Idealism* and *Honor*.

---

[23]It is worth noting that at the protocol level, BitTorrent enforces trading. Any file that is being downloaded has pieces uploaded to other users at the same time. Clients that throttle this trade will in turn be throttled back, and their download speed will decrease.

[24]While not pertinent to the discussion, downloading content from *UKNova* is almost always an instance of piracy.

Chapter 6

Interface patterns

Interface patterns are patterns that address how the interface communicates to the user, and how the user can affect that communication. Interfaces PRAISE us so that we know we're doing the right thing, they offer up PREDICABLE RESULTS so we know what's going to happen when we click on a button, and they let us UNDO things when we decide we don't like the results.

Interface patterns are the reason why we find ourselves so drawn to our smartphones, idly flicking around icons and tapping at the screen. Every time we poke at a little green pig on the screen and it giggles, or we tap on a shower head at random to see if it would do something—and then finding out it does—the interface PRAISES us. "You found the secret! Your tapping is important! You can't provide an incorrect input!" As facile as this seems, it's satisfying.

Look to including interface patterns whenever spending idle time in an application is important, or when users are being asked to perform complicated procedures. Communicating to the user that things are moving along smoothly will help keep them *Tranquil* and engaged.

6.1 Overview

6.2 Pattern: Praise

Description: Approval for performing actions

Reiss desires: *Acceptance*, *Curiosity*, *Tranquility*

Also known as: *Toxboe*: Praise

Related to: Score, Undo

Examples: *FarmVille*, *Smashing Magazine*, *Where's My Water?*

Use: Praise describes feedback systems that communicate to users that their behavior is correct and even desirable. They help users feel at peace with the interface, a sense of *Tranquility* that nothing they are doing can harm them, and a sense of *Acceptance* from the approval that they are offered.

Praise is used by games often and liberally. Schell (2008) describes this as "Praise," from which *Toxboe*: Praise takes the name.

> Praise [is] the simplest of rewards, the game tells you that you did good work, either through an explicit statement, a special sound effect, or even an in-game character speaking to you. It all amounts to the same thing: the game has judged you, and it approves. Nintendo games are famous for giving players lots of secondary praise via sounds and animations for every reward they get.

This pattern only relates to Praise as the form of feedback and reward. Schell (2008) explicitly separates Score from Praise.

Today, one might look to social games for offering Praise frequently, as shown in Fig. 6.1. This large amount of Praise for clicking resulted in Jacobs (2012) referring to Zynga games as "the gamification of clicking," as she describes in her analysis of *The Pioneer Trail*:

> It appears that although some aspects of game structure can be found in the social games of Zynga, at the same time, it's difficult to definitely argue

Figure 6.1: A screenshot from *CastleVille* showing PRAISE. Here, the act of tending a flower, which required a single left-click of the mouse, results in three separate rewards: experience points (the star), flower seeds, and honey. The important thing is not what those rewards confer to gameplay (these are separate concerns, and are more related to SCORE), but the response given: you have clicked, and the click was awesome, so here is a big animation, and three pretty icons replete with sound effects on their appearance and collection. This is what Schell calls "secondary praise," the resources themselves being the primary reward.

that it is indeed a game. But the question is then: what is it gamifying? How is it gamification? I argue that "social games" like *The Pioneer Trail* are the "gamification of clicking"...Rather than needing to click to explore the environment, the environment is built to accommodate clicking. The graphical overlay and rudimentary storyline work together to create not so much a game, but rather a clever, yet simple, example of the gamification of clicking.

"Juicy feedback," coined by Hunicke (2009) helps provide a better framework for understanding how PRAISE can be communicated through user interfaces, instead of through explicit statements and rewards. Hunicke describes it as:

Tactile The user can almost feel the feedback coming from the screen.

Inviting The user wants to interact with the application for the positive feedback that's on offer.

Continuous The feedback is offered all the time.

Repeatable If the same goals are met, the same feedback is offered again.

Emergent Feedback flows naturally from the application.

Balanced The user isn't overwhelmed by the feedback given.

Fresh Feedback has a little surprise or twist, and is welcomed when combined with the continuous feedback.

Juicy feedback is especially present in mobile games. Figs. 6.2 to 6.3 shows two examples of juicy feedback before players even enter the games. Approval is constantly being offered. When such feedback isn't available, disappointment sets in, and even a nagging worry that perhaps touches aren't being registered by the app or device. Wigdor et al. (2009) calls this the "touch feedback ambiguity problem," and notes that "this ambiguity can lead to a disconnection from the system, and frustration, or a loss
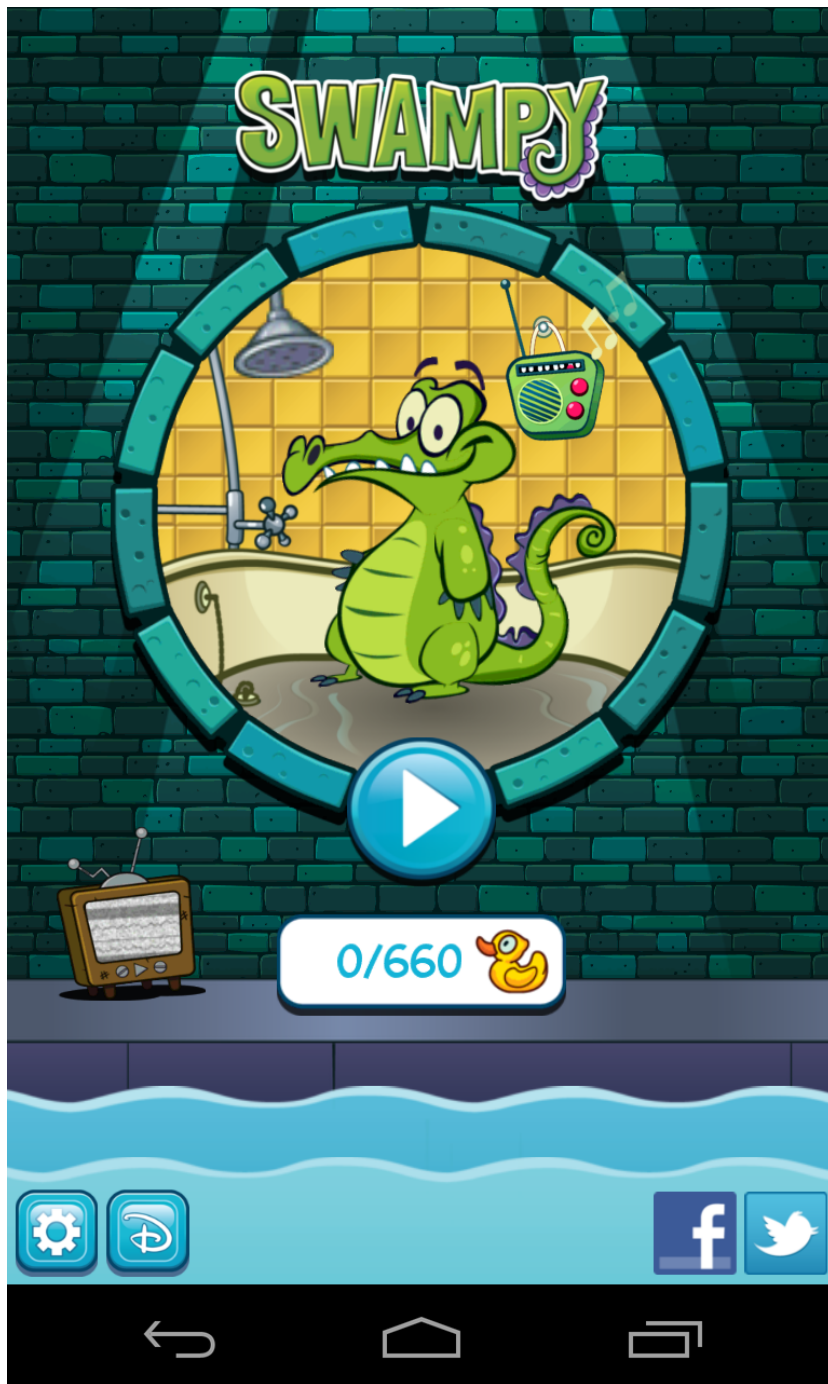
Figure 6.2: A screenshot from *Where's My Water?* on *Android* showing the start screen. Here, there are three touchable elements that provide feedback that aren't part of the UI navigation. Touching the alligator (Swampy) results in an animation; touching the radio results in a little quack noise and the radio bouncing around; and touching the shower head makes water drip out.

Figure 6.3: A screenshot from *Angry Birds Star Wars* on *Android* showing the level select screen. Besides all of the touchable level numbers, the asteroids, TIE Fighter pigs, and pig with the black helmet on, are all touchable and pleasingly explode when pressed.

of sense of control." Losing a sense of autonomy and mastery from touch applications is highly detrimental to the experience, and juice feedback provides one means of dealing with the issue, increasing users motivational needs.

Other uses of PRAISE are not so obvious. The design magazine, *Smashing Magazine*, describes three channels to provide PRAISE through UI: visual (static), animation and audio (Boag 2012). One of the best, and yet most nuanced, pieces of PRAISE on show is a description of when users click on a link on *Smashing Magazine*. When a link is clicked, it highlights in a large red box immediately, providing instant feedback that the user successfully interacted with the link. "It surprises me how many websites fail to show the user they have successfully clicked on a link," writes Boag. "Relying on the browser to provide positive feedback can be problematic as the user may miss it. This is because the browser shows that it is loading a page using the address bar, while the user's attention is on the link that they have just clicked."

When users feel safe and accepted by the app, they're far more likely to take risks and explore the application more fully, allowing them to satiate their *Curiosity*. This is a core part of motivational design, but is a quality that is difficult to express.

Watch for: PRAISE, particularly in small doses such as small visual cues, cannot be relied upon as a motivator in and of itself. Instead, PRAISE helps users feel at ease with the application, which should then begin to meet other motivational needs. When an application focuses around PRAISE, such as many social games, the rewards have the ability to become overused. This is hinted at by juicy feedback's requirement for "freshness": once a reward isn't surprising, it doesn't feel rewarding. As noted in the discussion of CET, this is explained by prospect theory, which states that when we get used to gains, we'll set that as a reference point. Thus a constant state of gains creates a normalcy at that reference point, and so gains are only valuable if they are greater than that reference point. The gains have to increase in order to provide value.

When the reward is offered for clicking alone, as Jacobs (2012) theorizes, then

the rewards must be increased in order to retain users. This puts your application in a rewards arms race that you likely won't win. Instead, focus on "a little goes a long way," and use PRAISE as a secondary dynamic that helps users feel the *Tranquility* they need to reach other motivational needs. Don't scare users off by forgetting to PRAISE them, but don't make PRAISE the only reason they're there.

PRAISE is often provided in verbal or written forms, and it's important to avoid controlling language such as "you should," "you must" or "you ought to." When using phrases like "Excellent work!" it's important to back up such a statement with specific reasoning, such as "Excellent work, you managed to catch all this fish within the time limit" (Reeve 2005, p. 150).

6.3 Pattern: Predictable Results

Description: Actions taken in the application should have predictable outcomes

Reiss desires: *Power*, *Tranquility*

Related to: Praise

Examples: *Google Search*, *iTunes*

Use: PREDICTABLE RESULTS allow users to understand how software works, and be able to correctly guess the outcome of actions they take. This pattern is part and parcel of "intuitive design", which is well described by James (2009):

> The general thought in user interface design is to make the interface (webpage, software, physical device, whatever) be intuitive to the user. In other words, when the user performs an action in the interface, the product should do what the user expects the device to do. For example, if you press a right-facing arrow button on a music player, you expect this should start some music playing.

Or, more succinctly, "Don't make me think" (Krug 2009). When expectations are violated, the user has to sit, think and wonder whether she is actually doing the right thing. Meeting user expectations raises feelings of *Tranquility* and *Power*, violating them

reduces them. Users should be able to perceive the affordances an interface provides, and correctly predict the general outcome of a particular set of inputs. This often requires being consistent throughout the app, and consistent with UI conventions on the platform. It requires providing enough information to the user for them to make correct decisions.

For example, *Twitter* is generally poor at helping the user, by forcing shortened URLs. Fig. 6.4 shows an example of this, and clearly violates the "Don't make me think" principle, with a confusing set of links that don't conform to the Web platform expectation of descriptive targets. Where does the first link go? What about the second link (it has 'pic' in the string, so it'll probably show an image, but will it be the same one or different)? How are the shoes being shown, when tweets are all text? What happens if the shoes are clicked? The answers are not easily predictable. The first link takes you to Target where the shoes can be purchased. The second link shows a lightbox of the shoe image. The picture of the shoes, when clicked, doesn't take you to Target (as could be reasonably expected), but instead shows the lightbox as well. Fig. 6.5 shows an example of how this tweet might look instead. The unnecessary link to view the image in a lightbox is removed (the full image is already visible, anyway), and the text on the link to Target is replaced with the actual title of the page. A small "expand" icon is added to the image to clarify what will happen when the user clicks on it. It is worth nothing that features such of these are not uncommon: both *Facebook* and *Google+* resolve the titles for attached links and images, and *Google+* will rewrite a raw URL to its title in the text if it's from a *YouTube* video.[1]

Intuitive design is covered in more detail as part of usability design, and it will not be investigated here, as it is much better served by other texts. Krug (2009) is especially good for Web applications.

---

[1]This only seems to happen with *YouTube* links. Links to other sites, even ones that Google owns, such as *Google Play*, do not get rewritten.

Figure 6.4: A screenshot of a tweet from Target on *Twitter*. The first link takes you to Target, where you can buy the shoes. The second link takes you to the picture of the shoes (again). Clicking the shoes also opens the picture of the shoes.

Figure 6.5: A render of a tweet from Target on *Twitter*, with the links modified. In this version of the tweet, the unnecessary link to view the shoes in a lightbox is removed. The shortened URL to Target is expanded, and now is much clearer at where it will take the user.

PREDICTABLE RESULTS is equally applicable to games, but with less emphasis on the general outcome, and more on the immediate result of the input. For example, playing *Far Cry 3* is a wonderful experience because outcomes are generally unknown. Players can rely on knowing that the right trigger will shoot their gun, the left will aim, the stick will move their avatar around and so forth. What they don't know is that running into that particular spot in the river will end up with them being ravaged by a crocodile,[2] or that shooting that explosive barrel will result not just in the barrel exploding, but the whole base being set on fire. These *emergent* outcomes are wonderful in games, but much less so in general applications.[3]

6.4   Pattern: State Preservation

Description: Applications can be exited at any time, with the user safe in the knowledge that the application has saved their state

Reiss desires: *Acceptance*, *Independence*, *Power*, *Tranquility*

Also known as: Autosave, Checkpoints

Related to: PREDICTABLE RESULTS, UNDO

Examples: *Google Docs*, *Halo*, *iOS*

Use: STATE PRESERVATION constantly saves the state of the application, allowing users to drop out of it, safe in the knowledge that it will resume where the user left off. For an app to implement STATE PRESERVATION, it must not require input from the user to express the wish to save.[4] STATE PRESERVATION allows users to feel *Tranquility*, safe in the knowledge that if the program crashes or is accidentally exited, all work will be saved. It allows users to feel *Independence*, able to switch around applications (or even

---

[2]The game lets players swim around the water without incident for the most part, and so when a crocodile does happen to be there it's a shocking experience.

[3]Sometimes it feels like *Microsoft Word* is an application with emergent outcomes. Who knows how the formatting will change/break when the next letter is entered?

[4]Although it can implement that alongside, such as in *Google Docs*, which auto-saves the document state constantly, but also allows the user to press a save button if she wishes.

leave the computer completely), and not feel tied to the current application. Finally, a state of *Acceptance* and *Power* is created, as users no longer have to build mental models of operating system process management.

STATE PRESERVATION has been constant throughout computing history, in the context of non-document-oriented apps. While *Microsoft Word* required explicit saving, email applications always remembered where you had sorted mail, and *iPhoto* doesn't require the user to save how they organized their albums. Whether the application is Web-based or on the desktop, STATE PRESERVATION is part of many applications, even though we don't really notice it.

That being said, STATE PRESERVATION is much more prevalent on mobile platforms, where the limited amount of RAM originally prevented any sort of multi-tasking at all, as was the case with the first iPhone. This pattern takes its name from *State Preservation and Restoration* (2013), which describes it as:

> At some point, the system might need to terminate your app to free up memory for the current foreground app. However, the user should never have to care if an app is already running or was terminated. From the user's perspective, quitting an app should just seem like a temporary interruption. When the user returns to an app, that app should always return the user to the last point of use, so that the user can continue with whatever task was in progress. This behavior provides a better experience for the user.

By saving the state of the application when the user left and then having the system quit the application (mobile applications have no specific quit button), a pseudo-multitasking was created. While phones now have more RAM at their disposal, software developers appear to have decided that users don't want to manage processes as they once did. Mobile platforms continue to background and/or quit applications at the discretion of the system (*State Preservation and Restoration* 2013; *Recreating an Activity*), and *Windows 8* introduces this system too, as it can act as a mobile platform (Barras 2012).

However, the prevalence of this pattern even extends past memory-limited devices. *Mac OS X 10.7* (codenamed Lion) introduced support for automatic termination, with the justification that the system is better equipped to manage system processes than users ("The Core App Design" 2012):

> Automatic termination transfers the job of managing processes from the user to the system, which is better equipped to handle the job. Users do not need to manage processes manually anyway. All they really need is to run apps and have those apps available when they need them.

To support this, apps must implement STATE PRESERVATION, saving both the active document and the current interface state.

While we now often encounter STATE PRESERVATION in general apps, STATE PRESERVATION was used for much longer in video games that utilized checkpointing and autosaves. The canonical example for this is *Halo* on the Xbox, which was released six years prior to the release of the iPhone and *iOS*. As players progressed through the game, it would autosave the player's current position, resetting her to the checkpoint should she die. Unlike respawning, which simply resets the world to a hardcoded initial state, the state of the game at the time of checkpointing was restored, including what weapons and ammunition the player had, where she was in the game space and whether she was riding in a vehicle or not. STATE PRESERVATION is now a common feature of console games.[5]

Watch for: STATE PRESERVATION is a relatively new concept at a process level, and while it removes the cognitive burden of process management from users, it introduces a new layer of abstraction that can prove problematic. One example is that STATE PRESERVATION must always be coupled with UNDO. When users have no control of what is saved and when, they must have a clear path back to previous states. Before STATE PRESERVATION, users could be sure that not saving, and then closing the document or quitting the program, would return them to the state when they last expressly saved.

---

[5]PC games still tend to rely on user-initiated quick saving.
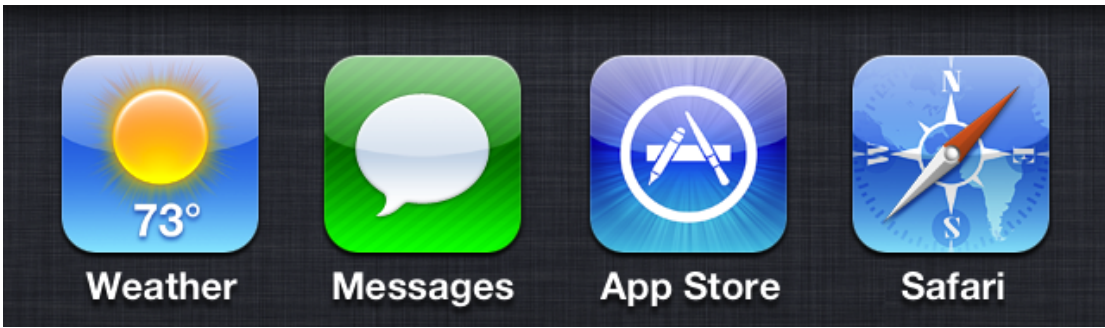
Figure 6.6: A screenshot of the recently used apps bar in *iOS*. Taken from `http://support.apple.com/kb/ht4211`.
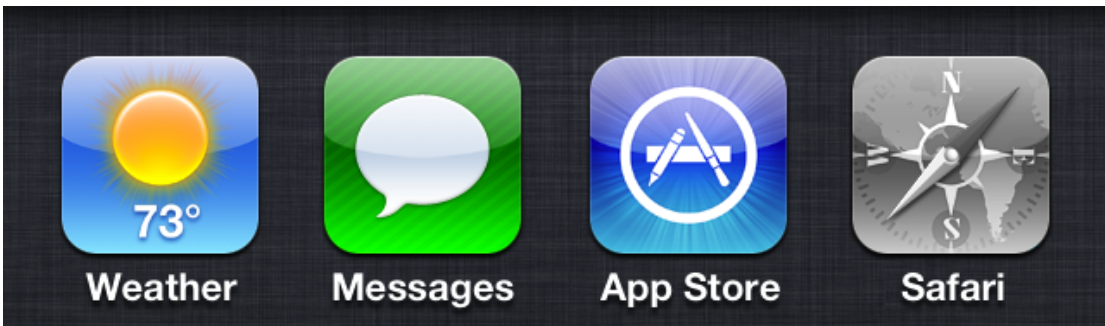
Figure 6.7: A mockup of the recently used apps bar in *iOS*, where *Safari* has been terminated, resulting in a greying out of the icon.

This is no longer the case. An example of alleviating this problem with UNDO can be found in GOOGLE DOCS, which creates revisions (sometimes many revisions in a minute), allowing users to rollback to any point in time when they felt their document was in a good state.

A second issue with user mental models and STATE PRESERVATION can be found when users build an incorrect mental model of what is happening. For example, on *iOS*, double-tapping the home button brings up a bar of recently used apps. Some of these may be backgrounded, and some may have already been terminated. Using the *iOS* convention of holding one of the icons brings up a small quit logo, which allows users to remove icons from the task bar, which will also terminate apps that were still

148

running.[6] One user was witnessed systematically opening the list, and closing all the applications. She would do this multiple times a day, and it was quite an arduous process, as *iOS* can keep a long list of the recent applications. When it was inquired as to why she did this, she replied that she had been told by a friend that doing this would conserve battery, because all of these applications were running. Her mental model, and that of her friend, was reasonable given the information at hand in the interface, and matches that of operating systems like *Windows 7*, where all apps in the taskbar are actively running. However, her mental model reduced her *Tranquility* and *Autonomy*, almost to a state of paranoia, where she believed her phone would quickly run out of battery if she didn't return at regular intervals. One could imagine an easy fix in this case without resorting to a manual: apps could have some sort of signifier that indicates they are not running, such as being greyed out. Fig. 6.7 shows what this might look like. Another option is to not surface recently used apps without it expressly being turned on in the options, where some small help text could inform the user what the bar actually means.

The automatic termination in *Mac OS X 10.7* and *10.8* is poorly implemented. Applications can be terminated at any time, as long as they match some fuzzy heuristic of being "unused." Apps that have no windows open or are minimized meet this description, but an app can even have open windows and be terminated. Neuberg (2011) illustrates an example where this is clearly not making the user experience better:

> The fact is that when *Lion* caused *Preview* to quit automatically yesterday on my machine, I was using *Preview*. I wasn't using it actively at that moment in a way that Lion knew about—there were no open *Preview* windows, and *Preview* wasn't frontmost—but I was engaged in some activity involving *Preview*. I had switched away from *Preview* only in order to prepare things in the *Finder* so that the document I intended to open in *Preview* would be ready. But when I switched back to *Preview* with `Command-Tab`, *Preview* was gone. That's not helpful or useful; it's annoying, confusing, and a hindrance. I had to launch *Preview* explicitly again in order to continue

---

[6]This can be used as a Force Quit when an app was misbehaving.

with my task.

Again, this is part of a user's mental model on *Mac OS X*: applications should not quit unless they are told to do so. Violation of this model reduces the user's *Tranquility*, who is now left unsure as to whether the system will begin closing windows on a whim. As Neuberg had no windows open, the system could have terminated *Preview*, but left the *Preview* icon in the dock and app-switcher giving the appearance of it running. When he switched to *Preview*, the system could have resumed the application. This solution is even suggested by the author in a follow-up article on *Mac OS X 10.8*, which maintains the behavior of *10.7* ("Mountain Lion is (Still) a Quitter"). Ironically, the solution is exactly the same way that *iOS* acts, which is where this idea of automatic termination has come from. There is no distinction between running and not running when the user chooses an application to switch to.

The final example of a broken use of STATE PRESERVATION comes from the *Halo* series. The problem is that *Halo*'s checkpointing *is only retained while the game is running*. If the player pauses the game, then presses "Save and Quit", the transient checkpoint is saved, and the game can be safely quit. If "Save and Quit" isn't used, and the player leaves the game by some other means, the save is not kept and the state is not restored; players have to return to their latest save, or restart the game from built-in checkpoints.[7] This is unlike other games that implement checkpointing, where the user can safely leave the game and come back right where she left off. The rumor is that this is a limitation in the game engine, but the release of *Halo 4* had not fixed this issue, which caused much grinding of teeth on gaming forum *NeoGAF*, in a thread titled "I hate you, archaic and useless Halo save system" (2012):

> Why not just do that automatically with the checkpoints? I can't see someone saying, "you know, I really don't want to save this entire afternoon"

---

[7]At least in the case of *Halo 3*, starting from a checkpoint meant that the game did not register you as having finished the game; it counted as skipping through.

so why even make us do that extra step when you already have fucking checkpoints.

They want to keep all your good Halo memories intact!

it's not even archaic. Games from 15 years ago have better save systems. I really really hate when a game has checkpoints, but no way to hard save (I personally just don't trust "Save & Quit" systems) Drives me up the wall.

Happened with every damn Halo title I played. It's ridiculous. Lost like 3 missions in *ODST* on legendary.

Halo 4's campaign wasn't very compelling to begin with. But when I read "Checkpoint Reached" at the top of the screen I once went straight back to the dashboard. In gaming parlance, that means "saved." Next time I sat down to play I realized I'd lost two of the most dull hours in the campaign. Took me a week to muster the interest in playing again. It felt like having to rewrite a term paper after a disk failure in the 1990s.

6.5   Pattern: Undo

Description: Allow for actions to be reverted, rolling the system back to a prior state

Reiss desires: *Curiosity*, *Independence*, *Tranquility*

Also known as: *Tidwell*: Multi-Level Undo

Related to: *Tidwell*: Safe Exploration, State Preservation

Examples: *Microsoft Word*, *Photoshop*, *Google Docs*

Use: Undo is a pattern that we are all familiar with: when we make a mistake, we press

the undo button and the mistake is erased (hopefully). While the implementation of

undo is actually far more difficult than one might think, requiring deep understanding of

aspects such as what constitutes a unit of action, or what actions are actually reversible,

I'll leave deeper discussion of implementations to other literature (for example, see

Tidwell 2010; Folmer, Welie, and Bosch 2006).[8] Undo supports *Tranquility*, providing a

---

[8]Group undo, it seems, has produced much grist for the academic mill over the years, with no clear "winning" formula. One particularly interesting approach was taken by Abowd and Dix (1992), who attacked the problem using formal methods.

safety net for experimentation, which allows for *Curiosity*, as users can backtrack from anything potentially harmful. Tidwell (2010) describes it as:

> The ability to undo a long sequence of operations lets users feel that the interface is safe to explore. While they learn the interface, they can experiment with it, confident that they aren't making irrevocable changes—even if they accidentally do something "bad." This is true for users of all levels of skill, not just beginners.

Tidwell (2010) describes this as *Tidwell*: Multi-Level Undo, which better highlights that undo operations should have a stack, allowing users to perform multiple undo operations, and points out research that indicates users don't tend to move past twelve operations (Constantine and Lockwood 2002).

It is worth noting that the experimentation and fulfillment of *Curiosity* that is afforded by Undo in general applications is generally provided by State Preservation in video games.

When preserved states are expressly made available to the user, the relationship between Undo and State Preservation is close, but they are not one and the same. For example, *Google Docs* offers both. Undo rolls back one granular action at a time, and the undo stack is usually lost when a user exits the application. State Preservation unrolls multiple actions, and persists throughout multiple application launches.

6.6   Pattern: Notifications

Description: Interface elements that alert the user to some application state change

Reiss desires: *Curiosity, Order, Tranquility*

Related to: Activity Stream, Broadcast, Dark Pattern: Interaction by Demand

Examples: *Android, Facebook, iOS*

Use: NOTIFICATIONS are small messages that appear to indicate some state change has occurred in an application. A new email has arrived. A friend wants to add us to her CONTACT LIST. An invite to a party. NOTIFICATIONS can either occur both inside the application (such as the "orangered" envelope icon on *Reddit* that indicates a new message has arrived) or outside of the application (such as the notification systems used on smartphones, or more traditional methods like email or even text messages). NOTIFICATIONS are similar to BROADCASTS, except that NOTIFICATIONS are part of the relationship between the receiver and the system, whereas BROADCASTS are the relationship between the sender and the system. For example, a user can BROADCAST a status message to *Facebook*. Receivers may subscribe to NOTIFICATIONS about updated statuses, or they may not. As the act of BROADCASTING does not necessarily imply any NOTIFICATIONS—one could build a BROADCASTING system without a NOTIFICATION system—these patterns are separate.

As with BROADCASTS, NOTIFICATIONS can be included in an ACTIVITY STREAM to provide a single place to find them.[9]

NOTIFICATIONS warn us that a potential INTRIGUING BRANCH is available, and provide a quick means with which to satisfy our *Curiosity*. They're a classic example of a variable ratio schedule; we never quite know whether a notification is going to point to something immediately important, or something completely superfluous (this is described in more detail in 7.1).

The absence of NOTIFICATIONS also provides us with a sense of *Tranquility*, safe in the knowledge that nothing that may require attention has happened. How many of us are attuned to reflexively checking our smartphone screens "just to see" and, depending on the current situation, be either glad or disappointed to see that

---

[9]At the time of writing, ACTIVITY STREAMS are a common means of organizing NOTIFICATIONS, but this was not always so. For example, before the creation of the Notification Center in *iOS 5*, NOTIFICATIONS took the form of single modal pop-ups. *Windows Phone 7 and 8* doesn't include an ACTIVITY STREAM either, and uses transient 'toast' pop-ups when a NOTIFICATION arrives, and then adds a badge to the app 'tile' indicating something has changed.
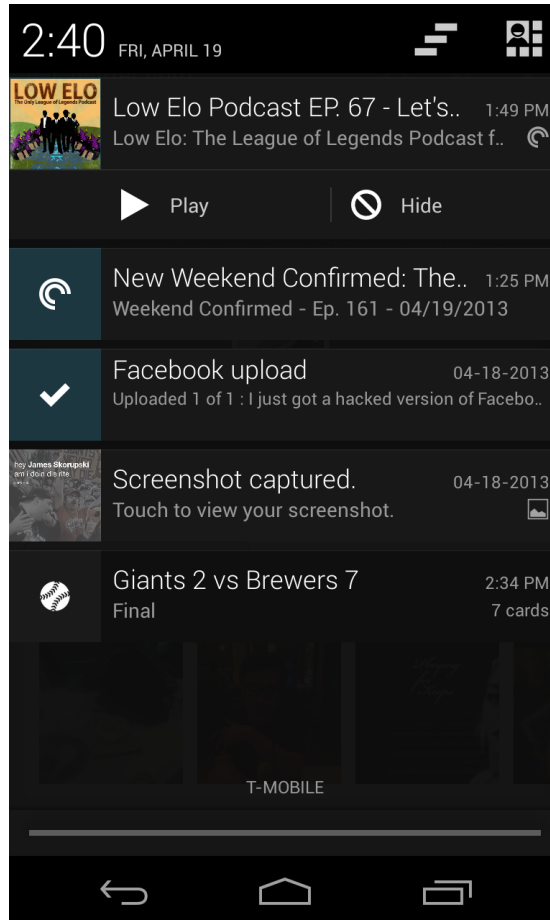
Figure 6.8: A screenshot of an *Android* Notifications Drawer, which constitutes an Ac-TIVITY STREAM of NOTIFICATIONS. Here we can see just how many different ways that NOTIFICATIONS are (ab)used. The first notification is from a podcast player, allowing the user to play and pause it from the drawer, which can be useful for quickly changing the state while in another app. The next notification is from the same podcast application, informing the user that a new podcast is available. The next two relate to a screenshot that was taken and then subsequently uploaded to *Facebook*. The final notification is the baseball score. What's interesting about these NOTIFICATIONS is that none of them really require any immediate *attention*, as we tend to think of NOTIFICATIONS. Here, they just offer up some sort of information which could be digested at any time.

there are none?

One aspect of NOTIFICATIONS that may be unappreciated is that *Orderly* people will feel the desire to clean them up often. The small badges indicating that *something* has happened indicates a state of disorder, and so they either need attending to or deleting. A means of quickly clearing all notifications, or disabling them altogether, would help these users feel a better sense of order.

Watch for: This pattern could, in and of itself, be considered a dark pattern. NOTIFICATIONS intrude in our lives in a way that we are uneasy with. We find ourselves drawn to NOTIFICATIONS, and smartphones have placed them within our reach at all times. Instead of finding the willpower to turn these off, we turn to more and more convoluted solutions. Google Glass, a set of glasses that deliver notifications to a small screen in the top-right of the wearers vision, is one such attempt to address the issue. But is this truly the answer? As Topolsky (2013) puts it:

> Does it seem weird to you, that to get people having more human interactions that we've reached a point... that we have to augment ourselves with Glass? Have we done something wrong? Have we screwed up something fundamental? You're like "I just want to have a human experience, let me put on these robot sunglasses."

NOTIFICATIONS hold a particularly strong power over us, likely due to their variable ratio schedule construction. As such, it seems unlikely that we will find a design that is less addictive without machine help to decide when and where a notification is actually important.[10]

The power of NOTIFICATIONS is easily abused, either by sending junk notifications that are unrelated to actions the user has taken in the application, or by sending them often. Doing so constitutes the use of DARK PATTERN: INTERACTION BY DEMAND.

---

[10]Of course, we need to trust the machine to actually get this right, otherwise we'll just default back to checking them all ourselves again, *just in case.*

Chapter 7

Information patterns

7.1 Overview

Information patterns allow us to interact with content, SEARCHING to find INTRIGUING BRANCHES, ORGANIZING our own information, and FILTERING that which we don't want. All the patterns in this chapter relate to how we consume and organize content.

These patterns are primarily about satisfying our *Curiosity* about the world, and keeping a sense of *Order* of the constant flood of information that bombards us.

Use these patterns when your application requires users to be exposed to large amounts of information. Without a good means of preventing users feeling like they are "drinking from the firehose," they'll lose their sense of control of the subject, and even forget what they intended to do in the first place.

7.2 Pattern: Customization

Description: Users can customize their interface, and objects within the virtual space, in order to improve their space and feel greater ownership over it

Reiss desires: *Independence*, *Order*, *Saving*, *Tranquility*

Figure 7.1: A screenshot of a customized car in *Forza Horizon. Used under Creative Commons Attribution-NonCommercial 2.0 Generic license. Taken by CommodoreLXIV* `http: // www. flickr. com/ photos/ 82931937@ N04/ 8223549656/`

Also known as: *Zichermann & Cunningham*: Customization

Related to: Growth, Identity Shaping

Examples: Forza Horizon, Microsoft Office, Twitter

Use: Customization refers to users being able to modify their computing environment, be it the application, or objects within the application itself. When Customization is used as a form of expression, rather than as a means to increase effectiveness, it also a use of the Identity Shaping pattern.

When we think of Customization, it's easy to remember a number of games that allow their in-game objects to be customized, such as *Forza Horizon*'s cars, which can be upgraded or painted with new decals, as in Fig. 7.1. We might even think of customizing our particular player characters in Role-Playing games, tweaking their statistics to our choosing.[1] Performing such customization helps us feel Independence,

---

[1]This isn't an example of Growth because none of the three requiremeents are present: there are

as are offered the freedom and autonomy to do what we wish, and our Saving needs are also being met, as we begin to build up a collection of customized items or characters. Our bonds to these things begins to grow, and soon the endowment effect kicks in. The endowment effect says that we place a higher value things we own than those we don't. Kahneman, Knetsch, and Thaler (1990) gave a group of people mugs. People who had been given the mug, and had established ownership of it, were willing to accept a sale of their mug at twice as high as how much they would be willing to pay for another one. The working theory is that loss-aversion, as described in prospect theory, is to blame: we simply can't face the loss of the thing we now own. This effect has also been found in areas such as basketball tickets. Once the tickets are in hand, people take ownership of the dream of seeing the match, and value their tickets more highly than they did before (Ariely 2010, p. 167–173). The same thing happens to our customized items. We feel ownership of them. We value them more highly than we did before. The things we are Saving are now more important than they once were.

However, customization extends past just virtual goods, but to the interface itself. We customize our Web browsers with extensions, rearrange icons in *Microsoft Office* and modify our wallpaper on *Twitter*. *Zichermann & Cunningham*: Customization describes the act of modification as making a commitment to the product, and this also represents the beginning of a *sunk cost*: the time expended to customize can't be taken to a new product, increasing the costs of switching. Users are able to feel more at home in the application, asserting *Independence* and a creation of a *Tranquil* environment that has *Order*. *Zichermann & Cunningham*: Customization does note the oft-cited paper "The Tyranny of Choice," which describes how adding too many customization options overwhelms users, and points towards the success of Apple's low-customization hardware as a win over Dell's monstrously large options list.

---

no delayed effects, nothing happens when the player is not present, and there is no non-deterministic aspect.

7.2.1 Specialization of Customization: Filters

Description: Tools that allow users to filter content, highlighting what they enjoy and hiding what they do not like

Reiss desires: *Idealism*, *Order*, *Tranquility*

Also known as: *Crumlish & Malone*: FILTERING

Related to: ACTIVITY STREAM, CUSTOMIZATION, ORDERING OF INFORMATION, REPORTING, SOCIAL FEEDBACK

Examples: *Gmail*, *Reddit*, *Slashdot*

Use: FILTERS are tools that allow the adding of or removing content from view. This can either be proactive, such as selecting subreddits, or reactive, such as deleting an email. It can also be only applicable to the current user, or a global filter. Filtering is a specialization of CUSTOMIZATION that appears often enough to warrant its own pattern. When the input given by the user is visible to the original content creator, FILTERING is a form of SOCIAL FEEDBACK.

FILTERS allow users to choose content they enjoy and avoid upsetting material, enhancing their feelings of *Order* and *Tranquility*. Those who actively moderate content in order to improve the community as a whole do so out of a wish to boost their sense of *Idealism*.

The archetypal use of FILTERS can be found at *Reddit*. *Reddit* is an enormous site. According to `http://www.reddit.com/about`, 56 million people visit the site each month, there are over 3500 active communities (subreddits), and `http://www.stattit.com` counts over 60,000 links submitted daily. The sheer size and scope of *Reddit* would be overwhelming, but *Reddit* utilizes FILTERS aggressively to ensure users only see content they want to:

- *Reddit* is split into *subreddits*, which users can subscribe or unsubscribe from. Each subreddit forms an IDENTIFIABLE COMMUNITY. Reddit by default subscribes

users to 20 different subreddits based on the number of unique visitors. These include 'pics', 'funny', 'science', and 'atheism.'[2] As Erik Martin, general manager of *Reddit* puts it, "*Reddit...* has become a platform and tool for a community, as well as a community itself" (Aidan 2013). This allows users to proactively tailor their *Reddit* experience.

- SOCIAL FEEDBACK is used heavily, and that feedback is fed into the FILTERING process. Stories can be upvoted or downvoted, increasing or decreasing their prominence. Highly upvoted stories reach the front page, where most people who browse, rather than actively engage in content selection, will see it. Comments are similarly voted up or down, and the most popular comments appear at the top of the list.

While FILTERING is most often largely guided by human signals, there is no requirement for humans to be the majority decider. Spam filters, for example, often use heuristics to decide if an email is spammy or not, and will remove such emails from view. This is also a form of FILTERING. When we do not have good spam filters, we know it. We feel like our inbox is disorganized and unwelcoming, reducing our sense of *Order* and *Tranquility*.

Watch for: FILTERING is a great way to target content to users, but can result in "filter bubbles" (Pariser 2011). Filter bubbles reflect a user's ability to view things outside their personalized profile. Content is either filtered by a specific user request, or by an algorithm that attempts to tailor content to specific users. This leads to a possible loss of *Curiosity* (Parramore 2010):

> We thought that the Internet was going to connect us all together... What it's looking like increasingly is that the Web is connecting us back to ourselves. There's a looping going on where if you have an interest, you're

---

[2]The inclusion of 'atheism' appears to be an odd choice for a default subreddit. On one hand, it helps shape the liberal, technolibertarian bent that *Reddit* tends to have, on the other, one would think it scares potential users away who don't realize there are very active subreddits dedicated to various faiths.

going to learn a lot about that interest. But you're not going to learn about the very next thing over. And you certainly won't learn about the opposite view. If you have a political position, you're not going to learn about the other one.

When users aren't aware their own BROADCASTS are being filtered, this is an instantiation of DARK PATTERN: HELLBROADCASTS.

7.3   Pattern: Personalization

Description: The system modifies itself to its perceived needs of the user

Reiss desires: *Acceptance*, *Status*, *Tranquility*

Related to: CUSTOMIZATION, GROWTH, IDENTITY SHAPING

Examples: AMAZON, GOOGLE ADSENSE

Use: PERSONALIZATION occurs when the system adapts to the user, *anticipating* things that it thinks she may want. Doman (2012) offers a more specific definition:

> Personalization technology enables the dynamic insertion, customization or suggestion of content in any format that is relevant to the individual user, based on the user's implicit behavior and preferences, and explicitly given details.

PERSONALIZATION increases our feelings of *Acceptance*, offers us *Status* by showing us that we are important enough to warrant our own *personal* version of something, and creates *Tranquility* by automatically FILTERING things we don't like.

The most common examples of PERSONALIZATION are found in shopping and advertising products. *Amazon*'s front page is full of items it thinks the user will be interested in, shown in Fig. 7.2. GOOGLE ADSENSE uses a user's browsing history to deduce your demographic, and then deliver advertisements based on that. PERSONAL-IZATION is now so common that "personal is the new normal—it is the baseline user expectation" (Goldberg 2013).

Figure 7.2: A screenshot of *Amazon* recommendations showing an instantiation of the PERSONALIZATION pattern. Note how each bar of products is the *same function*, but labelled differently. They all show items related to products that have been previously viewed (but were not purchased). The PERSONALIZATION function then tries to show other products that the user may want to purchase.

A number of implicit and explicit signals can be fed into a PERSONALIZATION algorithm. Implicit signals occur just through normal use, such as the number of times a page is viewed and for how long, how many interface elements the user clicks on the page, or whether the user shares the page with others. Explicit signals are ones where the user communicates to the system that the PERSONALIZATION is right or wrong, through the use of ratings, thumbs up or down, stars, or removal of items.

Fung (2008) investigated three ways a website could adapt to its users:

Remembering The system remembers frequently accessed pages. An example would be a portal site that shows news based on its derived interests for you.

Comprehension The system understands what action you're trying to perform and steps in to help, such as *Microsoft Office's* Clippy, who attempts to help users write letters when they start a document with "Dear."

Association The system clusters users together and infers customizations based on the cluster, such as *Amazon*'s personalized recommendations system.

Fung found that only comprehension and association increased user retention, while remembering had no effect. These types don't easily bind to Reiss motivations, and to do so would create tenuous linkages at best. Perhaps comprehension and association increase retention by simply being better *tools*, rather than meeting any intrinsic need. Association may help to provoke feelings of *Curiosity*, guiding users to interesting content (perhaps via an INTRIGUING BRANCH).

Watch for: PERSONALIZATION is very similar to CUSTOMIZATION, but the key difference is that while CUSTOMIZATION is driven by the user, PERSONALIZATION is driven by the system. While users don't notice this distinction,[3] it is an important one not

---

[3]Studies presented by Goldberg (2013) shows that "Users don't necessarily distinguish between personalization and customization. Users appreciate the transparency and that controls are available, even when they don't use them."

just academically, but from a motivational standpoint, as there are cases when both are suitable candidates.

For example, consider a user interface for a word processor that re-arranges itself based on functions that the user uses the most. This act of PERSONALIZATION makes sense, for as long as the user understands what is happening and is made aware that elements are being moved, and where they are moved to. Otherwise, the interface loses consistency, decreasing *Tranquility* and *Order*. However, offering users full control over their user interface with CUSTOMIZATION may result in too much overhead; many people just want the system to be "sensible" rather than require manual tweaking.

Choosing the right pattern is difficult. CUSTOMIZATION makes sense when the user is able to fully understand the consequences of the action, has the ability to make good decisions, and can easily find a means to UNDO their actions if they get it wrong. PERSONALIZATION is a better choice when the user is not necessarily expected to accurately predict or create the right outcome, but will "know it when they see it," with some means to feedback to the system that something isn't right.

Sometimes a two-pronged approach, utilizing both patterns, is the answer. In the word-processor example, an implementation may be to offer PERSONALIZATION, with the system explicitly calling out when it's moving an element, using a small animation showing it moving from a nested menu to a button on the user interface, with a small message saying "You change your paragraph formatting a lot, so now it's available on the toolbar." This could be coupled with both an UNDO button, "Do this less," "Do this more," and/or a "Disable" button, so users can provide feedback to the PERSONALIZATION algorithm as to how it is doing. Disabling the functionality moves it to a full CUSTOMIZATION implementation instead.

7.4 Pattern: Reporting

Description: Content that users deem unacceptable can be reported

Reiss desires: *Idealism, Tranquility*

Related to: ACTIVITY STREAM, FILTERING, *Crumlish & Malone*: REPORT ABUSE, SOCIAL FEEDBACK

Examples: *Facebook, Google+, Reddit*

Use: REPORTING is used for applications where user-generated BROADCASTS are hosted, which potentially allows offensive content to appear. One method to deal with offensive content is for moderators to pre-screen all submissions and FILTER those that are unacceptable, but this does not scale well.[4] The second, and far more popular method, is for users to REPORT offensive content after it has been published. UI buttons for this purpose are generally labelled with text like "Report abuse" or "Flag."

REPORTING is *generally* offered in conjunction with FILTERING, so that when a report is made, the content is immediately FILTERED. A clear example of this is the "Report Spam" functions of email programs, that immediately move spam messages that managed to reach the inbox to a folder the user doesn't have to see them. This provides *Tranquility* quickly, and is generally the most preferable means of implementing the REPORTING pattern (with the ability to UNDO the FILTER or the REPORT if the user changes her mind). This is not always the case, for example *Stack Overflow, Khan Academy* and *Hacker News* all keep the BROADCAST visible even after it has been reported.

As with FILTERS, REPORTING also provides users with a feeling of IDEALISM, making them feel like they are actively contributing to the community in a virtual clean-up initiative. Whenever BROADCASTS are offered, use of REPORTING should be

---

[4]It was reported that this was the method Nintendo would take with its *Miiverse* application on the Nintendo Wii U in order to ensure children would never see offensive content (Pereira 2012). However, there does not appear to be any confirmation in the press that this strategy was actually employed when the console was released.

strongly considered.

7.5 Pattern: Search

Description: A means for users to search for content

Reiss desires: *Curiosity*, *Independence*, *Tranquility*

Related to: Organization of Information

Examples: *Facebook* Graph Search, *Google Search*

Use: Search describes systems that allow users to search for content. We're all familiar with it via search engines like *Google* and *Yahoo*. Search is an important part of any application that offers content. Primarily, it forms a way of satisfying our desire for knowledge, represented by *Curiosity*, and a means for us to find knowledge that won't necessarily be handed to us, represented by *Independence*.

However, the presence of Search doesn't just offer serendipitous knowledge hunts, but also a means of retrieval. We are constantly adding information to the digital domain, be it *Evernote* notes, photos to *Flickr* or documents into *Microsoft Word*. Placing information into a digital form is only useful if it is retrievable again, and the sheer amount of information that we now create can outpace our ability to Organize that information. Results from research about how we retrieve data is not good. For example, Whittaker, Bergman, and Clough (2010) found that users didn't want to Organize their photo libraries because they found the task onerous, even though the users had communicated how important they felt their photos were. However, when it came to looking for specific photos, they failed to find them around 40% of the time. If a photo is irretrievable, regardless of whether it is saved or not, it essentially ceases to exist to the user. Even worse, the authors note that "As [users'] digital picture collection continues to grow in size, their ability to retrieve pictures of a certain family event might be expected to decrease: both because the users' memory for its location degrades and

Figure 7.3: A photo taken of the "Andre has a posse", with an image of the wrestler, and actor, Andre the Giant.

because each new folder they add distracts them from their target." The deep anxiety surrounding whether saved information is truly available to the user creates a loss of *Tranquility.* SEARCH is offered as a technical means to alleviate this issue.

Two particularly good search experiences can be found in *Google Drive* and *Facebook Graph Search.*

Google Drive is an application that stores documents created by *Google Docs*,

Figure 7.4: A photo taken of a San Francisco Giants baseball cap.

but can also synchronize files from a computer hard drive. One of the benefits provided is that SEARCHES can make use of Google's massive computing power and machine learning techniques. Figs. 7.3 to 7.4 are two results from a search for "giants." One shows an image of a sketch of Andre the Giant, and one shows a picture of a San Francisco Giants baseball cap. Neither was ever given any meta information. Google's computer vision algorithms were able to spot that there were recognizable images within the pictures. When searching vacation pictures for landmarks, the user can simply type the landmark and *Google Drive* will retrieve it.[5] This is hugely liberating, and offers *Independence* and *Tranquility* to users who do not wish to spend the time organizing their photo albums.

In 2013, *Facebook* unveiled Graph Search, a natural language means of searching for content on *Facebook*, which relies on the meta-information that its users add. With it, complex queries can be created. For example, one can search for "Photos of me and John Doe in Hawaii," and all the photos geo-located in Hawaii, with the user and John tagged in the photo, will appear.[6] Similarly, the user can use "Music Jane Doe listens to" to find bands she likes, or "Bars Jane Doe has been to" to find watering holes. Pleasingly, synonyms are supported, so "comrades who dig my shorty," "randos who live in new york" and "trends my besties are into" are all valid queries (Lynley 2013). Here, SEARCH not only fulfills the user's *Curiosity*, but is also able to support *Facebook's* wider goal of *Social Contact*. A well-implemented SEARCH is not only motivational in its own right, but can enhance the around it motivational patterns that serve other needs.

Watch for: SEARCH is often coupled with FILTERS, creating filter bubbles. Users should be offered a means of breaking out of bubbles if they choose. For instance,

---

[5]You can also go the other way, and give a picture of a landmark to *Google Images* and have it search for other pictures.

[6]The act of tagging itself count as ORGANIZATION OF INFORMATION, the important thing here is that the SEARCH cross-references the tags of both our faces, and of the location.

Google Search offers a button to opt-out of its personalized search results on a search-by-search basis (Levy 2012).

7.6 Pattern: Organization of Information

Description: Information can be organized for easy retrieval later

Reiss desires: *Order*, *Tranquility*

Related to: CUSTOMIZATION, SEARCH

Examples: *Evernote*, *Flickr*, *Outlook*

Use: ORGANIZATION OF INFORMATION complements SEARCH. Whereas SEARCH is a means of a computer retrieving information, the ORGANIZATION OF INFORMATION is designed to allow us to find the information ourselves, using systems such as tagging and folders. There has been research into the effectiveness of these systems, such as Bergman et al. (2010), which found that users successfully find their files in over nine out of ten cases, and take about fifteen seconds to do it. What is interesting from a motivational standpoint is not that the organizational systems appear to work, but why we organize in the first place.

Organizing our information helps us feel a sense of *Order*. Things are in their right place, and we feel comfortable in the environment (much like CUSTOMIZATION). When our information isn't ordered, it feels chaotic and messy, decreasing not only our sense of *Order*, but often our *tranquility* as well, as anxiety builds when we can't find something. As with all Reiss motivations, some people have higher or lower tolerances than others. Fig. 7.5 sets many on edge, but the author didn't seem too perturbed. Somewhere else on *Flickr*, a much cleaner desktop was clearly frustrating its owner, "My messy desktop, because files are piling up on it, because I keep telling myself that in just 1 more day I'll totally reclaim this machine" (McLean 2008).

ORGANIZATION OF INFORMATION should generally be coupled with SEARCH.

169
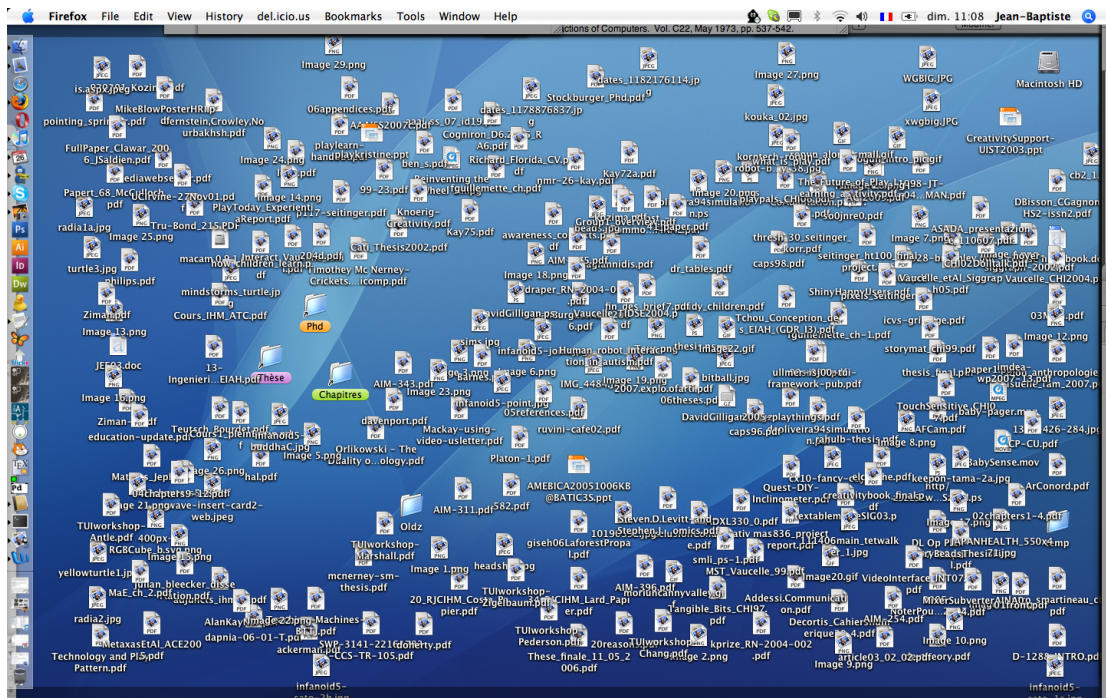
Figure 7.5: A screenshot of a messy desktop full of academic articles, taken "after three days of PhD writing." *Used under Creative Commons Attribution-ShareAlike 2.0 Generic, taken by Jean-Baptiste Labrune* `http://www.flickr.com/photos/jeanbaptisteparis/1238170643`.
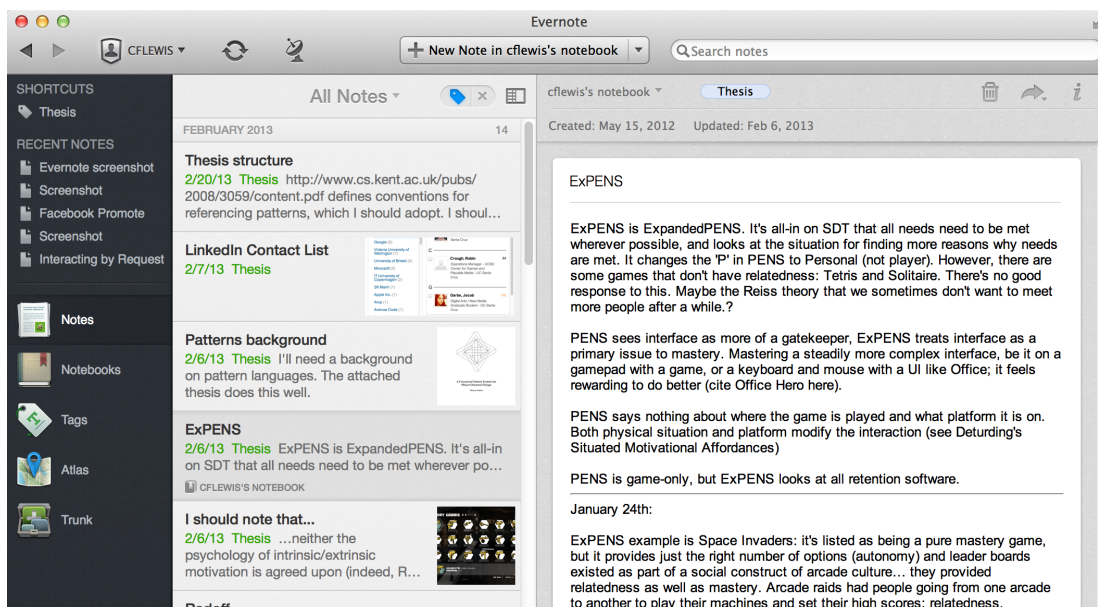
Figure 7.6: A screenshot of an *Evernote* repository. On the left are shortcuts to tags used to organize the notes. In the middle are recent notes, and on the right is the note itself. In the top-right is a search box.

Until SEARCH becomes much more capable at understanding our personal ideas about how we mentally group things, SEARCH is generally focused on pinpointing individual items, such as a photo of Yosemite National Park. However, we might organize all those photos under a broader idea of "Travel," and organizing helps us cast a broader net that is more difficult to express through SEARCH. This allows for browsing and serendipitous navigation, following INTRIGUING BRANCHES through our data. *Evernote* is a good example of an application that allows for both (shown in Fig. 7.6). Tags can be used to organize notes into a taxonomy, so all the notes required for writing this dissertation could be tagged with 'Dissertation.' This gives the user the feeling that even if she doesn't really know what she wants to search for—perhaps because she forgot the content of the note —she can use her organization as a means of locating notes when the time is right.
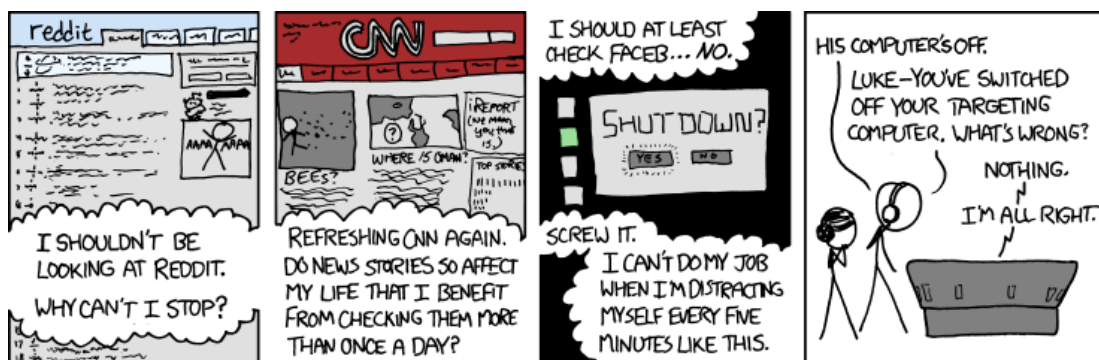
Figure 7.7: Randall Munroe's take on the irresistible draw of INTRIGUING BRANCHES as an XKCD comic, located at `http://xkcd.com/862/`. *Reprinted under Munroe's "occasional" use license clause.*

7.7 Pattern: Intriguing Branches

Description: Placing links to interesting content

Reiss desires: *Curiosity, Independence*

Examples: Email, *Kindle, Reddit, Wikipedia*

Use: INTRIGUING BRANCHES is a pattern first described by *Tidwell*: INTRIGUING BRANCHES, who defines it as "Place links to interesting content in unexpected places, and label them in a way that attracts the curious user." Here, this pattern is limited to just discussing "links labeled to attract the curious user." The content is not really the point; we don't know what the content is until we click the link. The important thing is that the link holds the *promise* that the content *could be* intriguing, piquing our *Curiosity*.

Tidwell actually removes this pattern from her second edition noting that while the pattern is still useful, it is now "blindingly obvious to everyone." While we may now take the humble hyperlink for granted, it's important to note it in this pattern library. In some instances, it may be one of the most motivating patterns of all.

As shown in Fig. 7.7, we often find ourselves drawn to sites like *Reddit* and *CNN*, seemingly unable to escape their powers to fascinate us, when more often than not

172

the branches we follow are unproductive, and sometimes not even entertaining. Even sites that we can rationalize away as educational, such as *Hacker News*, don't really require the amount of attention they often garner.

Intriguing Branches also applies to our email inboxes. Each email seductively calls out to us, and we open it to satisfy our *Curiosity*, but also our anxiety that perhaps something important looms within, satisfying our wish for *Tranquility*.

From a behavioral economics standpoint, Intriguing Branches are a clear instance of a variable ratio schedule, as Ariely (2010) describes:

> Email is very much like gambling. Most of it is junk and the equivalent to pulling the lever of a slot machine and losing, but every so often we receive a message that we really want. Maybe it contains good news about a job, a bit of gossip, a note from someone we haven't heard from in a long time, or some important piece of information. We are so happy to receive the unexpected email that we become addicted to checking, hoping for more such surprises. We just keep pressing that lever, over and over again, until we get out reward.

The majority of our online time is spent in this gambling mindset: checking Notifications, following *Reddit* links, email, instant messaging. Even the humble book, our last bastion of tranquility, has this dice roll included on the Kindle, as words can be instantly looked up in *Wikipedia*.[7]

7.8 Pattern: Task Queue

Description: A list of tasks is placed in a queue, providing an easy means for users to decide what to do next

Reiss desires: *Curiosity*, *Independence*, *Power*, *Tranquility*

Also known as: *Direkova*: Quests, *Zichermann & Cunningham*: Quests

Related to: Badge, *Toxboe*: Tunneling, Dark Pattern: Grinding

---

[7]It's debatable whether these lookups are technically links or triggers for searches. In personal discussions, Jim Whitehead, who has published a number of papers on hypermedia, counted them as computed links. Ted Nelson, who coined the terms hypertext and hypermedia, counted them as searches.

Examples: *Amazon*, *LinkedIn*, *Skyrim*

Use: TASK QUEUES contain a list of tasks for the user to do, so the user is never left wondering what she should do next. This aids feelings of autonomy, and piquing *Curiosity.* Completion of items in a TASK QUEUE provides feelings of achievement, and thus feeds user motivation of *Power.* TASK QUEUES commonly appear as Quest Logs in games, particularly Role-Playing Games like *Skyrim*, as shown in Fig. 7.8. The inclusion of *Zichermann & Cunningham*: QUESTS in *Gamification by Design* seems to have driven the idea that this pattern is somehow tied to games, when of course *Task Queues* have appeared in all sorts of forms before quests. From the productivity management of Getting Things Done all the way down to the lowly shopping list, TASK QUEUES have been a common form of human organization for a very long time.

In general apps, *LinkedIn* has used TASK QUEUES for a number of years to encourage users to complete their profile.[8] Since the original implementation (Fig. 7.9), *LinkedIn*'s use of a TASK QUEUE has become more sophisticated, with multiple layers of task queues being employed, such as moving from the 'Beginner' tasks in Fig. 7.10 to the 'Intermediate' tasks in Fig. 7.11, offering a *Crumlish & Malone*: NAMED LEVELS version of the SCORE pattern.

TASK QUEUES have an implicit goal: users should get to the end of the queue, clearing out tasks as they go. Because users get feedback as to how far they are through their TASK QUEUE, we can apply the goal-gradient hypothesis to their activity. As they reach the end of the queue, they become more engaged, and the pace of completion increases (Kivetz, Urminsky, and Zheng 2006). A way of exploiting this finding would be to complete tasks for users on their list already. For example, *LinkedIn* starts their list at 'Basics,' but the user has already entered details such as their name and email address. Instead of starting the list at 'Basics,' the list could have 'Name' and 'Email

---

[8]Charitably, this is to help users find each other and advertise their skills. Uncharitably, this is to make the data more valuable when sold to recruiters (Anders 2012).
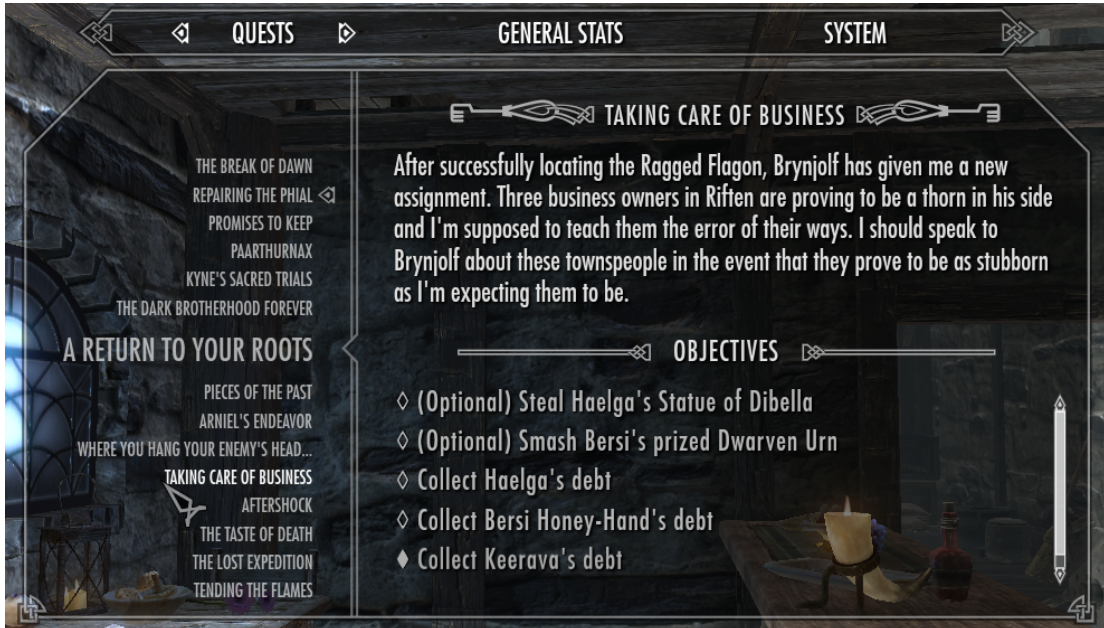
Figure 7.8: A screenshot from *Skyrim* showing its use of Task Queues in the quest log. Like many games, *Skyrim* uses a two-tiered Task Queue. The first tier is the actual quests themselves, listed on the left. On the right is the second tier, which shows the tasks required to complete the current quest. These secondary tasks are hidden from the player until they encounter them, so as not to spoil the surprise of what will happen on the quest. *Taken from* `http: // elderscrolls. wikia. com/ wiki/ File: Quests_ ( Skyrim) _Interface. png`, *used under fair use.*
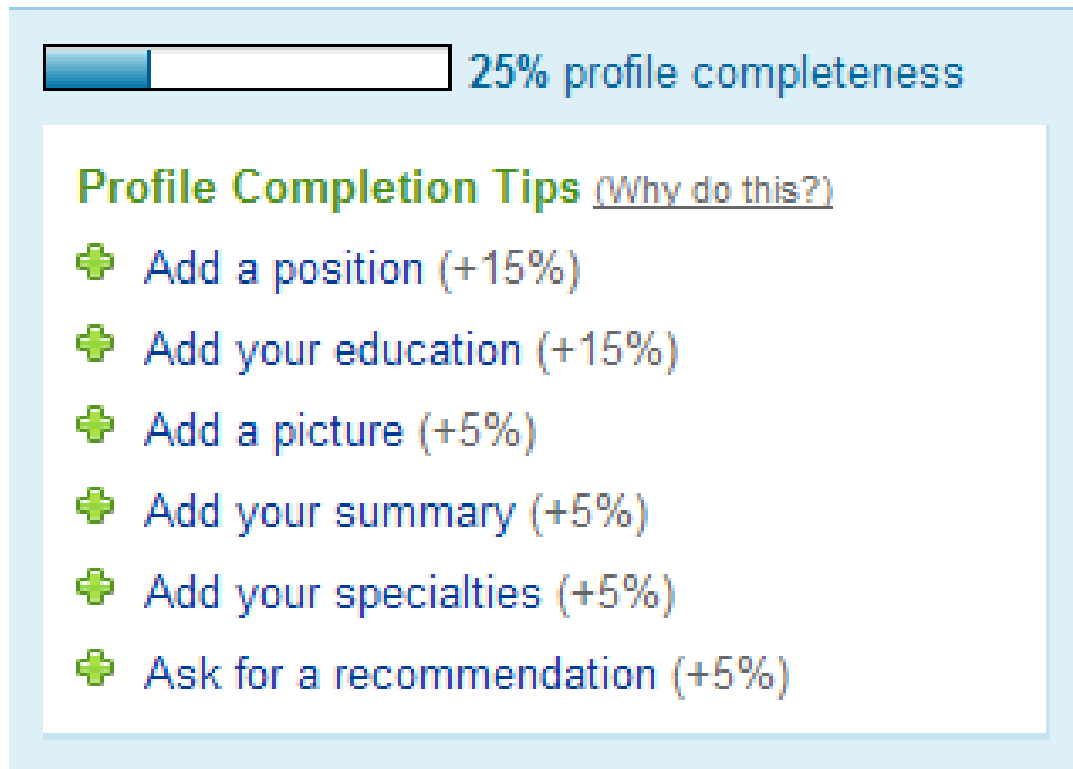
Figure 7.9: An old screenshot from *LinkedIn* showing its use of a Task Queue to encourage users to complete their profile. This design utilizes a progress bar that indicates completeness, and assigns more value to adding positions and education than to adding a picture or specialities.
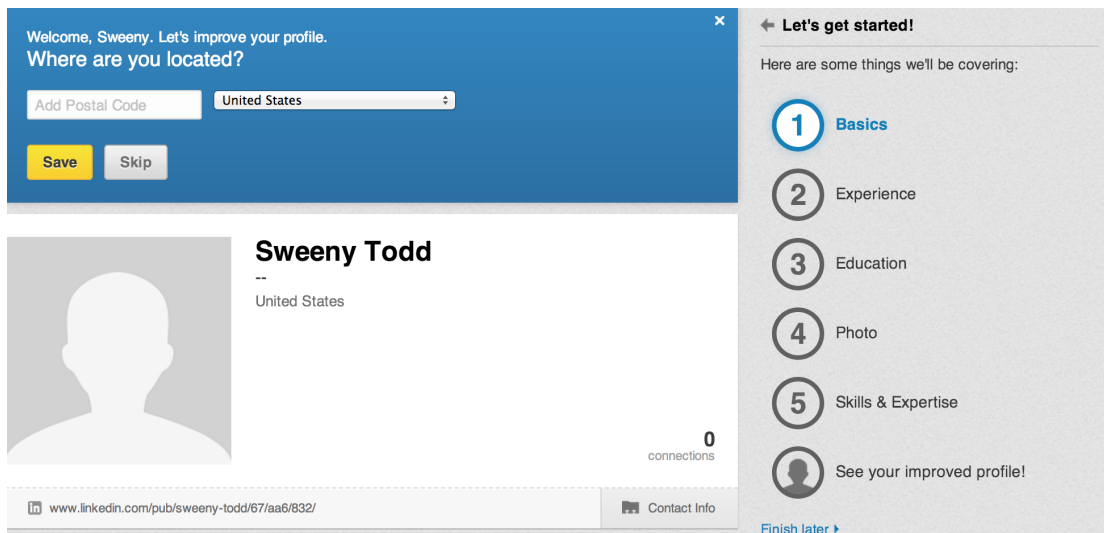
Figure 7.10: A screenshot from *LinkedIn* showing its use of a Task Queue at the time of writing. The explicit progress bar is gone in favor of a numbered system, which forms its own implicit progress where each step is worth 20% towards the goal of an "improved profile." Instead of using a queue where the user is forced to click on a link, edit their profile, save and return to the queue, this queue uses a *Toxboe*: Tunneling method, that allows users to quickly move from step-to-step. As each step is finished, a check mark appears next to the step. Like the old system, users can also head directly to their 'Edit Profile' page and modify their profile that way.
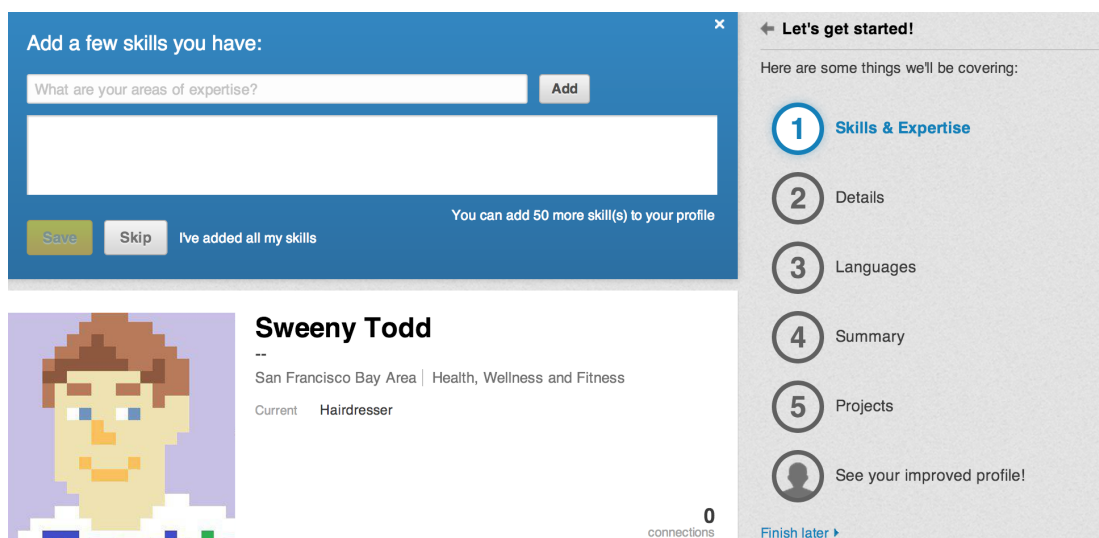
Figure 7.11: A screenshot from *LinkedIn* showing its use of a Task Queue at the time of writing. As the user finishes the first 'Beginner' tasks, she then moves onto 'Intermediate' tasks, which use new, more advanced profile tasks instead, again with five tasks.

address' as the first two steps, and check them off immediately.[9] Kivetz, Urminsky, and Zheng (2006) doesn't investigate the impact of the decision about whether to continue showing completed tasks or not. It seems that it would be useful to do so, in order to give users a better understanding of just how close they are to the goal. Without that feedback, they may forget how far they have come, and perhaps this will affect their effort levels.

Ariely and Norton (2009) suggests that a reason for the goal-gradient hypothesis may be that people are engaged in "conceptual consumption," the theory that people may consume *concepts* in the same manner that they do tangible goods. The desire to consume goals drives people towards achieving them. Ariely and Norton (2009) notes "a goal can come to serve not as a motivator to engage in some desired behavior, but, ironically, as a goal in and out itself." Task Queues place goals directly in our path,

---

[9]Note that Kivetz, Urminsky, and Zheng (2006) didn't test to see whether people accelerate towards goals during single-visits, so we cannot extrapolate whether this is true in very short timeframes.

178

and our desire to consume them, like Pac-Man steadily gobbling up pills, is insatiable.

Another aspect of TASK QUEUES worthy of note is that they constantly remind users of tasks that are left unfinished. The *Zeigarnik effect*, named after Russian psychologist Blume Zeigarnik, observes that people feel tension and intrusive thoughts about goals that are started but incomplete. She found that people found a strong draw to resume incomplete tasks. Schiffman and Greist-Bousquet (1992) studies two groups of people given a simple anagram solving task, where each group was shown a list of all the problems available, providing a "well-defined goal directed task." One group was allowed to finish, while the other was constantly interrupted. The interrupted group experienced frustration and feelings of failure, which the authors speculate was due to having the tasks left undone. Rigby and Ryan (2011, p. 110) argues that TASK QUEUES in Massively Multiplayer Online Role-Playing Games exhibit the Zeigarnik effect, and provide a strong driver to keep players engaged in the game:

> It is uniformly hard to log off because in MMOs, you are always in the middle of something, indeed usually in the middle of several goals simultaneously. Most MMOs allow you to track dozens of very specific unfinished goals simultaneously, each with their own promise of meaningful reward. This creates a feeling of ever-present unfinished business.
>
> MMOs are designed so your list of tasks is never done... Quests are often linked together in a series that helps move a story along, but never provides much closure. If I need to find 12 jewels to complete my quest, I will not stop at 11. Even when I have all 12, if yet another quest is then offered to me in the chain—perhaps finding the golden setting into which all 12 gems fit—then I will not be likely to quit either. And so it goes, while my dinner gets cold and my wife contemplates how heavy an object to throw at me.

TASK QUEUES can be specialized into a *Toxboe*: TUNNEL. *Toxboe*: TUNNELING involves stripping away the rest of the app so that the user is locked into the task queue, and ideally only leaves when finished. Examples include sign-up processes, or shopping cart checkout systems. Tunneling is effective because it makes it easier to go through the TASK QUEUE, and users value the consistency afforded to them (Fogg 2003). While users have the *Independence* to commit to the goal of finishing the tunnel

179

at the beginning, Fogg (2003) argues that they lose autonomy once inside. Rigby and Ryan would likely disagree; the only options available once inside a well-designed tunnel are the pertinent ones that the user would want to take (such as adding or subtracting items from the cart, adding a credit card etc.), and thus users have the autonomy to make the meaningful choices they require once in the tunnel. If a user is stuck, unable to make a choice (such as being unable to edit the contents of her shopping cart), that's when a reduction in *Independence* occurs. Fogg (2003) also notes that tunnels have ethical concerns, such as only allowing a user to finish installation of a program after they've handed out their personal details (or provide false information), and so tunnels should always include an escape mechanism without causing any damage.

Watch for: When designing the tasks that should be included in a TASK QUEUE, Deturding (2011) writes that the tasks should be "interesting (as in novel), meaningful (as in relevant to the user's goals and values), and most of all, well, challenging (as in neither boring nor frustratingly hard)." Games are very good at choosing meaningful tasks, whereas apps are less so. In the *LinkedIn* example, *LinkedIn* does a poor job of proving to the user that she should find them meaningful, as the act of filling out forms will never be interesting or challenging. *LinkedIn* presents task after task, asking the user to do things like "Add a few skills you have," but never justifies why. A more meaningful formulation might be "Add a few skills you have, and we'll be able to put you in contact with people in your field." This would add a small amount of meaning to the task, and help motivate users through the queue.

Chapter 8

Motivational dark patterns

8.1 Overview

Motivational design poses ethical issues. While we are battered throughout the day with designs intending to persuade us or motivate some activity—be it slowing down the car (Goetz 2011), spending more time in the supermarket (Berlinger 2012), or the constant barrage of advertising—practitioners must still be wary of the ethical implications of designing for retention. One man's retention is another's ensnarement.

In this chapter, *motivational dark patterns* will be introduced.[1] These patterns violate user expectations by encouraging them to give up or jeopardize some resource (time, money, social capital). The loss of these resources may cause users to experience reductions in *Acceptance*, *Honor*, *Independence*, *Saving*, *Social Contact*, *Status* and *Tranquility*.

---

[1]This work was also performed with Staffan Björk, an associate professor at the University of Gothenburg, many of whose patterns from *Patterns in Game Design* have already been presented, and Jose Zagal, an assistant professor at DePaul University, who edited *The Videogame Ethics Reader*, and was first author on "Towards an Ontological Language for Game Analysis." The ideas of categorizing the work by the depleted resource, as well as the patterns themselves, were developed collaboratively. The idea of manipulation literacy, and how dark patterns relate to play styles they support, was developed on my own. Aspects of this work has been previously published in Zagal, Björk, and Lewis (2013).

8.2 Defining dark patterns

8.2.1 Anti-patterns

Before we get to motivational dark patterns, it is important to understand the difference between them and anti-patterns, an earlier term which appears to have much in common. Anti-patterns arose from software engineering, and the term was suggested by Koenig (1998), inspired by Gamma et al. (1994). He describes antipatterns as "just like a pattern, except that instead of a solution it gives something that looks superficially like a solution but isn't one." This idea was taken further by Brown et al. (1998), who stated that an antipattern initially appears beneficial, but proves not to be, and that there is some alternative solution that actually works.[2]

8.2.2 Dark patterns

Dark patterns differ from anti-patterns. Dark patterns imply some *intent* on the designer to do something that is not in a user's best interests. They were first suggested by Brignull (2010a), and later presented in Brignull (2010b). Brignull describes them thusly:

> Normally when you think of "bad design," you think of laziness or mistakes. These are known as design anti-patterns. Dark patterns are different—they are not mistakes, they are carefully crafted with a solid understanding of human psychology, and they do not have the user's interests in mind.

He collects user interface dark patterns at `http://darkpatterns.org`. In Brignull (2011), he describes how psychological insights can be used for both light (honest) and dark (dishonest) purposes, reproduced in Table 8.1.

---

[2]The reasoning for the latter clause is fairly intuitive: there's no value in pointing to something and saying "that looks bad" without having a better means of doing it. Things that are inherently bad aren't anti-patterns, as they're the best way we know how.

Psychological Insight	Applied Honesty (benefits users)	Applied Deceptively (benefits businesses)
"We don't read pages. We scan them." — Steve Krug	*Aid rapid comprehension:* ensure key content is shown in headings, subheadings (etc), using a strong visual hierarchy.	*Hide key information:* Bury facts within paragraphs of text, so some users will proceed without fully understanding the transaction.
"People tend to stick to the defaults." — Jakob Nielsen	*Prevent mistakes:* Default to the option that's safest for the user. In important contexts, don't use defaults and require the user to make an explicit choice.	*Benefit from mistakes:* Ensure default options benefit the business, even if this means some users convert without meaning to.
"People will do things that they see other people are doing" — Robert Cialdini	*Show unedited feedback:* Allow real customers to share their experiences, so they can make accurate pre-purchase evaluations.	*Bury negative feedback:* Hand-pick positive feedback and display it prominently. Bury negative feedback so it is hard to find.

Table 8.1: A reproduction of examples of psychological insight being used for both beneficial and negative purposes, taken from Brignull (2011).

Anti-patterns are mistakes, either encouraged by ignorance or lack of resources. No-one really sets out to make a bad piece of software, but all development is constrained by something: team expertise, financial or temporal budget, or even audience expectation. So, anti-patterns slip in, and by the time the mistake is spotted (if it is at all), one of the resources has dried up and the pattern can no longer be removed. Dark patterns are something else. Dark patterns require intent.

Dark patterns are difficult to spot, and how they are defined is entirely subjective. From an external viewpoint, trying to infer whether a pattern is simply "Bad Design" or the more insidious "Designing for Bad" is tricky. We don't have the ability to go back in time and find out the motivations and decision-making that was made throughout a product's development.

8.3 Manipulation literacy

One facet of dark patterns is that, for them to be dark, they must be successful at performing manipulations without the user's consent. When consent is given, however, users are allowed the *Independence* to choose what they believe to be best for them, and are meeting motivational needs (even if an external viewpoint may call the decision-making into question). To be able to give consent, a user must first be made aware that she is entering into a manipulative environment, which requires "manipulation literacy."[3] Manipulation literacy describes the ability for users to identify manipulative techniques, understand their possible effects, and then make consensual, informed choices about how to proceed. For a concrete example, consider casinos. In the case of gambling, someone who believes she may be susceptible to risk-seeking might resolve to extract $200 in cash from an ATM, then leaves all her bank cards in her room before entering the casino, thus ensuring that the maximum lost money will be $200. This person has used her manipulation literacy to identify a coercive environment, and formulate a strategy for how she will deal with it. When dark patterns are applied to low manipulation literacy users, they can be coerced and misled. When dark patterns are employed with a high manipulation literacy audience, they become ineffective, and are ordinary interaction design patterns.

A user cannot give reasonable consent to manipulation if one does not have the literacy with which to understand when persuasion is occurring and how it is being conveyed or effected.[4] While most of us have a grasp of spoken and written rhetoric, and where we can expect to find it (Berdichevsky and Neuenschwander (1999) uses the example of a used car salesperson being someone we are less likely to trust that someone who has no stake in selling the car), we do not yet have such an understanding when

---

[3]This idea was developed independently from Björk and Zagal, who offered feedback later.

[4]Especially given that even when we are aware of our own biases, we still find it difficult to act against them (Kahneman 2011).

we are faced with persuasive technologies and new media.

We have seen new literacies with new media before. For example, both Bogost (2005) and Mateas (2005) argue for the teaching of "procedural literacy," defined by Bogost (2005) as "the ability to reconfigure basic concepts and rules to understand and solve problems." Procedural literacy, once the domain of programmers alone, is as useful to those who consume programs as it is to programmers. Much as Bogost and Mateas champion the expansion of procedural literacy past programmers, manipulation literacy should extend to users as well. An understanding of how software effects changes of mood or behavior, and how that might encourage certain decision-making, could prove beneficial, not only for users to understand how they might better choose applications, but also to protect themselves from applications that might seek to manipulate them in an undesired way.[5] Returning to our conception of dark patterns, once users are literate enough to understand the effects of a pattern so that they can give consent, the pattern is no longer dark. This is the most likely reason that many of the dark patterns we identified come from game genres and software platforms that have emerged relatively recently, where developers are are looking to novel ways of monetizing their user base.[6]

Once manipulation literacy has developed in a user, she may well look upon manipulation techniques with disdain, and might opt-out of similar applications altogether. With respect to games, Miller (2012) reports on a panel where Nik Davidson of Amazon suggests that "[t]he long-term danger [of employing psychologically manipulative design techniques in games] is that we are poisoning the well; we're watching a large-scale tragedy of the commons play out on our player bases. Our audience is

---

[5]This could be seen as analogous to the "Defense Against the Dark Arts" class from Harry Potter.

[6]Anecdotally, it seems that players of social games are much more aware of dark patterns than when they were first launched. However, a look into whether social game audiences might be gaining manipulation literacy was inconclusive. When faced with trying to take an audience pulse, Google Trends can be used to look at how searches have changed. Searches with terms such as [`farmville evil`], [`farmville scam`], [`farmville bad`] tailed off dramatically in 2011 and 2012 (FarmVille being chosen due to its extensive popularity across a number of years), but replacing [`farmville`] with [`facebook`] in the same queries shows a steady rise in searches.

becoming inured to the viral trickery we employ to get people to do what we want them to do."

8.4 Going from dark to light

8.4.1 When a dark pattern is not a dark pattern

During a private conversation, John Romero[7]—who had been working on a social game as part of his startup—spoke about various patterns in social games which seemed opposed to the player. He noted how much he enjoyed DARK PATTERN: INTERACTION BY DEMAND, *when used correctly*. DARK PATTERN: INTERACTION BY DEMAND is when the user must conform the schedule that the application desires, either drawing the user in or pushing her away, and is described in detail in Section 9.1. *World of Warcraft* uses DARK PATTERN: INTERACTION BY DEMAND in some throughly enjoyable instances.

One of these is *Humar the Pridelord*. *World of Warcraft* offers a hunter class. Hunters are able to tame certain animals from the wilds, and these pets become allies to fight alongside the player in battle. Pets are always alongside the player, so while many choose pets only for their particular gameplay attributes, some choose them for vanity alone. Humar was a particularly prized catch, as he was the only black lion in the game, as shown in Fig. 8.1. However, Humar would only appear (gamers call this "spawning") very rarely, so only the most committed (or lucky) hunters would be able to tame him. These committed players are part of a DARK PATTERN: INTERACTION BY DEMAND mechanic, bending their lives around the game's schedule of when Humar is available.

The joy of Humar is the meta-game that surrounds him, which plays out across chat channels and the Internet. Where does he appear? What time does he appear?

---

[7]Creator of *Doom*, and one of the first true "rock star" game developers

Figure 8.1: Humar the Pridelord. *Created by fyrfli , $http://www.flickr.com/photos/fyrfli$. Used under Creative Commons Attribution-NonCommercial-Sharealike 2.0 license. This version of the image was cropped from the original, and is provided under the same license.*

Has anyone seen him recently? Players become mini-hunters themselves, stalking their prey across the Net, sharing rumors that become myth. At "Humar the Pridelord," players come together to share their knowledge:

- Take the western road out of Ratchet. As you reach the top of the hill turn to the right and follow the mountain back to the northeast. As of patch 4.0.3, his tree is right inside the mountain range. You'll know the tree if tracking beasts on your map because the other lions form a perfect circle around the tree. Type in `/target Humar the Pridelord` and if he's there, you'll know it. Otherwise, you wait.

- It is said that his spawn rate ranges from 8-12 hours. This timer resets whenever he is killed or tamed.

- Hunters who seek to tame Humar, should expect some competition. Try to get on at a non-busy server time, like a weekday lunch, and you might have better luck.

- In Cataclysm, perhaps as an homage to players waiting hours on end to tame Humar, Blizzard has included a small camp with a tent and fire near his tree.

187

This is a thrilling game in and of itself, and highly enjoyable for certain players. But why do they enjoy it, when they are stuck in a clear institution of INTERACTING BY REQUEST? The answer lies in the concept of support.

8.4.2 Support

Juul (2009) introduces the idea of *support*: that certain game mechanics *support* certain play styles. Players may flip between different play styles as they see fit (as Yee (2006) puts it: "just because you like ice-cream doesn't mean you will hate pasta"), and games can support different styles. *Rock Band 3*, for example, will let players simply bash away at the drums without rhyme or reason if the player chooses a free play mode, but goes all the way to a professional mode that will teach the player how to play the electric guitar. Role-playing games come with various classes that support different play styles,[8] and online games can let players take control of shops or define themselves through crafting, guild management and mediation, or even assassination plots (Francis 2008).

Using the idea of support, it is easy to see how patterns can lay out across a spectrum of darkness, rather than as a binary idea. Dark patterns are nullified when they support a player's play style. When the majority of players must engage with a pattern that does not support them, then that pattern's use can be considered dark. This idea can be adapted to motivational software in general. When a dark pattern offers a user the ability to meet their motivational needs, then its use is light. When a dark pattern forces the user to interact in a way that is against their needs— such as displaying power when they wish to be submissive, or force social interaction when the user wishes to be alone —then the usage is dark.

When we look at DARK PATTERN: INTERACTION BY DEMAND, we can see

---

[8]Although, notably, almost all these styles are combat driven, rather than allowing for other means of conflict resolution.

why Humar was enjoyable: the player *chooses* to engage with the pattern. Humar is by no means forced upon the player, but exists as a challenge if she wants it. However, DARK PATTERN: INTERACTION BY DEMAND used as a withering mechanic in a social game is dark. Crop harvesting is required for progression, and players are must return to the game if they have the time or not.

8.4.3 Disclosure

The other surefire way of changing a dark pattern to a light use is to simply follow the disclosure principle from Berdichevsky and Neuenschwander (1999):

> The creators of a persuasive technology should disclose their motivations, methods, and intended outcomes, except when such disclosure would significantly undermine an otherwise ethical goal.

Here, an example of an "otherwise ethical goal" might be a manipulation to encourage users to end substance abuse. It is difficult to define objectively what these "ethical goals" might be, and it seems prudent to assume this clause does not exist when designing software.

As an example of this put into practice, one could imagine a *FarmVille* splash screen stating the average amount of time and money spent by players. Disclosure has a knock-on effect of educating users and helping them to understand what to look for in applications, increasing their manipulation literacy.

8.5 Organization of dark patterns

To divide up the dark patterns, an economic approach is employed: what resource is the user being asked to expend? They are divided up into three categories: time, money and social capital.

The patterns were gathered by analyzing games first-hand, looking at descriptions of design strategies by professional designers, and monitoring critical and user

189

reactions on the Web. Any examples are just to help aid with the description of the pattern, and not to single out any particular developer or designer (they're patterns because they appear more than once).

Chapter 9

Temporal dark patterns

9.1 Overview

Temporal dark patterns are patterns that request more time from the user, or provide less entertainment time than expected. If users are unsure about how much time will be committed, or their expectations of the time required was significantly off, or that they feel they've wasted their time, then there is a temporal dark pattern.

Temporal dark patterns are dangerous as their power is already well-known. GRINDS, in particular, are used throughout many applications with the tacit knowledge that they encourage users to hand over more time than they otherwise would like to.

Instead of using temporal dark patterns, first focus on creating an application that is already meeting a variety of Reiss desires. An intrinsically motivating application will attract engaged users for a more meaningful, long-term relationship, than those who are stuck in a temporal dark pattern.

9.2 Dark Pattern: Grind

Description: Repetition of a skill-less task in order to progress

Reduces Reiss desires: *Acceptance*, *Independence*, *Power*

Figure 9.1: A screenshot from a *Max Payne 3*, showing Dark Pattern: Grinds in the game. *Max Payne 3* even owns the use of the pattern, by calling this the "Grinds" system. *Taken from* `http://www.gamesradar.com/max-payne-3-achievement-guide/`, *used under fair use.*

Related to: Leaderboard

Examples: *Final Fantasy XI*, *Halo 4*, *World of Warcraft*

Use: Dark Pattern: Grind is a pattern that primarily applies to games, but can also apply to applications with gameful patterns that offer extrinsic rewards to users. Leaderboards and Badges, in particular, are likely to include Dark Pattern: Grind as a knock-on effect if the goals are designed poorly. As Dark Pattern: Grind is more well-known within games, discussion here will be limited to just games.

Grinding is generally defined as repeating some task in order to progress ("Grinding - WoWWiki"; "Grinding - Wikipedia"), sometimes with the added clause that the task be tedious ("Grinding - Giant Bomb"; Sorens 2007). Neither definition is particularly satisfying. The former definition is too abstract, as many/most games

192

could be categorized as doing the same task over and over (shoot monster, run here, shoot that monster, run there). The *Halo* series provides a good example of this. *Halo* is well-known for its "30 second of fun" design ethos, as described by Thompson (2007):

> [T]he Bungie crew came up with a mantra that would eventually guide all aspects of Halo gameplay: "30 seconds of fun." The idea was to have Halo repeatedly immerse players in hectic battles that would last for half a minute—just long enough to create heart-thumping chaos and the risk of death—before offering a respite.

This 30 seconds definition would fit under the loose umbrella of a "repeated task," and so the general definition of grinding would hold, but *Halo* is not described as a grind.

This dissertation proposes a marginally different definition, to help us better understand grinding:

Grinding is a series of similar tasks a player performs in order to achieve a reward, and that reward does not require skill to achieve.

Splitting up this definition creates what could be termed the "Three Pillars of Grind":

1. Similar tasks are repeated.

2. The reason the player is performing the tasks is to gain a reward.

3. The reward is not tied to the raising of player skill.

DARK PATTERN: GRIND, then, is a dark pattern that provides these three pillars.[1] DARK PATTERN: GRIND is often coupled with various ideas from behavioral economics, such as goal-gradient hypothesis (leveling systems) and ratio schedules (item

---

[1] There is a similarity to Skinner boxes, which was also noted by "Grinding - Wikipedia." The article put forward "pushing the bar" as a synonym for grinding that might have its roots in Skinner boxes.

193

drops), which helps explain why the Dark Pattern: Grind pattern can motivate engagement, even when the task is intrinsically unsatisfying.

In some instances, a Dark Pattern: Grind can offer *Status*, as those who have achieved their goal are able to boast about it to others. It is likely not a coincidence that grinds are usually associated with visible goals that can be shown to other players, such as titles for avatar names, or a particular Badge, thus enhancing the *Status* gained. To ensure players don't leave once they achieve a certain Dark Pattern: Grind, games designed around retention, such as MMOs, will often provide a number of different Dark Pattern: Grinds to partake in, so players always have a new goal to embark upon, creating a hedonic treadmill.

It is tempting, given the psychology literature, to assume that once available Dark Pattern: Grinds are completed, users will stop playing the game, as the Dark Pattern: Grind for an extrinsic reward has eroded any intrinsic motivation to play the game. However, given the findings of researchers in regards to *Stack Overflow* (4.1) which showed only a slight drop-off, there isn't any public evidence to support this.

Returning to *Halo*, let's look at how a Dark Pattern: Grind is created. In the *Halo* games before *Halo 4*, multiplayer players were all placed at equal footing. Each had equal access to equipment or skills. This changed in *Halo 4*, where a leveling mechanic was introduced, and players earned "Spartan Points" for leveling up. These points were required to unlock equipment and powers. During early levels, where players have not unlocked everything, some players may be at an advantage over others in map and game type combinations (e.g. if a player chose to unlock a long-range weapon, she would have an advantage on a large map over a player who chose a close quarters weapon). This system introduced Dark Pattern: Grind into a game that previously did not have it, which upset some players, such as Hernandez (2012):

> The other day, a colleague mentioned that she felt like there was some-
> thing off about Halo 4's multiplayer. She was getting destroyed by other
> players, eventually feeling like she didn't have much of a chance when up

against people with advanced abilities or gear gained from Spartan point unlocks.

This seemed like a marked difference from earlier Halo titles, where it was possible to drop in with your starting gear and have a reasonable shot at being competitive - even against people who totally out-leveled you...

I thought about it, and it hit me: why in the world do I act as if this is okay? Simply because unlocks are so common now, and just because you eventually reach a point where you have everything you need to be competitive, doesn't erase the fact that the game starts out unbalanced.

Revisiting the three pillars, here is why the leveling implemntation in *Halo 4* can be described as DARK PATTERN: GRIND:

1. Similar tasks are repeated (each round of a multiplayer game will be broadly similar to another).

2. The reason the player is performing the tasks is to achieve a reward (players don't feel competitive until they have the unlocks they think they need).

3. The reward is not tied to the raising of player skill (experience points are mostly given for completing matches, and not for in-game performance).

DARK PATTERN: GRIND is a temporal dark pattern because players must invest more time than they originally envisaged. DARK PATTERN: GRIND usually plays off of a player's competitive nature, such as in this *Halo 4* example. Other examples include *Call of Duty 4: Modern Warfare* (the game that introduced this unlock pattern into multiplayer shooters) and *World of Warcraft*'s Player vs. Player combat (a level 85 character can repeatedly easily kill, or "grief," a lower-level character, so low-level players are forced to grind in order to protect their enjoyment of the game). If players refuse to engage in grinding, their enjoyment of the game is spoilt.

It is worth noting that Koster (2003) advocates for DARK PATTERN: GRIND as "not only good, but necessary" for multiplayer games:

195

You must find some way to overturn the rich get richer scenario in all games with some form of accumulation. There must be not just drains, but occasional catastrophic behaviors that topple the 20% at the top of [the power law curve], so that others get a chance.

Koster is discussing how the top 20% of the player population dominates and dissuades the lower 80% from playing. He sees the investiture of time as a way of nullifying skill, so that the 80% who will never be as skillful as the 20%, have a means of remaining competitive. One possible counter to that argument is that there are some people who will never have the time to invest into grinding, so this is simply another form of exclusionary design. The only real way of avoiding a "rich get richer scenario" is matchmaking that will ensure players are matched against those who they will find an exciting, and *level*, playing field.

9.3 Dark Pattern: Interaction by Demand

Description: Forcing users to engage with the application on its schedule

Reduces Reiss desires: *Independence, Tranquility*

Also known as: Playing by Appointment

Related to: Notification, Encouraged Return Visits (Björk 2011), Dark Pattern: Grind

Examples: *Farmville, Tiny Tower, TuneIn*

Use: Dark Pattern: Interaction by Demand is a name given for designs that force users to interact with the application on its own schedule, either by drawing absent users in, usually via Notifications, or pushing active users away, usually by denying them content until some time has passed. Dark Pattern: Interaction by Demand reduces user *Independence* to choose when they interact with the application.

One common Dark Pattern: Interaction by Demand method is to send a Notification to the user for dubious reasons, requesting she return to the application.

196
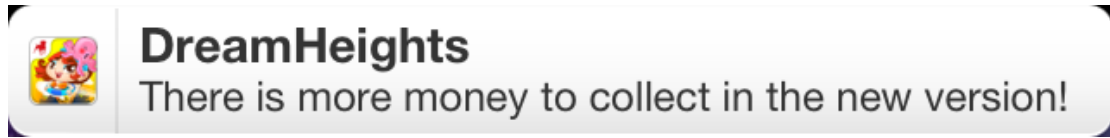
Figure 9.2: A screenshot from a *Dream Heights* notification on iOS. This notification requests users to upgrade.
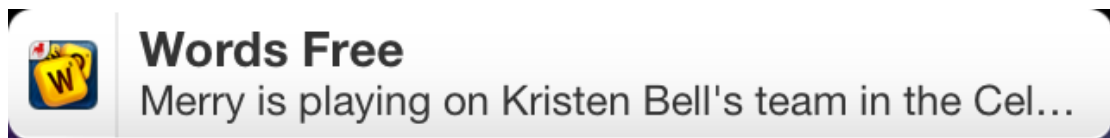
Figure 9.3: A screenshot from a *Words With Friends* notification on iOS. This notification is informing the user that a friend on *Facebook* has joined a team in their Celebrity Challenge. It does not pertain to any current games that are being played, and is a tenuous reason to send a notification.

Figure 9.4: A screenshot from a *TuneIn* notification on iOS. This notification requests users open the application to hear about Hurricane Sandy.

Figs. 9.2 to 9.4 shows a number of notifications. None of the notifications pertain to anything actively happening in the user's interactions with the application. These notifications only exist to ask the user to reengage with the application. Apple's own Developer Guidelines for notifications prohibit such usage:

5.5 Apps that use Push Notifications to send unsolicited messages, or for the purpose of phishing or spamming will be rejected

5.6 Apps cannot use Push Notifications to send advertising, promotions, or direct marketing of any kind

Such usage enrages users, as described in Sadun (2012):

Want to know a quick road to a bad app review from your users? Local notification spam. Frankly, we're sick of it...Just because local notifications don't require opt-in doesn't mean that developers should abuse them for marketing. Here's a rule of thumb: if your notification doesn't deliver information that your user specifically requested, don't send it. This goes for push notifications as well. When users opt in, they're not opting in for spam.

This usage does not apply to the use of NOTIFICATIONS when they provide information pertinent to the activity the user has taken. For example, many social games limit the amount of interactions a player can make in a day by using a harvesting mechanic, which in and of itself is a use of this pattern by forcing users away. The harvesting mechanic requires seeds to be planted, and then only when they have grown after a period of time can they be reaped (other games use metaphors such as transportation of goods or construction). An appropriate NOTIFICATION would inform the user that their plants are ready to be harvested. However, even if the individual NOTIFICATIONS are warranted, INTERACTION BY DEMAND can also occur if NOTIFICATIONS are sent too frequently. If a NOTIFICATION is sent many times a day, the purpose is clearly to control the user, and again, *Independence* is reduced.

Developers often send such unsolicited notifications via email, of which we can all think of particular examples. Such emails are effective at driving users back to the

site (*Email: The Easiest Way to Improve Retention*), but are often worded in such a way to minimize the *Independence* of the user. One particular example of this was sent from the online bank *Simple* to users who had stopped using the service. The subject line read: "Have you been cheating on us?" This sets up a relationship where the user is expected to reengage with the application, if the user doesn't, she is accused of being dishonest (a reduction of *Honor*). DARK PATTERN: INTERACTION BY DEMAND is the application itself acting dishonorably, and users should never be expected to comply with such requests. Sending an email worded this way showed an immense lack of understanding of users' personal needs.

Another form of this pattern is to force active users away, demanding that they come back later. This most often occurs in video games, such as with the previously mentioned harvesting mechanic, or with the previously mentioned schedule forced upon hunters of Humar the Pridelord. This use of the pattern can often be a light use, as long as the schedule conforms to the player base's wishes. For example, harvesting is often seen as a way of providing bite-size gameplay that better fits into busy players' days. As with all dark patterns, the key to using this pattern in a light manner is disclosure, by letting the player know how long she will need to wait.

9.4 Dark Pattern: Hellbroadcast

Description: FILTERING a user's BROADCASTS without consent.

Reduces Reiss desires: *Independence*, *Social Contact*, *Tranquility*

Related to: FILTER, BROADCAST

Examples: *Facebook*, *Hacker News*, *Something Awful*

Use: DARK PATTERN: HELLBROADCAST takes its name from "hellbanning." Hellbanning is a method of banning users without their knowledge.[2] They can continue to

---

[2]DARK PATTERN: HELLBROADCASTING is a pattern created independently of Björk and Zagal.

interact with the site, and even post BROADCASTS, but these are visible only to that user.[3] This inoculates the rest of the community from the problem user, without that user realizing she is banned and then attempting to circumvent the ban by creating a new account. The intention is for "the troll [to think] everyone has learned to ignore him, and he gets discouraged and goes away." (Thompson 2009) This pattern is called DARK PATTERN: HELLBROADCAST in order to provide consistency with the naming of BROADCASTING, but also to invert the issue from an action that the administrators of the community perform, to focus on how this affects a user's motivational state.

DARK PATTERN: HELLBROADCASTING is used by a number of communities, including *Something Awful* and *Hacker News*. While a user has a hellban, she might begin to lose feelings of *Social Contact* as others will no longer reply to her. As one user on *Hacker News* wrote (`http://news.ycombinator.com/item?id=5270855`):

> I had an account hellbanned here for over a year before someone finally told me. There's no way to find if someone replies to you on this forum so I rarely went back and checked if anyone replied to me[4], but it still bothered me that the admins would find it acceptable to let someone waste their time over an entire year without telling them their account is useless.

Users who are deliberately trolling will be unable to meet theirs needs of *Power* and *Vengeance*, as they can no longer bait others into arguments.[5] Once a user realizes she has been hellbanned, it's reasonable to believe her *Tranquility* suffers, as she wonders if she's been hellbanned from other sites too.

HELLBROADCASTING can be seen as a necessary evil for the good of the group, preventing the group's overall requirement for *Tranquility* being put in jeopardy. While HELLBROADCASTING is explicitly designed to prevent disclosure of a ban, it flirts with the "otherwise ethical goal" clause of the disclosure principle. The question that remains

---

[3]One variant is for all hellbanned users to see each other, subjecting them to the style of problem BROADCASTS that they had given others. *Max Payne 3* uses a similar system, where players caught cheating being banished to a "Cheaters Pool," where they can only play against other cheaters (K 2012).

[4]Actually, *Hacker News* does let users see who replies to them via the comments page on their user profiles.

[5]Albeit what counts as a troll, and what counts as legitimate debate, is often highly controversial.

up for debate is whether hiding problem users from others counts as ethical. Surprisingly, *Facebook* implements its own form of monetized DARK PATTERN: HELLBROADCASTING and applies it to fan pages and brands. Each post a fan page or brand makes is only visible to less than 15% of their subscribers (Holiday 2013). Essentially, these posts are turned into DARK PATTERN: HELLBROADCASTS that don't appear in 85% of users' news feeds. If they want their posts to reach a wider audience, they must pay *Facebook* for the privilege.[6] *Facebook* claims they're trying to stop spam (Lyons 2012), and from a certain viewpoint, this would seem reasonable. From another viewpoint, it might also seem that these posts are advertisements, and as such, should also require payment. However, not only does this undermine the motivational desires of brands, it also undermines those of their fans. Fans *choose* which pages they subscribe to. They have opted-in to seeing these posts, but *Facebook* selectively filters them. This undermines a fan's *Tranquility* and *Independence*, as they must now check their favorite fan pages in order to make sure they don't miss out on a deal or exciting news snippet. The monetization of HELLBROADCASTING hurts everyone, except *Facebook*, and is a particularly dark implementation.

---

[6]The Dallas Mavericks have to pay around $3000 for a post to be seen by a million fans (Lyons 2012).

Chapter 10

Monetary dark patterns

10.1 Overview

If time is money, then it should come as no surprise that if there are temporal dark patterns, there are monetary dark patterns as well. A cynic would view these patterns as a means for companies to extract extra money from users, but no pattern comes close to anything one could characterize as *fraudulent* or *illegal*, so a different view might conclude these patterns are actually good short-term business.

Users that regret spending money, lose track of how much money they will spend, or don't know how much money will be required in order to progress, are engaged with a monetary dark pattern.

Monetary dark patterns are easily avoided: tell the user how much money is expected in real currency, give them a trial period, and offer them the ability to refund purchases that they turn out to not like. *League of Legends*, for example, is monetized by offering a set of different champions for players to play. Each week, ten new champions are offered for free on a rotation. If players wish to continue playing with a champion outside of the rotation, they have to pay. The company also offers three refunds to each account automatically, so users who do end up having buyers remorse can undo

the damage. Treating users with respect is good long-term business; think of companies like Amazon and Costco who create lifelong customers through generous return policies.

10.2   Dark Pattern: Currency Confusion

Description: Substituting money with an arbitrary currency, leading the user to being unaware of how much money she is spending

Reduces Reiss desires: *Independence*, *Saving*, *Tranquility*

Examples: Microsoft Points, Facebook Credits, *CastleVille*

Use: DARK PATTERN: CURRENCY CONFUSION is a pattern where real money is taken from users, and converted into an arbitrary secondary currency. There are two legitimate reasons to do this:

1. To create a global currency, where users can express the prices for something without the need for conversion to local currencies. *Castle Crashers* on the Xbox 360 costs 1200 points the world over.

2. Interfaces do not require localization. The monetary exchange encapsulates exchange rates from the rest of the program, so there is no need to worry about how to express Indian Rupees and US Dollars in the same interface throughout the application.

While these do have their benefits, the undesirable effect for users is that they are no longer able to discern what things cost. In the United States, $5.00 buys 400 Microsoft Points. Renting an HD movie costs 480 Points. Very few people can perform the currency conversion in their head (it's actually a flat $6.00). However, this is just in the USA. In other countries, things are even trickier. In the UK, 400 Points costs £3.50. 480 Points is now even more difficult to calculate (it's £4.20). The extra sting in the tail, as noted by Madrigan (2011), is that the 480 Points doesn't divide evenly

into the multiples of 400 that Microsoft allows users to buy, leaving points left over. Madrigan theorizes that waste aversion, an off-shoot of sunk cost fallacy, leads us to want to spend the remaining points in order to fully utilize our money, buying things we otherwise wouldn't have done.

Madrigan (2011) also points out a trick that Microsoft missed, which Zynga and other social game companies have not. Anchoring is a phenomenon studied by behavioral economics where people tie their mental calculations to the first number they see, then use that number as the guiding means of estimating product valuation from that point on. One particularly relevant experiment was performed by Ariely (2010), who asked students to write down the last two digits of their social security number, then bid on items whose value they did not know, such as fine wine or chocolate. Students whose SSN ended between 80 and 99 placed bids that were 216 to 346 percent higher than those whose SSN ended in 01 to 20! Their brains were primed by the higher number, and so their estimates were off.

Microsoft Points don't use anchoring, as Microsoft Points translate to higher values than the money put in. $5.00 translates to 400 Points, and anchoring causes the 400 Points looks much more expensive. As Madrigan (2011) puts it, "Microsoft could actually be letting us off easier than they could if they gave you just .8 points for a dollar and charged 12 points for a new game. But shhhh! Don't tell them!"

Zynga uses anchoring in their social games to price items. Zynga games utilize two currencies, one which will be termed the "premium" currency, which can either be bought by Facebook Credits (another level of indirection) or earned very slowly in-game. The other currency, which will be termed "in-game" currency, is offered in large amounts often. At the time of researching, in *CastleVille*, the premium currency, crowns, cost $2.00 for 15 ($0.133 a crown). To buy in-game currency, it also cost $2.00 ($0.00166 a coin). None of the in-game items have any tangible value, so players are unable to discern what their value is. Consider the case of the Jester Leggings and

204

Figure 10.1: A pair of Jester Leggings from *Castleville*. These cost 7000 coins, a currency that players get in-game.

Figure 10.2: A pair of Lederhosten from *Castleville*. These cost 2 crowns, a currency that players primarily purchase with real money.

the Lederhosen, in Fig. 10.1 and Fig. 10.2 respectively. Which one is worth more? It's hard to shake the feeling that the Lederhosen is much less than the Jester Leggings, but that's anchoring fooling us. The Jester Leggings are easily purchased with the free, in-game currency, whereas the Lederhosen can only be bought using with the premium currency, and hence, real money.

But the plot thickens! If we were to buy the the Jester Leggings using purchased coins, they would cost an astonishing $11.62. Why encourage players to buy the much cheaper Lederhosen? It may be that because the Jester Leggings are purchasable by in-game currency alone, they are essentially free (especially to a younger audience who have much more free time, and the "time is money" axiom doesn't hold for them). The other possibility is that the Jester Leggings aren't supposed to be purchased at all, and exist only to provide an anchor for the Lederhosen. At first read, this sounds insane, but is actually a well-used technique. Consider a restaurant. At the top of the menu is always an extravagantly priced steak. Restauranteurs don't expect to sell many of these, but provide them to create an anchor that makes diners think other expensive items are cheaper. How many times have you ordered a wine bottle from the middle of the menu? (Speiser 2009).

10.3   Dark Pattern: Monetized Rivalries

Description: Exploiting user competitiveness to incentivize purchasing

Reiss desires: *Vengeance*, *Social Status*

Reduces Reiss desires: *Independence*, *Saving*

Related to: Leaderboard

Examples: *Robot Unicorn Attack: Evolution*, *Words with Friends*

Use: Dark Pattern: Monetized Rivalries exploits user competitiveness, encouraging them to spend money they would not otherwise in order to achieve *Status*. Apps

Figure 10.3: The *Robot Unicorn Attack: Evolution* storefront. Boosters can be bought with both in-game currency and Facebook credit. Players cannot specifically select which enhancement they want, leading to multiple purchases often being required for the exactly desired enhancements.

which utilize gameful components, particularly LEADERBOARDS that expressly pit users against one another, are at risk of including an DARK PATTERN: MONETIZED RIVALRY if there is some means of payment to provide a boost. Colloquially, this pattern is sometimes described as PAY TO WIN. Järvinen notes that a "virtual arms race between individual or alliance rivals" is an effective pattern and that it can be combined with GRIND when stating "[d]esign for pay to win (but balance for grind to compete)" (Järvinen 2012). This pattern doesn't appear to have been used in a general app context. However, it would be easy to add to apps: a fictional example might be to pay $1 to gain five check-ins on *Foursquare*.

Figure 10.4: *Robot Unicorn Attack: Evolution* offers a number of different enhancements, each with different values. Up to three (or none) can be used at a time. Note how the store can be accessed directly from here.

A good example of a DARK PATTERN: MONETIZED RIVALRIES system can be found in *Robot Unicorn Attack: Evolution* (RUAE). In RUAE, the player's unicorn runs across the screen, gradually picking up speed, and the player must jump across gaps and bash through stars, with the aim to get a place on the LEADERBOARD (which is always prominently displayed at the bottom of the screen). Players can buy single-use enhancements, as listed in Fig. 10.3, which include slowing down their unicorn or adding a triple-jump ability, which make the game easier. As the LEADERBOARD makes no differentiation between players who play with or without enhancements, players need to use these to remain competitive. While they can be purchased with in-game money, they can be more quickly acquired by spending Facebook credits (Facebook credits being a use of DARK PATTERN: CURRENCY CONFUSION). When each play session ends with the message "You died, just like your dreams" and a tearful unicorn, players can replay immediately. In the event that the player doesn't place highly in the high-score table, she will probably be subject to a series of frustrating failures, combined with the goading game over message. According to ego depletion theory (Baumeister et al. 1998), she will have less self-control to resist making a purchase, and so when the enhancements choice screen appears, she will be more likely to head to the store and purchase boosters.

Some games with the appearance of a level playing field include DARK PATTERN: MONETIZED RIVALRIES. *Words with Friends*, the popular *Scrabble*-clone, allows players to purchase instruments that help players assess the game state. One instrument, the tile pile, tells players how many tiles of each letter are left. Another, the Word-O-Meter, gives feedback about how strong a word the player is attempting to play is relative to those which the player *could* play, allowing players to ensure a maximal word use before playing.

Figure 10.5: In *Star Wars Angry Birds*, the "Mighty Eagle" mechanic is implemented using a Millennium Falcon (I am sure the bird-based pun was intended). When the player is stuck on a particularly difficult puzzle, she can use a credit to have the Millennium Falcon swoop in and shoot towards a point on the screen selected by the player. The game grants limited uses of the Falcon, otherwise the player can purchase more credits using an in-app purchase. The Falcon doesn't guarantee success, as the player might still miss completely in her target selection, forcing her to use another credit.

10.4 Dark Pattern: Pay to Skip

Description: Users can pay money to skip onerous issues, sometimes issues that were added arbitrarily by the developer

Reduces Reiss desires: *Independence*, *Saving*

Examples: *Angry Birds*, *Real Racing 3*, *Sim City Social*

Use: DARK PATTERN: PAY TO SKIP is a pattern where a particular time-consuming issue is baked into the application, and then an escape means is offered by paying money.

One of the most famous examples of this pattern is from the *Angry Birds* series,

which uses a "Mighty Eagle" (a screenshot from *Star Wars Angry Birds* showing this mechanic can be seen in Fig. 10.5). The Mighty Eagle allows players to skip difficult levels by providing a large bomb with which to drop on the play field, almost always ensuring the destruction of all the pigs. The player can then move onto the next puzzle, which hopefully (and usually) is easier. This style of PAY TO SKIP is roughly analogous to the monetization of cheat codes.

A slightly more insidious use of DARK PATTERN: PAY TO SKIP is by deliberately making the game experience more tedious to encourage the use of skipping. This design trait is discussed by Luban (2012):

> Frustration is what drives players to actually spend money in a "free" game. The mechanism is simple: 1) get the players to enjoy the game, 2) give them a taste for game progression through short and medium-term rewards, 3) make new progression rewards increasingly numerous AND long to get. When they cannot wait any longer, they'll start buying. There are variations around this principle, but you get the idea.
> That sounds awful—and it is hard to imagine that any gamer, in particular demanding hardcore gamers—would break down and buy something.
> But they do.
> A successful freemium game is designed in such a way that the player will never resent that situation.

This idea is more succinctly put by Kuchera (2012):

> Free-to-play games require that some aspect of their economy is unsatisfying; that lack of satisfaction should compel you to get your wallet and pay some real money to fix the issue.

Zynga's '*Ville* series often require buildings to be "staffed" by friends in order to be completed. The reason for this mechanic is that it leads to social network effects: one person requires staff members, and so ensnares other players into re-engaging with the game, or grows the game's reach by encouraging new players, forming a DARK PATTERN: SOCIAL PYRAMID SCHEME. For players that don't want to wait, or don't want to engage with the DARK PATTERN: SOCIAL PYRAMID SCHEME, DARK PATTERN:

Pay to Skip is employed using premium currency to staff the building with an AI agent. This means that those players who won't pay lead to more players engaging, and the chances of finding a player who will get frustrated and purchase a skip increases. This means that in-game neighbors have a *direct financial value* to the company. Zynga receives a financial gain for this mechanic, which provides no value to the player at all.

Dark Pattern: Pay to Skip is often combined with other dark patterns, usually Dark Pattern: Grind and Dark Pattern: Monetized Rivalries. *Battlefield 3* uses both in conjunction with Dark Pattern: Pay to Skip. Like *Halo 4*, *Battlefield 3* introduced an unlock system to the multiplayer game. Soon after release, an update provided the chance to purchase "shortcut items" that unlocked weapons and attachments. This opportunity was touted on the game's official blog as "the perfect way to gain some ground on the veterans online" (Brun 2012). The shortcut items are examples of Dark Pattern: Pay to Skip, the mechanic being skipped is a Dark Pattern: Grind, and an added motivation to skip is included with the call to Dark Pattern: Monetized Rivalries. Clever combining of dark patterns results in an effect more than the sum of its parts.

The worst use of this pattern was discovered in *SimCity Social*. As with many social games, *SimCity Social* utilizes an energy mechanic, where each action taken expends one energy. In some games, harvesting doesn't cost energy, and in some it does. *SimCity Social* charges players energy to harvest coins from buildings. However, while players can build multiple buildings per level, each increase in level only corresponds to their maximum energy increasing by one. What then happens is that as players level up and build more buildings, the amount of energy left over once they've collected from their buildings at each level *decreases*. This means that the amount of energy available to take any real, meaningful, action in the game diminishes as time goes on. Players can then either accept taking few actions, quit the game, or pay money for more action as they progress. They're steadily forced into a Dark Pattern: Pay to Skip pattern

that they were not aware of at the beginning. This surprise use of a dark pattern is particularly bad, as it intentionally misleads players to thinking they will have a different game experience.

This pattern appears as a dark pattern in games as the game design is usually modified by its inclusion, and players' expectations about what they thought they were involved with are violated. It does not appear in general applications, as the features users are paying for are usually *disclosed* before sale (albeit in order to upsell more expensive purchases). For example, *Evernote* includes a DARK PATTERN: PAY TO SKIP pattern, by placing Premium users at the front of the optical character recognition queue to make text on images searchable. However, this is a touted feature of the Premium product, and is clearly disclosed on the website, so is not a dark use due to the disclosure principle. It may be the case that this pattern's inclusion in games will quickly become recognized by players too, and hence will lose its darkness.[1]

---

[1]This pattern can exist as a dark pattern in other contexts. For example, the infamous Homeland Security checkpoints at American airports violated honest fliers' expectations that they should be able to get onto a flight with the minimum amount of hassle. In response to the awful experience, the TSA rolled out the "Precheck" program that skips the lines. This program has an application fee of $100 (McCartney 2012). In the same way games have had their integrity violated so returning to normal could be monetized, so too was the experience at the airport.

Chapter 11

Social capital dark patterns

11.1 Overview

Social capital is the "ability of actors to secure benefits by virtue of membership in social networks and other social structures" (Portes 1998). While social capital cannot be *spent* like other resources, people put their benefits at risk if they perform actions which could lead them to lose status or even membership from their social network.[1] Many applications are looking to become social activities, and the dark patterns in this chapter are ones that put a user's social capital in jeopardy. If they stand the risk of losing their social standing, or feel like they're interacting with software out of social obligation to fulfill an *honor* motivation, they're in a social capital dark pattern.

Social capital takes a long time to earn, likely much more so than time or money, so these patterns are particularly upsetting for users. Users who enjoy an application will naturally want to tell others about it, and have them enjoy the app too. Instead of using a social capital dark pattern, think about how to reduce friction for users to BROADCAST how much they enjoy it and share their experiences with it. *Insta-*

---

[1] It would be a particularly odd, yet incredibly enticing, world if social capital was a resource that could be spent and converted: imagine if there was an exchange rate between say, gut-punching your boss, and how many boxes of chocolates you'd need to buy him to square it up again.

gram is famous for its large social network, but it began by simply offering a great way of making visually-pleasing photos; photos which users then shared with their friends.

11.2 Dark Pattern: Social Pyramid Schemes

Description: A requirement for other people to be brought in to the application before it is interesting

Reiss desires: *Social Contact*, *Curiosity*

Reduces Reiss desires: *Independence*, *Status*

Related to: CONTACT LIST, BROADCAST

Examples: *FarmVille*, *SimCity Social*

Use: Broadly, pyramid schemes are investment schemes that offer high returns to investors by using incoming funds from future investors (Walsh 1998). In a loose sense, early investors are made happy by ensnaring new investors (or, more colloquially, "suckers") who are unhappy until they, in turn, ensnare some more people into the scheme. This dark pattern is pronounced in social games, but also applies to social platforms that don't have unidirectional relationships.

Many games encourage players to invite their friends: multiplayer games, in particular, are often more enjoyable with friends than faceless opponents. Usually the benefit to the player is an *enhanced* experience. Social games have begun to utilize social capital more expressly, by placing tangible in-game benefits for inviting new players. This begins to have shades of grey, but is still not necessarily dark yet. The darkness in DARK PATTERN: SOCIAL PYRAMID SCHEMES, much as with DARK PATTERN: INTERACTION BY DEMAND, is not the pattern itself, but when used in conjunction with blocked player progression. The previously mentioned staffing mechanic in '*Ville* style games is a clear DARK PATTERN: SOCIAL PYRAMID SCHEME blocker. As Liszkiewicz (2010) notes:

> My mother began playing *FarmVille* last fall, because her friend asked
> her to join and become her in-game neighbor. In *FarmVille*, neighbors send
> you gifts, help tend your farm, post bonuses to their Facebook pages, and
> allow you to earn larger plots of land. Without at least eight in-game neigh-
> bors, in fact, it is almost impossible to advance in *FarmVille* without spend-
> ing real money. This frustrating reality led my mother—who was now obli-
> gated to play because of her friend—to convince my father, two of her sisters,
> my fiancée and (much to my dismay) myself to join *FarmVille*.

The sting in the tail of this pattern is the entrapment of others who feel socially obliged to play, and who may themselves expand the scheme by inviting others. "Soon, we were all scheduling our days around harvesting, sending each other gifts of trees and elephants, and posting ribbons on our Facebook walls. And we were convincing our own friends to join FarmVille, too" (Liszkiewicz 2010). Social games are particularly aggressive to encourage new neighbors: *SimCity Social* requests players to invite their friends less than 60 seconds from starting the game, and usually will auto-populate their entire friends list to be invited.

While it's easy to see the problem in social games, this pattern existed for a number of years on *Facebook*. *Facebook* was an intentionally closed system that encouraged users to only interact with those they knew, and all other information would be hidden to those who weren't friends. While protecting privacy, it meant that interactions were largely meaningless until many bidirectional friend relationships were made. The application was boring until more people joined.[2] This led to users of *Facebook* harassing their friends to join up too.

Twitter popularized the unidirectional contact, where one could follow others who were publicly BROADCASTING. New users didn't need their friends to join in a DARK PATTERN: SOCIAL PYRAMID SCHEME to find interesting users to follow. In 2009, *Facebook* also introduced unidirectional relationships, which it also termed "following" (Kincaid 2009). With the difficulty of bootstrapping users to new social services, many

---

[2]When startups have to build an expectation of viral growth into their business plans, this is code for the application is boring by itself, and is an instance of a DARK PATTERN: SOCIAL PYRAMID SCHEME.

social startups also utilize a following mechanic, or employ Facebook's API to allow their applications to bring in users' current social graphs (*Spotify* being one such example). The transition of this pattern from social apps, its steady abandonment, and then revival inside social games indicates just how malleable dark patterns can be.

11.3   Dark Pattern: Impersonation

Description: An application creates BROADCASTS that appear to be from a user, when in fact the user hasn't created the BROADCAST at all.

Reiss desires: *Independence*, *Status*, *Tranquility*

Also known as: Friend Spam (Brignull 2012)

Related to: CONTACT LIST, BROADCAST

Examples: *Sim City Social*, *Facebook*

Use: Social networks allow for BROADCASTS which others can see. The expectation is that these BROADCASTS are always initiated by the user they are attached to. In many social games, players see representations of their friends (or other players) in their own games, and are similarly able to BROADCAST actions to others. DARK PATTERN: IMPERSONATION occurs when these BROADCASTS are communicate actions they users never performed, thus misleading others about the activities of their friends.

Sometimes, the BROADCASTS come from external areas, as described by Brignull's DARK PATTERN: FRIEND SPAM: "A site or game asks for your Twitter or email credentials... but then goes on publish content or send out bulk messages using your account - i.e. from you" (Brignull 2012). Sometimes this pattern is enabled by the user implicitly. For example, a UI might obscure a checkbox that provides permission for IMPERSONATION, so the user is not aware she is giving consent.

DARK PATTERN: IMPERSONATION is not simply an inconvenience to users, but something that has landed *Facebook* in hot water. *Facebook* shows sponsored posts

Figure 11.1: A screenshot of a *Facebook Sponsored Post*. This post gives the appearance that the user has just "liked" *Tomb Raider*. In fact, the user may have liked *Tomb Raider* some time ago, but *Facebook* has surfaced that fact now in order to show an advertisement. This is a form of DARK PATTERN: IMPERSONATION. This could be reworded to say "has liked" in order to indicate that the action is not active. Avatar and last name obscured for privacy. Taken on March 4, 2013.

to users, and uses their friends' likes as a means of deciding what posts to show, as shown in Fig. 11.1. However, while there is the indication that these posts have been actively Broadcasted, the "likes" may have happened some time ago and *Facebook* has simply chosen to surface them now. While this seems fairly innocuous, it can become quite disturbing. In one example, *Facebook* showed a sponsored post that was liked from someone who had died, but not yet had their death relayed to *Facebook* (Meisler 2012).[3] The Dark Pattern: Impersonation was so disliked that a class action lawsuit was filed and settled by *Facebook* for $20 million over the feature. All *Facebook* users who may have been featured received an email notifying them of the settlement, which began with the following:

> NOTICE OF PENDING CLASS ACTION AND NOTICE OF PROPOSED SETTLEMENT ANGEL FRALEY V. FACEBOOK, INC.
>
> You are receiving this e-mail because you may have been featured in a "Sponsored Story" on Facebook prior to December 3, 2012.
>
> A federal court authorized this Notice. This is not a solicitation from a lawyer.
>
> **Why did I get this notice?** This Notice relates to a proposed settlement ("Settlement") of a class action lawsuit ("Action") filed against Facebook relating to a particular Facebook feature called "Sponsored Stories." According to available records, you may be a "Class Member."
>
> **What is the Action about?** The Action claims that Facebook unlawfully used the names, profile pictures, photographs, likenesses, and identities of Facebook users in the United States to advertise or sell products and services through Sponsored Stories without obtaining those users' consent. Facebook denies any wrongdoing and any liability whatsoever. No court or other entity has made any judgment or other determination of any liability.

Users involved in the class action lawsuit could either choose to receive $10, or have it donated to charity, and *Facebook* agreed to update their Terms of Service and settings surrounding the feature.

Even though *Facebook* has lost $20 million over Dark Pattern: Imperson-

---

[3]This article even accuses other companies of somehow leveraging *Facebook* to get them to 'Like' brands they would never be associated with, such as someone with an anti-corporate stance liking Discover credit cards. As these accusations aren't corroborated, they are not commented on here.

ATION, they are far from the worst offender. *SimCity Social* uses the DARK PATTERN: IMPERSONATION pattern so mercilessly, it is worthy of its own particular analysis, which you can see through Figs. 11.2 to 11.6.

Figure 11.2: After the player levels up, this window will popup, asking players to "Take a second to enhance your gaming experience and earn these free resources!" Clicking "Let's Go!" pops up a small Facebook window, which asks for *SimCity Social* access to post to the wall on your behalf. Once permission is given, *SimCity Social* can post on a player's wall without popping up any other notifications. This is important to EA, as the game attempts to post to the wall often. Players can't opt-out. Note how this window can't be cancelled, the Facebook window *must be seen*. If the player denies access, this popup reappears once she levels up again. It cannot be hidden forever.

Figure 11.3: After completing a task, *SimCity Social* presents a window like this, with the giant "Share with Friends" button at the bottom. If you've given permission for *SimCity Social* to autoshare, this button is a single-press with no confirmation required. Astute users may notice the small 'X' in the top-right corner, which will close this window without performing the share action.

Figure 11.4: This list appears without any particular rhyme or reason. It populates the list with names of your friends, but you don't know which friends they are (who is "John"?). These friends don't have to be playing the game, and will instead be invited to the game instead of just getting the gift. This window is deliberately ambiguous and vague about what it will do.
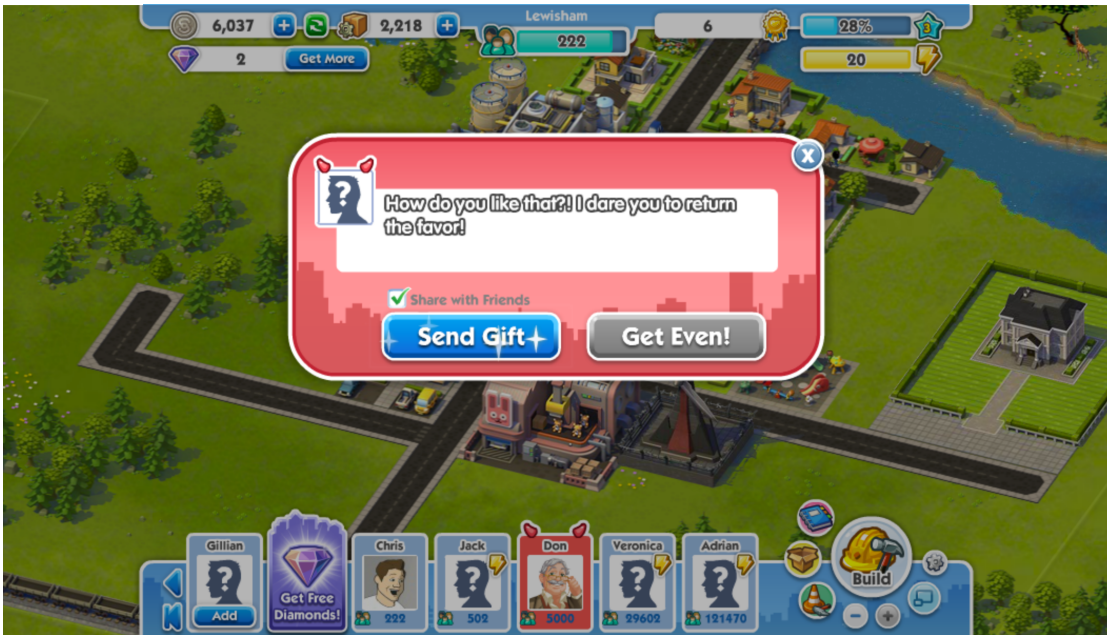
Figure 11.5: As part of the tutorial, *SimCity Social* will pick a friend, and impersonate that friend giving the player a gift (in this case, it's a mean gift). Social games often have friends (who are playing the game) come and visit to perform actions. In this instance, the action is fabricated.

Figure 11.6: In *SimCity Social*, friends can move into buildings. This mechanic seems to be intentionally analogous to the staffing mechanic of other games. In this instance, not only did the player not perform the action to move into the house, the player is not, and has never, played the game. This gives the impression that the player is playing, and can be sent requests without possible loss of social capital. Avatars obscured for privacy.

Chapter 12

Patterns as analysis

12.1 Overview

One use for patterns is as a tool to analyze other applications. Rather than simply pointing at various parts of an application and saying "This looks like it could be useful," patterns provide a language and understanding of an application as a whole by analyzing its constituent parts. Looking at the overall pattern mix describes what desires are being met by the product, and just as importantly, which desires are *not*. Knowing where the motivational gaps are can help improve the design, or identify needs that could be addressed with a new product.[1]

This chapter analyzes two very different products using motivational design patterns. *Khan Academy* is an educational website that attempts to motivate users through quick feedback loops. *Tiny Tower* is a small mobile game that motivates users through delayed feedback, requiring players to return again and again to see the fruits of their labors. Such radically different applications were chosen for two reasons. The first is to illustrate the wide spectrum of motivational software, and prove that the patterns apply equally. Secondly, choosing such different software shows how much variation can

---

[1]This new product could either be complementary or competitive.

be created by using different pattern mixes.[2]

Applications are analyzed in three stages. The first stage is to describe the application, what it does and which audience it caters to. This provides the background for the second step, where the patterns used are identified. Ascertaining the overall pattern mix can indicate the motivational goals of the application. Are they picking gameful patterns for quick feedback? Social patterns to leverage network effects? Interface patterns to make the user feel at ease? Finally, three patterns are chosen for their educational value: the patterns that best highlight the successes or failures of the product.

12.2   Case study: *Khan Academy*

12.2.1   Application description

Khan Academy is a website dedicated to educating school age students, with lessons ranging from simple addition to macroeconomics. Lessons are taught via videos, often created by the founder, Salman Khan. Some of the mathematics lessons allow students to engage in a series of question and answer sessions to test their knowledge. Students are encouraged to continue at their own pace, and educators can register as "coaches" to monitor students' progress.

Students can ask each other questions underneath the videos, although answers do not appear to be vetted.

Khan Academy is an excellent example for study. The application aims right at the heart of education, which many students in the classroom have little intrinsic motivation for.

---

[2]Alexander (1977) called these pattern mixes 'languages.' 'Mixes' is used here to make it clearer that different pattern subsets create different outcomes, and to avoid conflating small 'languages' with the pattern language as a whole.
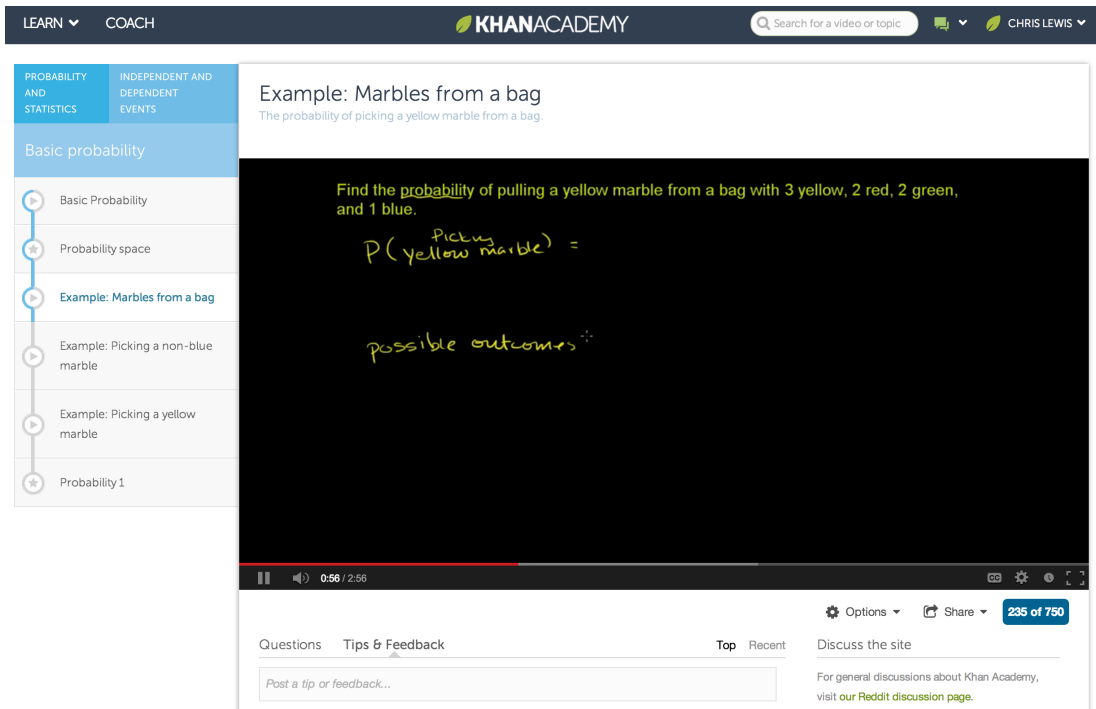
Figure 12.1: A screenshot from *Khan Academy*, showing the interface when watching a video. On the left hand side a TASK QUEUE is used to show how students move through the "Basic probability" lessons. At the top-right is a SEARCH box. At the bottom center is an area to post questions or provide tips, which is a BROADCAST mechanism. At the bottom-right is the ability to BROADCAST the video via the share dropdown, and a SCORE is provided based on how long the video has been watched for. Below that is a link to a META-AREA where users can discuss *Khan Academy* on *Reddit*. Taken on April 10, 2013.

12.2.2 Patterns used

Khan Academy makes heavy use of patterns that provide clear feedback, such as SCORE, BADGES and TASK QUEUES. This is unsurprising: the clear and quick feedback that these patterns offer tighten the traditionally elongated feedback loop in schools. Instead of doing work, handing it in to the teacher, waiting for assessment, and getting it back again, students receive confirmation of their answers immediately, and their progress is reflected in these patterns. Areas where they excel will have them moved to harder questions, whereas areas where they struggle will continue offering questions at the same difficulty until they get them right.

However, the gameful patterns offer the high possibility of unintended behavior being incentivized, and often utilize extrinsic motivations. Careful attention will be given to exactly how the patterns are used, and if there is evidence of controlling feedback that could undermine intrinsic motivation.

Khan Academy uses the following patterns:

Score Energy points are offered for watching videos and earning badges.

Badges Badges are handed out for specific actions.

Social Feedback Questions can be voted up or down, assuming the user has enough energy points.

Increased Responsibility Only users with 5000 energy points or more can vote up or down comments.

Praise The interface often congratulates students with phrases like "Nice job!"

State Preservation Progress is automatically saved for logged in users, and users can leave the site and come back at any time.

Reporting Comments can be flagged for review if they are spammy or unhelpful.

Search Videos can be searched for via a search box on the front page.

Task Queue Task queues appear as tutorials to the interface, as videos and Q&A sessions to be watched as part of a certain subject, as progress bars indicating progress through a particular Q&A session, and creation of goals with multiple parts.

Meta-Area *Khan Academy* runs a *Google Code* issue tracker where users can report problems or make feature requests. Users can also discuss wider issues at the *Khan Academy* community on *Reddit*.

Broadcast Students can post questions under videos, and reply with answers and tips. Particular lessons and badges can be shared to *Facebook*, *Twitter* or emailed.

Activity Stream Latest contributions to discussion, and badges earned as part of discussion, are shown as a stream on a student's profile page.

Identity Shaping Students can choose their avatar (new ones are unlocked once a student reaches certain numbers of energy points), display computer programs they have made, and highlight five particular badges.

Dark Pattern: Grind The large number of TASK QUEUES, and the use of energy points to unlock avatars, could place students in a grind.

12.2.3 In detail

12.2.3.1 Task Queue

Of all the patterns used by *Khan Academy*, the TASK QUEUE is the most prolific. Figs. 12.2 to 12.5 shows all the instances of a TASK QUEUE in its various instantiations. In the absence of a naming scheme for the lists of videos and questions from Khan Academy, these will be referred to as "playlists." The TASK QUEUES always

pertain to tracking either entire playlists, or particular videos and questions within each playlist. Fig. 12.2 is a playlist, showing all the videos and question sessions within. Fig. 12.3 is a view across the whole site, showing progress in all playlists. Fig. 12.4 and Fig. 12.5 overlap. Both show particular individual videos and questions from various playlists, the difference being that the "Suggested Activity" is a PERSONALIZATION instantiation of a TASK QUEUE, whereas the "Goals" system is a CUSTOMIZATION instantiation.

With the exception of the skill progress queue, each task can be completed in around 10 minutes or less. This provides an immediacy to the feedback from the queue as students can quickly see that they are progressing against the challenge, creating a feeling of *Power*. The large number of activities and queues allows students to heartily engage in goal consumption. That students can create their own playlists using the goals feature allows for an amount of autonomy.

By and large, there is little to be changed about the TASK QUEUES in *Khan Academy*; they appear often, and their inclusion in this educational context seems very apt. One possible change is to add to Suggested Activities an indication of how many other activities remain in the playlist, such as saying "You've finished 5 out of 7 in Basic Algebra, there's 2 left!" This may trigger both goal-gradient behavior and the Zeigarnik effect to motivate users to "clean up" their remaining activities in the playlist.

12.2.3.2 Score

Khan Academy utilizes a system called "energy points," which provide a SCORE mechanism for the site. "What are energy points?" describes energy points:

> Energy points measure effort on Khan Academy. Learners earn more energy points for pushing the edge of their knowledge. They are not a measure of mastery or ability.
> As of 1/1/2013, videos earn up to 750 points for completion and each problem done earns a base of 15 energy pts per problem; this base slowly decreases down to 5 energy points (but no fewer) per problem if a learner
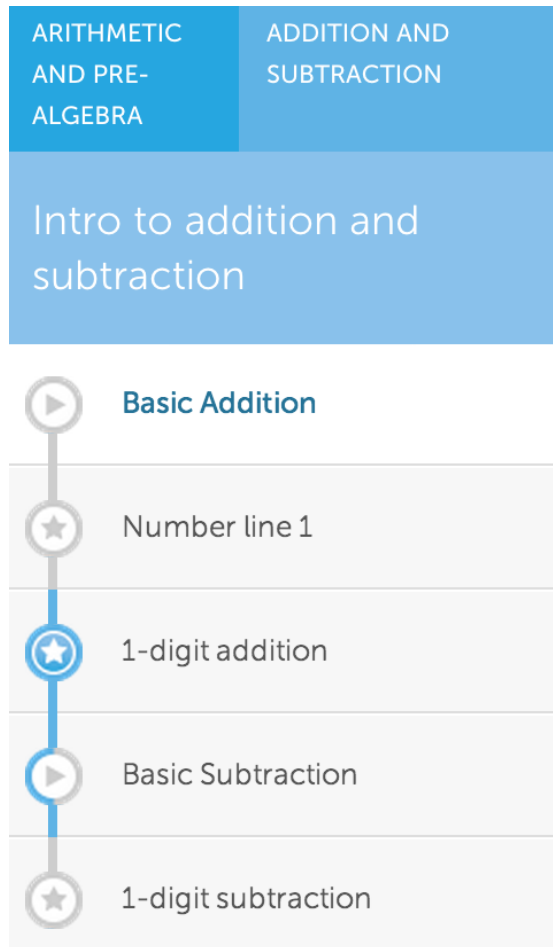
Figure 12.2: A screenshot of a playlist TASK QUEUE from *Khan Academy*. This shows a list of five activities within the playlist, the two activities with the play symbol are videos, the three activities with the stars are question and answer sessions. The full circle around the star next to "1-digit addition" shows that the session was completed. The half-circle around "Basic Subtraction" shows that the video has only been partially watched. Taken on April 10, 2013.

| Started | Proficient | Review | Struggling |

| Add. 1 | # line | Represnt #s | Add. 2 | Time 0.5 | Sub. 1 | Tell time | # line 2 | Sub. 2 | Mult. 0.5 |
| # line 3 | Tell time 2 | Sub. 3 | Add. 3 | Mult. 1 | Order Neg # | Arith WP1 | Add. 4 | Sub. 4 | Div. 0.5 |

Figure 12.3: A screenshot of the Skill Progress Task Queue from *Khan Academy*. This shows how the student is doing across the entire site, marking playlists as either being "Started," "Proficient," "Review," or "Struggling." Areas convert from "Proficient" to "Review" over time, requiring the student to revisit the area if they wish the "Proficient" label to return. Taken on April 10, 2013.

> has a long streak of correct answers in an exercise (demonstrating impressive proficiency) and continues working on it. If a learner is working in power mode (context-switching mode) or a recommended exercise or is not yet proficient, the base takes on a multiplier.
>
> Badges are awarded for behaviors—some related to points, but not necessarily (e.g., proficiencies on exercises, speed, or other behaviors like building community or engaging with computer science).

One thing that is not mentioned in the text is that energy points are used to unlock extra avatars when students pass certain thresholds. At the time of writing, these thresholds were 10,000 (14 videos needed if no questions are answered), 50,000 (67 videos), 100,000 (134 videos) and 250,000 (334 videos).

This use of Score seems particularly confusing. The rationale given is that *Khan Academy* wants to incentivize effort, and yet points are handed out for zero effort tasks. The inclusion of a Score will naturally motivate some of their students to maximize it, and the problem with Score is that it is always at high risk of being gamed. *Khan Academy* is at particular risk, because many of the answers can be found by simply plugging an equation into *Wolfram Alpha*. One method of maximizing energy points would be to open tabs to as many videos as possible and let them play through. There is no concept of liveness checks with these videos (such as pausing the video and asking the student to answer a relevant question). Another way of getting easy energy

Suggested Activity

Representing numbers in Addition with carrying — Rock out

Number line 1 in Intro to addition and subtraction — Rock out

Telling time 0.5 in Telling time — Rock out

Probability with Playing Cards and Venn D… — Watch on

Basic Probability — Watch on

Addition Rule for Probability — Watch on

Figure 12.4: A screenshot of the Suggested Activity Task Queue from *Khan Academy*. The activities suggested are either questions, represented by the blue icons, or watching videos, represented by the gray video icons. The activities appear to be taken from incomplete playlists, and are suggested by the system. Because they are system suggestions, this is an instance of the Personalization pattern. Taken on April 10, 2013.
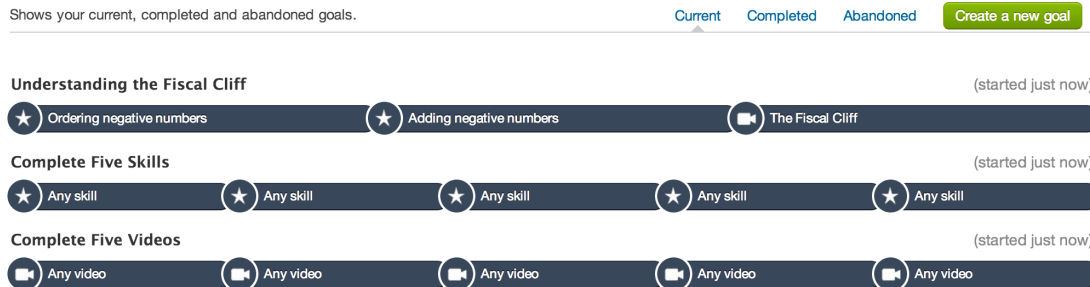
Figure 12.5: A screenshot of the Goals TASK QUEUES from *Khan Academy*. This shows goals that have been created. A goal is a list of videos or activities to watch. Each goal is a TASK QUEUE, and this interface showing all the goals is itself a TASK QUEUE. These goals are created by the user. For example, the "Understanding the Fiscal Cliff" goal includes question sessions on negative numbers, and then a video on the Fiscal Cliff. As these goals are user-created, they are an instance of the CUSTOMIZATION pattern. Taken on April 10, 2013.

Figure 12.6: A screenshot from *Khan Academy*, showing a SCORE provided for watching a video. Taken on April 10, 2013.

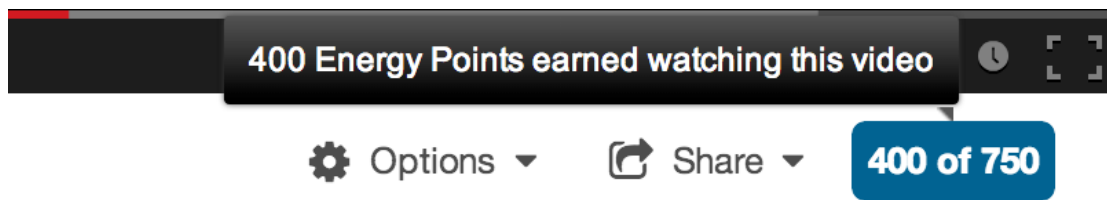points is to head over to lessons such as "Addition and subtraction" and "Telling time" to quickly answer a number of questions. Even with the diminishing returns, answering three trivial questions for 5 points each is quicker than answering one hard question for 15 points.

Another feature of *Khan Academy* is to offer hints for each question, eventually leading to the answer. One user opened an issue at the KHAN ACADEMY Google Code Issues page ("Issue 6994 - khanacademy - a way to cheat"), writing:

> If you have one window in Chrome with all the hints tabbed through (that eventually point to the answer), you can have another window open with the same problem and enter the correct answer....
> This way, you can get through every lesson by simply looking at the correct answer in another window...

Another aspect of SCORE is that it implicitly incentivizes certain actions. Here, the incentives are offered on watching videos and correctly answering questions. This feels a lot like "teaching to the test" where the test comes right after a short lecture. What is not incentivized is learning how to address the problem. There is a reason math teachers usually ask students to show their work; the result isn't necessarily the important aspect, but how the student arrived at the answer. One can imagine this is solvable with more a more complex set up—such as writing in each step of a long division calculation —but this is not present in *Khan Academy.*

SCORE can be used to indicate mastery over a subject, but without the step-by-step implementation, anyone who is being controlled into maximizing their score will circumvent the system. It's also not clear that many students *want* to maximize their *Power* over mathematics or science. *Khan Academy* suffers the same issues with intrinsic motivation as education in the classroom, and while SCORE is a gameful pattern, it doesn't ensure that students will engage with the application in a gameful context.

This pattern should probably be removed entirely. It doesn't offer any intrinsic motivation, and could possibly undermine any intrinsic motivation that was there.

236

Instead, *Khan Academy* would do better at thinking about other motivations that students might have, and tie things together with the liveness of the Web. Many are excited about sports, enjoying the *Physical Exercise*, *Vengeance* and *Power* aspects. That interest could be translated to exercises where students use probability to do bracket picks for March Madness, and see how their picks do in real-time. They could then chat with their friends about how they are doing as part of *Social Contact*. This provides the same *Power* motivation that SCORE looks to provide, but also loops in many other motivational desires.

12.2.3.3 Badges

BADGES appear in the form of "Achievements." Achievements take different categories, moving from meteorite to Moon, Earth, Sun and ending at black hole badges. Some badges confer energy points to the user. Like *Stack Overflow*, some badges are used to show expected behavior, such as "Flag Duty: Flag your first discussion post on a video or program for a Guardian's attention" and "Tinkerer: Pause a tutorial and tinker with the code." Others reward certain proficiency, such as "Ridiculous Streak: Correctly answer 80 problems in a row in a single skill," while others reward perseverance, like "Redwood: Remain a member of the *Khan Academy* for 2 years."

One particular implementation detail of note is that the majority of badges a student collects are not shown publicly; they are visible only to themselves and their coaches. Badges that students collect for performing actions on discussions do publicly appear as an ACTIVITY STREAM on their "Discussion" summary pages, but their academic progress is never revealed. Up to five badges that students are proud of can be optionally displayed on their public page. This limited number may serve to dampen the motivation to collect badges for *Status* alone. Certainly, one would want to pick the badges that shows one in their best light, but this means there is not necessarily a big drive for status-oriented students to collect all of them. Rather, they'll attempt

Figure 12.7: A screenshot of Sun BADGES from *Khan Academy*. The blue boxes in the upper-right of some badges means that earning this badge rewards the user with extra energy points. Taken on April 10, 2013.

to gain the badges that are most meaningful to them and show the most *Power* over a subject. This is a good mechanism to focus students, rather than engage them in broad, unfocused actions.

As with SCORE, BADGES incentivize certain behavior. This is why they exist: they provide extrinsic reward for performing a behavior that the developer wishes to take place. However, many of the BADGES seem poorly designed, either resulting in unintentional consequences or rewarding unhelpful learning aims.

Unintentional consequences badges include those such as:

Flag Duty Flag your first discussion post on a video or program for a Guardian's attention.

Thumbs Up Cast your first up vote for a helpful discussion post or program.

Thumbs Down Cast your first down vote for an unhelpful discussion post or program.

Bibliographer Reference a timestamp when answering a question on a video or program.

These badges all offer reward for performing actions, but don't require those actions to be beneficial. Randomly flagging or voting posts would acquire the first three badges, and a quick post of "lol khan sux @ writin @2:14" would garner the fourth. This is highly unlikely to be what was intended when these badges were written, but the actions do correctly meet the specifications.

There are badges that reward watching certain hours of video or earning certain numbers of energy points, but how those can be gamed has already been described in relation to SCORE.

Unhelpful learning aim badges include:

Picking Up Steam Quickly & correctly answer 5 skill problems in a row (time limit depends on skill difficulty).

Double Power Hour Correctly answer 180 problems and watch 30 minutes of video in 2 hours.[3]

Great Streak Correctly answer 40 problems in a row in a single skill.

The problem with these badges is that they don't support *learning*. "Picking Up Steam" and "Double Power Hour" rewards students for finishing many questions quickly, but this is not a metric by which students should be ideally be judged. Their accuracy and attention to detail may well be better metrics, but "Great Streak" punishes mistakes. Better badges of this nature are "Steadfastness: Answer more than 40 problems mostly correctly in a skill before becoming proficient." This neither places a time constraint nor a perfection constraint, and seems far more suited to rewarding good work without inducing anxiety. Another good badge is "Persistence: Answer a problem correctly after having some trouble with many problems and sticking with the skill." While one can imagine easily gaming this one by deliberately flunking answers, when it is offered in earnest to a struggling student, it provides something to be proud of and to provide feedback that the student really is gaining *Power* over the subject.

Perhaps the worst badge is "Make It Rain: Support Khan Academy with a donation of any size," which just seems very *gauche*.

It would be difficult to implement unexpected, endogenous rewards as recommended by Hecker (2010), and *Khan Academy* already uses absolute measures for its rankings. Thus, only small changes are required to improve the BADGE system. Firstly, instructional badges that can be easily gamed negatively should be removed. Secondly, a review should be undertaken to ensure that badges meet modern pedagogical goals.

---

[3]Assuming the tasks are performed separately, this means a user would need to correctly answer a question every 30 seconds.

12.2.4 Conclusion

Khan Academy uses a number of patterns designed to improve feedback, and for good reason. They offer students fine-grained feedback on their progress, feeding their desires for *Curiosity* and *Power* over the subject, and *Acceptance* that what they are doing is right. However, there are some flaws in the execution of these patterns that require resolution, particularly in the case of the SCORE pattern and the BADGES pattern. From a wider view, expanding *Khan Academy* to become more of a learning *community*—rather than a portal that simply contains educational content with a thin social veneer— would better meet needs for *Social Contact*, and may help keep students engaged over a longer term. In particular, providing IDENTIFIABLE COMMUNITIES who may share and nurture a given student's particular interests, may well prove beneficial. Another option would be to appeal to *Curiosity*, and more heavily leverage interaction with the wider concepts by providing INTRIGUING BRANCHES. This could be achieved by incorporating Wikipedia entries onto the site, rather than providing links leading away from *Khan Academy* that could cause students to be lost in distraction.

Khan Academy has already achieved much in providing new learning experiences and motivational frameworks for students, and continued growth seems both assured and well-deserved.

12.3 Case study: *Tiny Tower*

12.3.1 Application description

Tiny Tower is a game for *iOS* and *Android* released in 2011. The game tasks users with steadily growing a city tower, adding one floor at a time. The building is filled with virtual characters who live in apartments, who go to work in stores on other levels. While there is no stated goal, the implicit aim is to build as high a tower as

possible. The game is popular; winning Apple's iPhone Game of the Year award (Curtis 2011), and garnered a score of 82/100 on Metacritic ("Tiny Tower").[4]

Tiny Tower is a strong candidate for analysis versus other highly rated games[5] because there is little in the way of *gameplay*, allowing focus on the motivational patterns used. The game design is very simple: money is required to build new floors, money is gained by selling items in stores, stores sell items by being stocked and having workers, money for stock and workers for stores are provided by residential levels. The gameplay cycle is collect money, restock stores, then add new levels when possible. Because the game design is so barebones, it's addictiveness needs to be present at the application design level, rather than the mechanics it offers. *Tiny Tower* offers the ability to study the use of motivational design patterns in a game in isolation from the gameplay.

12.3.2 Patterns used

Tiny Tower uses a mix of patterns from across the various categories, leveraging gameful, social and interface patterns. The combination is designed to encourage small interactions over a long period of time. NOTIFICATIONS, social BROADCASTS and GROWTH encourage reengagement with the application, but once there, there are often few tasks to perform before the game locks the player out again; forcing her to wait for money or restocking before progress can continue. Systems that block progress in this way can often frustrate, but don't elevate to the level of a dark pattern. Many gamers find this bite-size gameplay a strength, allowing them to avoid spending too much time in the game.

This pattern mixture is fairly common for games that aim for small interactions over long periods; similar mixes can be found in many social games on platforms like *Facebook*, as well as in many other mobile games.

---

[4]The game also gained infamy after accusing Zynga of cloning the game to create *Dream Heights* (Hodapp 2012).

[5]Such as *Punch Quest*, *Ridiculous Fishing* or *Might & Magic: Clash of Heroes*.

Figure 12.8: A screenshot of *Tiny Tower* on *Android*. This image shows the zoomed out view of the tower. Taken on April 11, 2013.

Figure 12.9: A screenshot of *Tiny Tower* on *Android*. This image shows the zoomed in view of the tower. Construction of level 5 takes two hours, and the photo studio and tea house requires around 30 minutes to be restocked. As the tea house has no stock, it is closed and not making money. Taken on April 11, 2013.

Notifications *Tiny Tower* uses local notifications to indicate when a store needs re-stocking.

Growth The tower grows over time as the player builds (but she must wait to see the floor completed), and money is collected when the player is outside of the application.

Task Queue A "Missions" system provides a list of items to stock, providing bonus "Bux" if they are completed.

State Preservation The tower is always saved when the application is closed, and there is no concept of having multiple saves or different towers.

Badges An achievements system offers badges for things such as finishing 13 floors, or fully stocking different combinations of stores.[6]

Contact List A list of towers that the player's friends own lets her see how they are doing. As *Tiny Tower* has no explicit ranking metric, this doesn't function as a LEADERBOARD, but if a player engages with the CONTACT LIST in a gameful way, this feature could be perceived as a LEADERBOARD. Users that perceive the CONTACT LIST in this way are offered the fulfillment of their *Vengeance* desires. However, users that have a lower *Vengeance* requirement are not forced to perceive the CONTACT LIST as a LEADERBOARD. This allows the support of both non-competitive and competitive players, who can choose whether to have a more relaxed or more frantic engagement pattern with the game. This support for players to *Independently* choose how to engage with competitive play is smart, and provides a good template for others to follow.

Score (implicit) There is no explicit goal in the game, so there is no explicit SCORE present. However, players could view money, tower height, or number of workers

---

[6]Particularly amusing is "Bar Fight: Fully stock a pub and martial arts studio."

as an implicit SCORE.

Broadcast A "Sharetower" feature lets users tweet a link to view their tower.

Dark Pattern: Currency Confusion Bux are a premium currency that is used in the game to buy upgrades to the elevator and rush building or restocking. Bux can be earned in game by completing missions or performing a "Where's Waldo?" style task to try and find a worker based on a picture. Bux can be purchased with real money, but in-game currency cannot; an intermediate conversion from Bux into the in-game currency must be performed.

Dark Pattern: Interaction by Demand The *Tiny Tower* restock NOTIFICATIONS usually appear about 30 minutes after playing. This short timeline means that the game is constantly nagging the user to return to it.

12.3.3   In detail

12.3.3.1   Notifications

As mentioned previously, *Tiny Tower* uses NOTIFICATIONS on the host device to let the user know when a store is ready to be restocked. Restocking happens on cycles of about 30 minutes. Players must first choose what to stock from a series of three items, then initiate a restocking order, which is what takes time. Once the order is ready, players must then expressly press the stock button to restock the store. The store will not restock itself after the order has been put in. This design decision exists only to force players back into the game; the choice about what to stock was already made at the ordering stage. All players do is press the stock button, and the game continues. There is no reason why the game could not automatically restock.

While this mechanic exists only to reengage the user, the NOTIFICATION that is used meets the requirements for being a use of DARK PATTERN: INTERACTION

246

by Demand. The notification avoids the use of controlling language that would instantiate this dark pattern immediately. The particular wording is "The [store] on floor [number] is ready to be restocked!" This message is *informational*, and does not pose a risk to undermining a sense of autonomy. However, the short cycle on these notifications—occurring about 30 minutes after leaving the application and remaining in the notifications bar until dismissed—does create a feeling of reduced *Independence*, as Tiny Tower gives the impression that it requires babysitting. The constant need for attention, rather than the message wording, is what creates the use of Dark Pattern: Interaction by Demand. This grows to define the relationship the player has with the game: constant nagging creates an almost combative environment, where players have to fight *Tiny Tower* to stop intruding into their lives.

The strange thing about the use of this pattern is that it seems unnecessary, particularly given the ease with which it is disabled by utilizing the host platform's notification controls. While it is undoubtedly powerful at getting the user to return, it hinges on a mechanic that is fundamentally unnecessary and unexciting. It's likely that this is a case where the game design was compromised in order for the payment model to be a success; encouraging users to spend Bux to hurry restocking to avoid the drudgery. However, there are many other areas in the game where Bux can be spent, such as hiring new workers into apartments or building a specific floor (like specifying a sushi house, rather than a random restaurant). The restocking mechanic seems superfluous and unnecessary.

A better implementation would be to allow the player to queue up stocking (allowing the game to run independently for hours at a time), and notify the player when she has enough money to build a new level. The excitement around the Growth of the tower would easily draw users back into the game, and provide a more meaningful and fulfilling experience when they return.

12.3.3.2 Growth

The strongest motivational draw for *Tiny Tower* is its use of Growth. The appeal of *Tiny Tower* is to grow the tower as high as possible, and watch all the workers within scurry about their business.

The Growth pattern is instantiated by the use of all three required elements: liveliness, delayed time effects and non-determinism. The tower continues to sell items and take rent from workers when the player has left the app, providing liveliness. When the user builds a floor, the impact of the floor is not immediately seen; it takes time to construct. The identity of the floor is specified as one of six different categories: food, service, recreation, retail, creative or apartments. Within each category are various floor types (such as sushi, burgers, pub for the food category). If the player wants to build a specific floor type, she must use Bux to pay for it.

The Growth pattern in *Tiny Tower* is used well. Firstly, being able to zoom-out and gauge how high the tower is helps players feel their *Power* to construct something important. The buildings in the background are quite short (see Fig. 12.9), so players immediately feel like their tower is a dominating construction in the town. Secondly, timers are found throughout the user interface, measuring how long it will take a floor to build or restock, setting up reinforcement schedules that prime the user to return when they are over. Finally, users have the ability to cosmetically alter the tower, moving around the floors, or altering workers' clothes and floor paint schemes, creating *Order* and *Power* over how the tower looks.

The only thing missing from *Tiny Tower*'s use of Growth is that it doesn't tap into users' *Curiosity*. Nothing of real interest happens when the user is gone; money flows in and stores run out of stock, but that is the extent of it. One could imagine implementing various timed events that the player could check-in for.[7] For example,

---

[7] *Animal Crossing* used a mechanic like this often. For example, KK Slider would play a music set on Saturday nights.

workers could hold parties on random weekend nights, and if the player logs in while there's a party on, she could stock the pub with expensive beer that will sell out more quickly. Not only is her *Curiosity* rewarded with the visually pleasing aspect of seeing all the workers doing something out of the ordinary, but a gameplay bonus is conferred as well. Keeping the game fresh, so that users feel they are constantly curious about what's happening in the tower, would provide a strong draw, and, again negate the need for the use of the DARK PATTERN: INTERACTION BY DEMAND.

12.3.3.3 Score

Tiny Tower offers no explicit SCORE. There is no provided metric which indicates success, and no goal is announced to the player at the beginning of the game. However, three numerical values, workers, money and building height, are all on-screen at the same time. It seems reasonable to think that most players would consider the height of their tower to be their score, and thus the tower height is an implicit use of SCORE, providing feelings of *Power* over the game.

As *Tiny Tower* only offers implicit SCORES, and does not rate players in LEADERBOARDS, a more relaxed atmosphere is cultivated, where the struggle to increase the size of the tower feels to be part of an *Independent* choice, rather than something forced upon the player. This independence of choice is a trait that a number of GROWTH games have, and the use of it here underlines its importance to creating the gameplay experience it does.

12.3.4 Conclusion

Tiny Tower is a game that is equally easy to resent as it is to love. The controlling manner that it interacts with players via DARK PATTERN: INTERACTION BY DEMAND seems both calculated and lazy at the same time. This is a shame, as the core game provides a relaxing space to grow something that feels important and unique.

Removal of this pattern, and replacing it with more focus on the GROWTH aspects of the game, would greatly improve the experience.

The first lesson of *Tiny Tower* is that it shows that adding dark patterns, even just one, can hamper the application significantly. As mentioned before, what is so puzzling about it is that the pattern isn't necessary, and the goals the designers looked to achieve through its inclusion are better met via other means.

The second lesson is that the GROWTH mechanism is very strong at creating engagement. We see GROWTH everywhere, from *Tamagotchi* to *Nintendogs*, *Sim City* to *FarmVille*. *Tiny Tower*, in all its simplicity, exposes just how powerful the pattern can be. The difficulty is finding a way for GROWTH to be meaningfully applied outside of games, and whichever software developer solves it first may well have a new hit product on its hands.

Chapter 13

Patterns as generation

13.1 Overview

In the previous chapter, patterns were used to analyze applications. Now it's time to generate new applications using motivational design patterns. Using patterns in this way not only helps provide the answers to "Why will our audience come back to the application?" but also helps bring such questions to an earlier stage in the design process, avoiding the "If you build it, they will come" design mentality.

Two different problem statements are used for building applications. The first problem is for a Web application where the overall application design is yet to be decided, aside from the key voting interaction. This study shows how to create an application from just a small seed of an idea. The second study revolves around a mobile game where much of the design is fixed by necessity, and motivational design patterns are being overlaid. As with the analysis chapter, these two problems are very different, and illustrate the versatility of the pattern library.

In each case, a problem statement is given, and then patterns are chosen from the library. Annotated mockups are used to show how the patterns could be implemented.

13.2 Case study: Website built around voting

13.2.1 Problem statement

The company has received funding to build a website that revolves around people voting on answers to questions, which it calls *Votester*. The core interaction involves a user posting a question, and then offering up different answers that others can choose. Aside from this, the rest of the site is yet to be decided, and flexibility is offered in how the voting system is framed.

The concern is that just asking or voting on questions is not enough to keep people engaged for long-term periods. Ideally the site should build a community, which is expected to lead to long-term engagement.

13.2.2 Pattern choice

The flexibility in the problem statement gives plenty of areas to work on, but focus should first be given to the voting interaction. Voting just isn't particularly motivating. Voting (outside of political elections and such) is largely driven by *Curiosity*: what do I think, what do others think, and do my feelings match theirs? This may spur some interesting discussion, but otherwise voting feels like it is largely inert.

In instances where the broad picture can be designed, it is useful to start with an self-determination theory (SDT)-based approach. The wider scope of SDT makes brainstorming easier than using the Reiss desires framework. Reiss can be brought in later to drill down in to specific design ideas.

Autonomy Autonomy is largely baked into the system: the only meaningful choices available are what to vote on, and how to vote. The only issue that arises is when a vote doesn't offer the choice that you wish to pick, which indicates the voting system should have some form of Social Feedback that lets users add their own answers to the answer list.

Mastery What does it mean to get *better* at voting? Can you? The idea that your opinion can get better or worse by any objective or even subjective measure is objectionable at best. The only way one could get better at a voting system would be one's ability to guess how others vote. One could imagine a system where the user places her vote, then places another vote on what she thinks will be the most popular answer, but this doesn't seem particularly interesting.

Relatedness Relatedness is the real key to this site. The voting acts as a catalyst for discussion and as a way for people to better understand themselves and others. Such a site has strong parallels to dating sites. Dating sites build up profiles of users by asking questions such as "Regardless of future plans, what's more interesting to you right now? Sex or True Love?" These then feed into a profile about the user, which is then matched against others. One possibility would be to create a dating site without the dating, to try and find others that share specific interests. However, finding specific interest forums is easily achieved by searching for [[specific interest] forum] in any search engine. Another idea would be to use answers to create a profile page that then acts as a personal homepage for users. The page is built up implicitly from their votes on the site, and creates things such as "Chris likes burgers over pizza" or "When asked if the San Francisco Giants will win the World Series this year, Chris said 'Yes.'"

Using the ideas from this brainstorm, it seems wisest to continue pushing on the *Social Contact* and *Curiosity* aspects that voting creates, and trying to maximize the potential from these. Social and information patterns are likely to be of the most use. While gameful patterns may help create engagement, the trickiness in ascribing a value to people's opinions makes them difficult to use without any unintended consequences.

The site will largely resemble *Reddit*, with a home page displaying popular BROADCASTS in the form of questions. Popular questions are deduced via the amount

of answers the question gets, rather than any specific upvote/downvote system. Each question has discussion associated with it, where SOCIAL FEEDBACK is offered. When SOCIAL FEEDBACK is created, NOTIFICATIONS are generated. Questions can be tagged with various meta-data that indicates what the question is about, and users can FILTER based on those tags to find IDENTIFIABLE COMMUNITIES that they are interested in. Tags are a form of ORGANIZATION OF INFORMATION, and provide INTRIGUING BRANCHES for the user.

Users that register go through an on-boarding process that tries to PERSONALIZE the site to their particular interests, showing questions with particular tags. Registered users are allowed to leave comments and REPORT questionable material. They have a profile page where they can engage in IDENTITY SHAPING, showing the answers they wish to have on their profile.

A META-AREA that uses the same voting system as elsewhere is offered for people to share their opinions.

This creates a list of the following patterns:

- BROADCAST

- SOCIAL FEEDBACK

- NOTIFICATIONS

- FILTER

- IDENTIFIABLE COMMUNITIES

- ORGANIZATION OF INFORMATION

- INTRIGUING BRANCHES

- PERSONALIZE

254

- REPORT

- IDENTITY SHAPING

- META-AREA

- NOTIFICATIONS

13.2.3   Annotated mockups

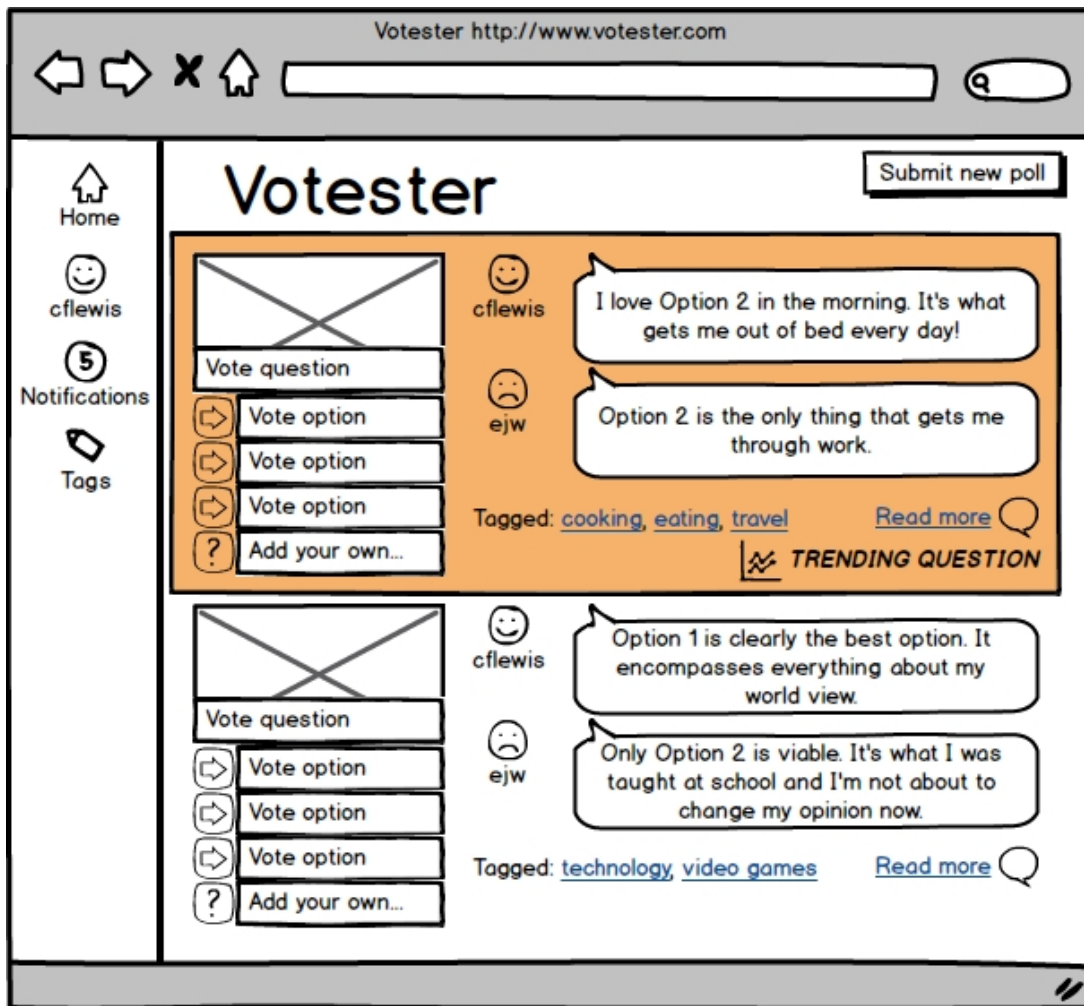Mockups for *Votester* appear on the following pages.

Figure 13.1: The front page, by default, displays the user's favorited tags (which the user has either collected, or added during an on-boarding process during registration), FILTERING unwanted content. In order to prevent filter bubbling, the top question is a popular question outside of the user's tags, which the user may find interesting and add the tag, expanding her bubble. SOCIAL FEEDBACK in the form of comments are presented, showing comments from people who voted differently. *Vengeance* may come into play if the user sees a comment she vehemently disagrees with, and may prompt her to enter the discussion.

The ability to add a new poll is included in a button in the top-right of the page.
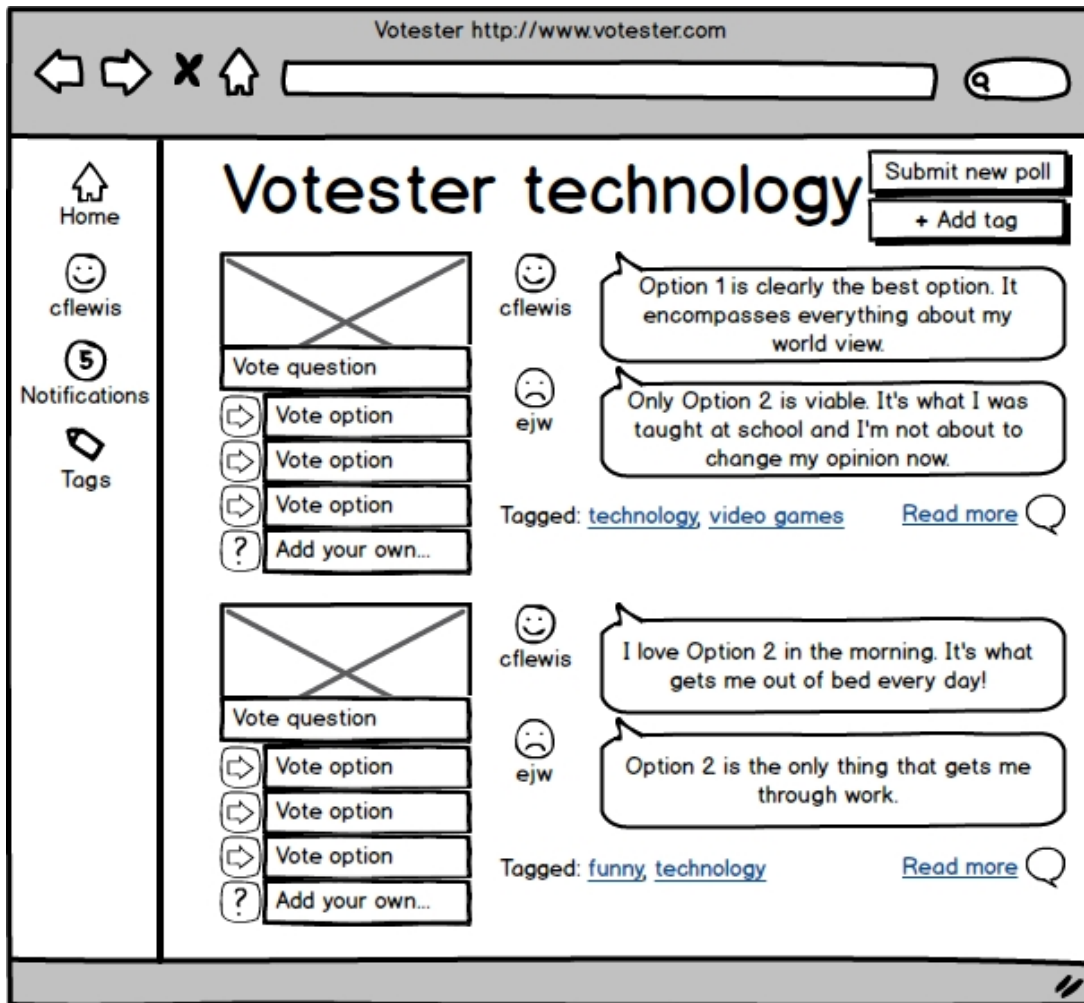
Figure 13.2: When a user clicks on a tag, she is taken to the tag page, which FILTERS all content unrelated to the tag. A tag page is an IDENTIFIABLE COMMUNITY, where people of similar interest congregate. If she enjoys the content here, she can add the tag to her front page.
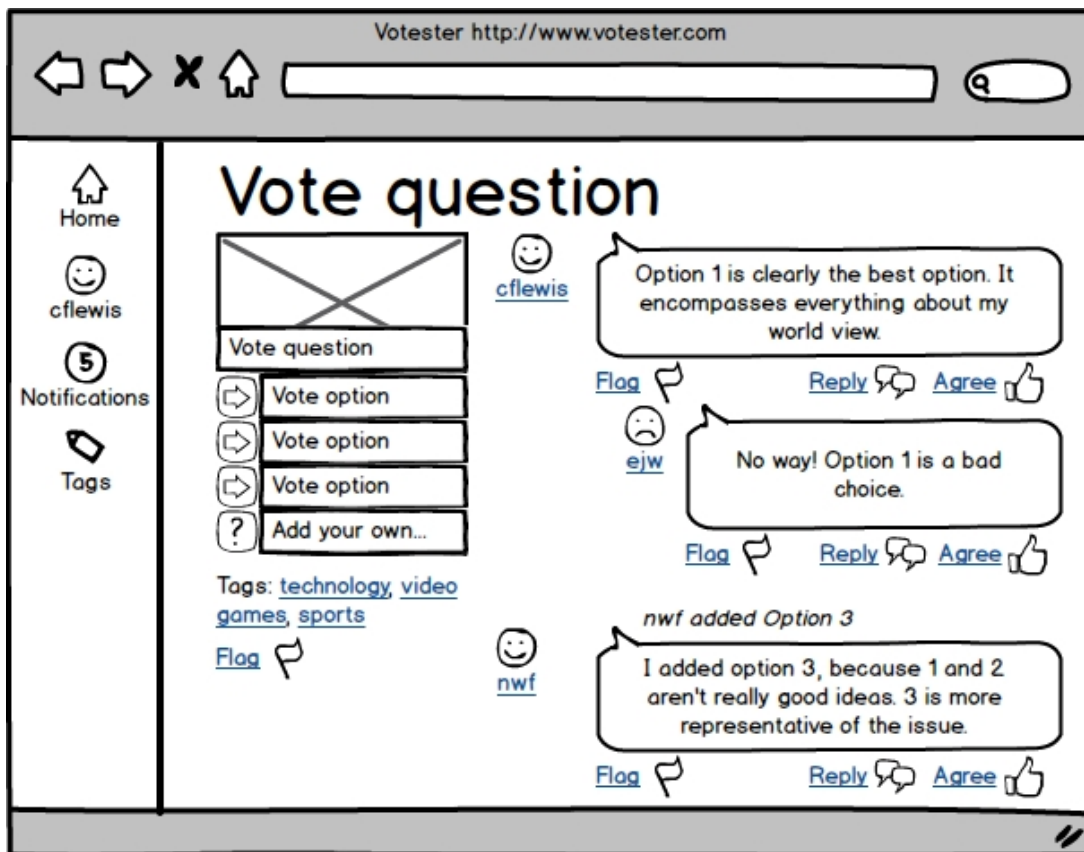
Figure 13.3: Discussions are where the majority of Social Feedback occurs. Users can reply to one another as unstructured feedback, or give structured feedback in the form of an 'Agree' button. If the user is upset by either the poll or a particular comment, she can REPORT it using the 'Flag' button.
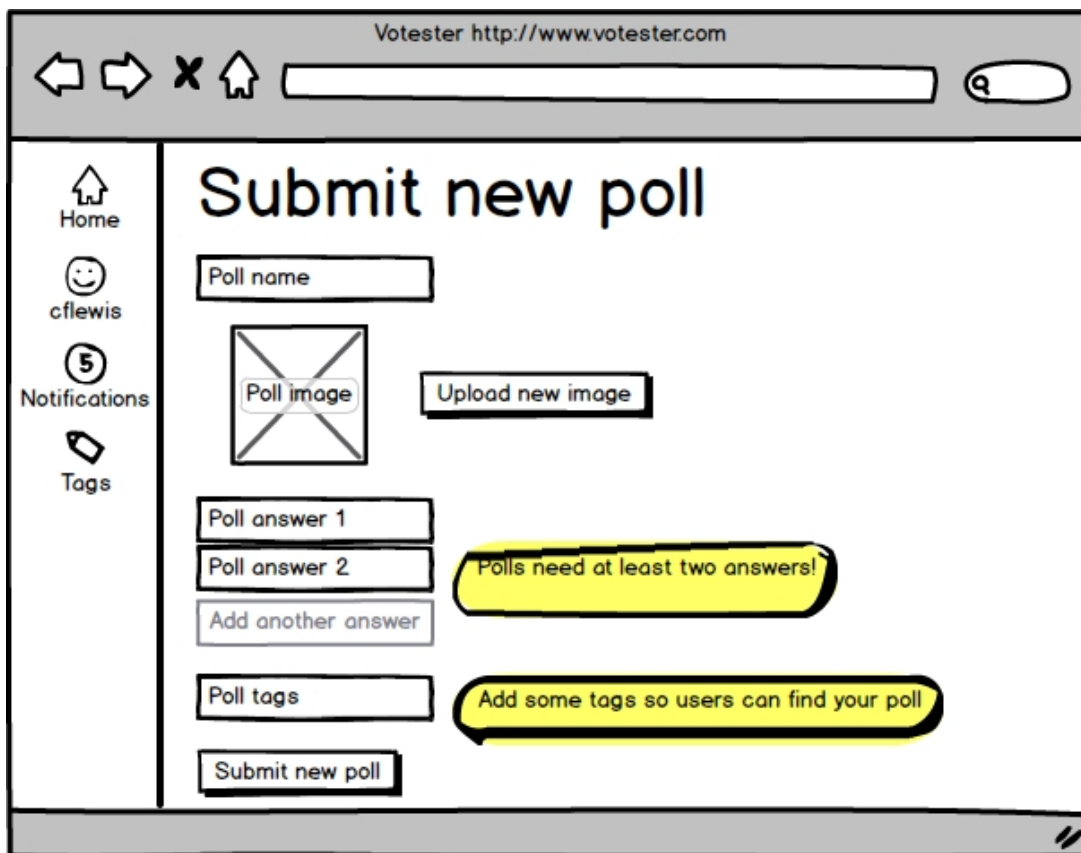
Figure 13.4: This page is where the user can Broadcast a new poll. Images are encouraged to beautify the poll, enhancing the *Romance*.
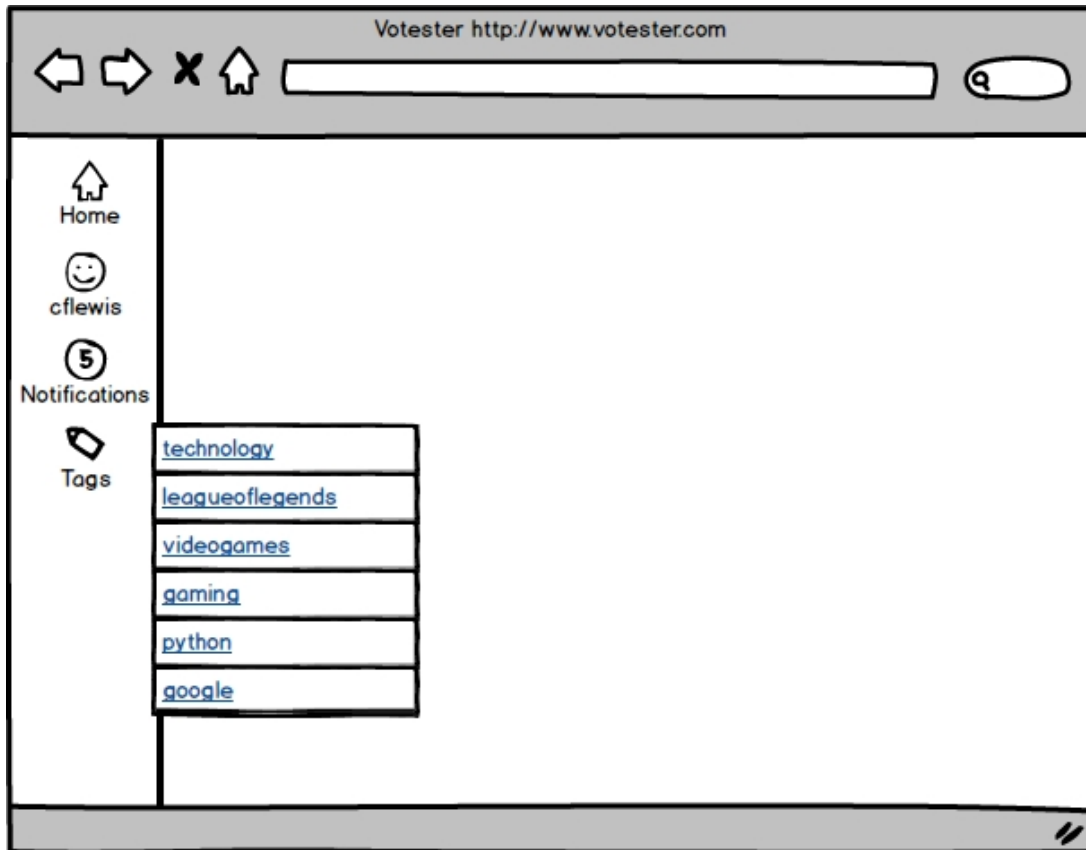
Figure 13.5: Users can quickly move around their various tags using the flyout.
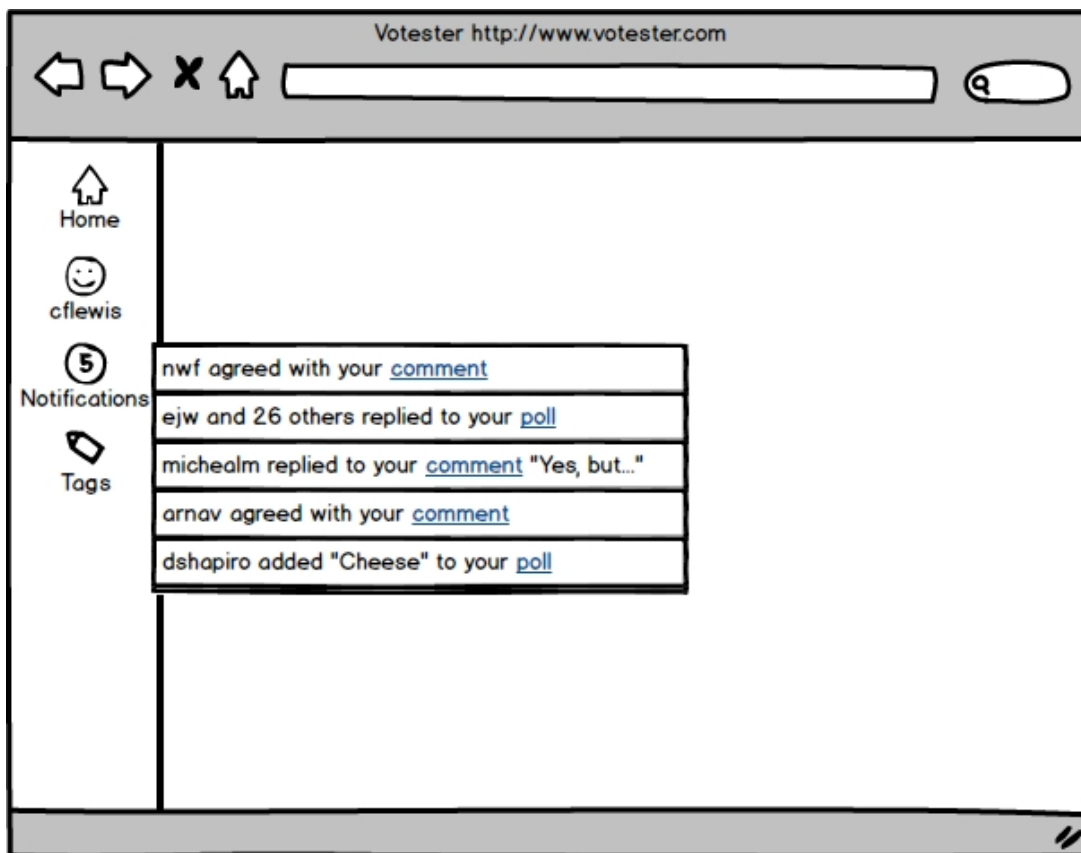
Figure 13.6: NOTIFICATIONS are organized into an ACTIVITY STREAM, which flies out from a button on the left.
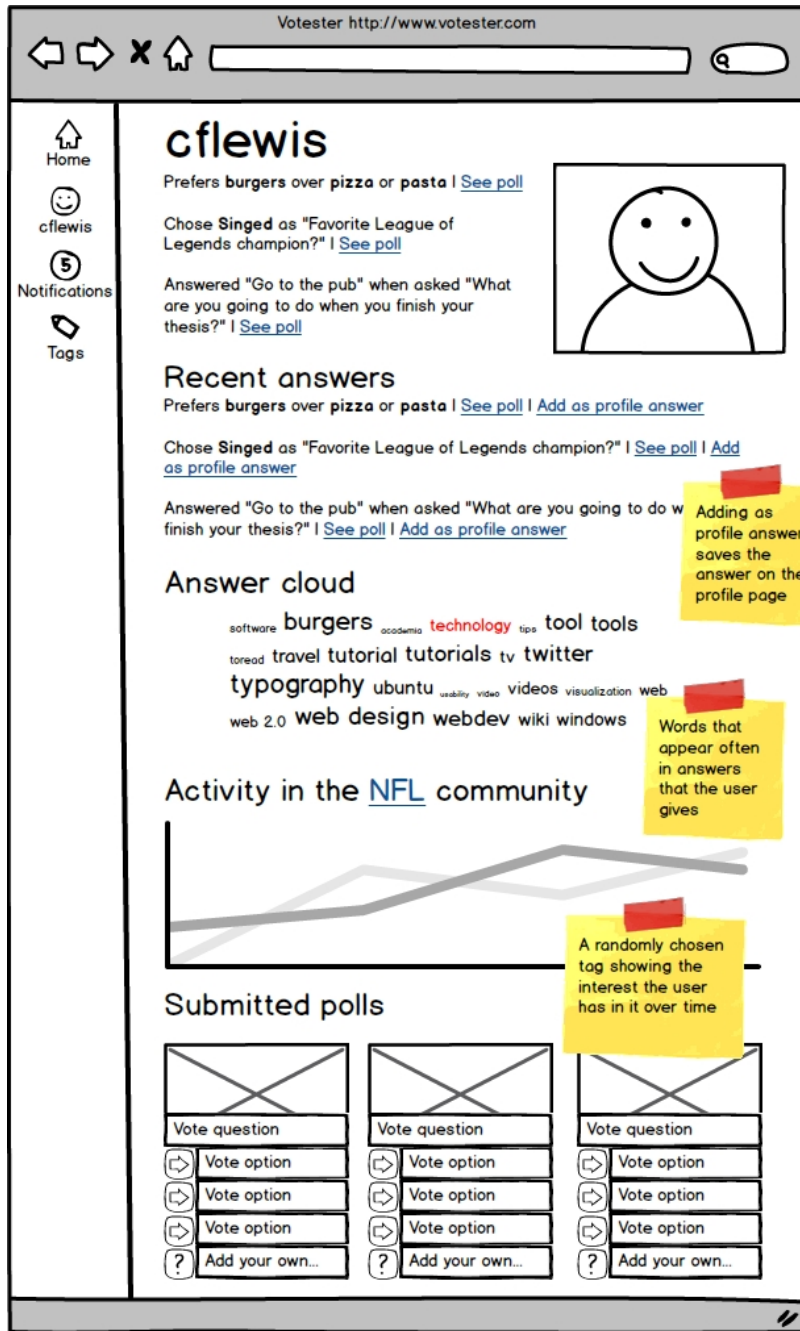
Figure 13.7: Profile pages are where users can engage in IDENTITY SHAPING, choosing a username and avatar. Recent answers are listed here, and the user can add them to a list of interesting answers in the top-left of the profile page. This list cycles through, so the user doesn't need to worry about which of her answers are the most important. An answer cloud and activity chart provides some insight into the user's activity, satisfying her *Curiosity*, as well as others.

13.3 Case study: crowdsourced game for science

13.3.1 Problem statement

The team is building an iPad game, *Xylem*, where players solve puzzles. These puzzles represent scientific problems, and players who solve them are directly helping science. The players are not explicitly aware that the puzzles represent any scientific problem. Due to the complexity of the science involved, only high-level players will ever see real problems; other players will see simpler puzzles that don't provide any research benefit. Players need to be engaged long enough that they graduate up to the higher levels. While the actual game will be a large part of the motivational draw, the team is looking at other ways to motivate play to try and maximize the number of players who graduate to real problems.

The game uses a gardening theme, and each problem is represented as a plant. The plant representation is tied to the underlying scientific problem. The plant cannot transform or grow in the problem screen. Players will interact with many problems, so will see many plants. Players find relationships about the plant, then submit them to a server to be checked for validity. It is expected that easier plants take less than five minutes, whereas later plants could take hours to work on. Plants represent problems, and the problems are automatically generated, so it is possible that some plants are more aesthetically desirable than others.

13.3.2 Pattern choice

As *Xylem* is a game, it makes sense to first start with gameful patterns, and SCORE, LEADERBOARDS, COLLECTION and GROWTH seem particularly applicable. SCORE and LEADERBOARDS provide a way for users to ascertain how they are doing and engage in competition if they wish, while COLLECTION and GROWTH let players create implicit goals for themselves. GROWTH will prove difficult to implement

263

as the plant itself can't grow, but the power of SMALL CAPS GROWTH in *Tiny Tower* indicates that it should be present in some capacity in *Xylem*.

The difficulty of the game means that the interface patterns PRAISE and UNDO should make an appearance, as well as the social patterns META-AREA and BROADCAST for people to share problems, ideas and solutions. A CONTACT LIST (ideally imported from an existing social graph) would allow players to share problems with one another, and feelings of *Honor* may cause them to re-engage with the app to help, while trying to avoid creating a DARK PATTERN: SOCIAL PYRAMID SCHEME.

As the game is on a mobile platform, STATE PRESERVATION should be implemented, as well as NOTIFICATIONS informing the user of game events while they aren't in the application.

This creates a list of the following patterns:

- SCORE

- LEADERBOARDS

- COLLECTION

- GROWTH

- PRAISE

- UNDO

- META-AREA (not mocked up, external to the app)

- BROADCAST

- CONTACT LIST

- STATE PRESERVATION

- NOTIFICATIONS

264

13.3.3 Annotated mockups

Mockups for *Xylem* appear on the following pages.
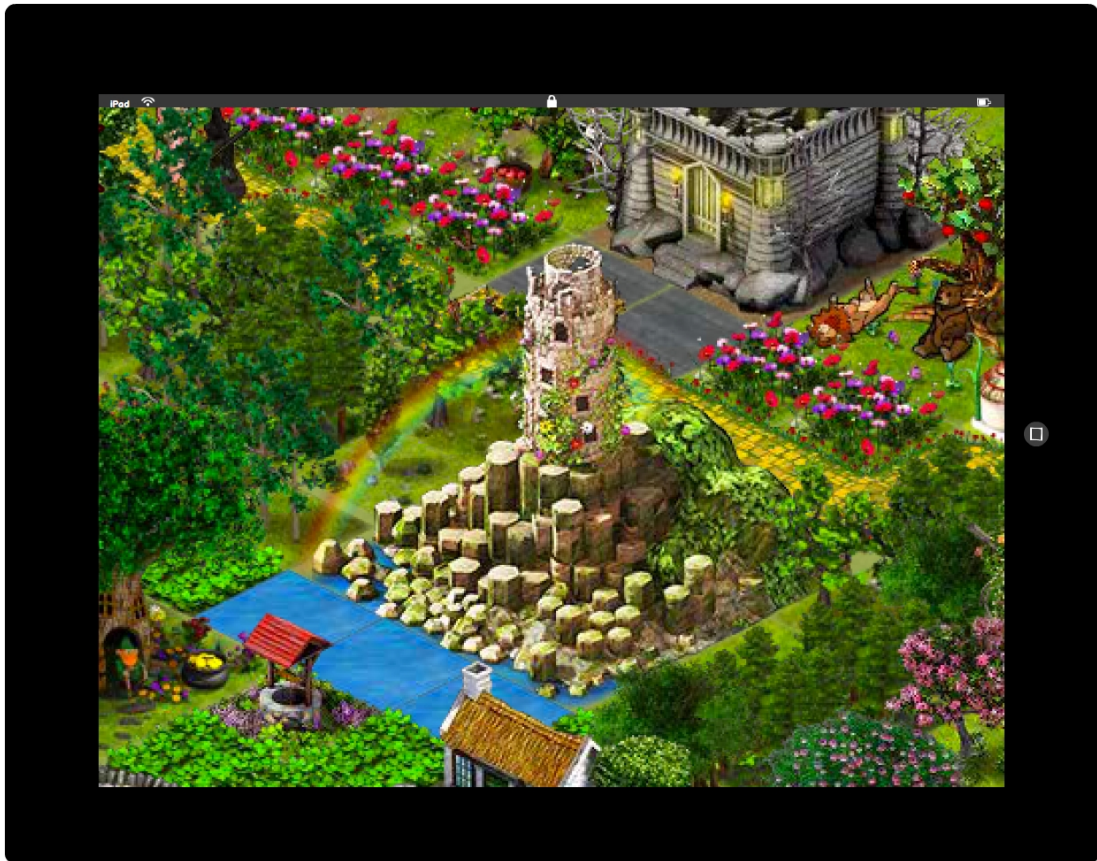
Figure 13.8: To incorporate GROWTH into *Xylem*, a garden could be used, where users can plant plants that they have won by answering problems correctly. These plants should themselves grow over a certain time period, say a week, from when they are planted. This creates a delayed time effect that primes users to come back to see how their garden is progressing. The use of a garden area also incorporates the COLLECTION pattern. By allowing users to visit each other's gardens, the garden can be used by players as a means of IDENTITY SHAPING. As the plants for each problem are different, the META-AREA could be utilized, where users could ask for plants that have particular attributes that they wish for their garden. For example, a player may want more pink flowers, and could be pointed towards problems when plants with pink flowers could be won.

This figure uses a screenshot from *Gardens of Time* to illustrate how the garden could look. Taken on April 14, 2013.

266
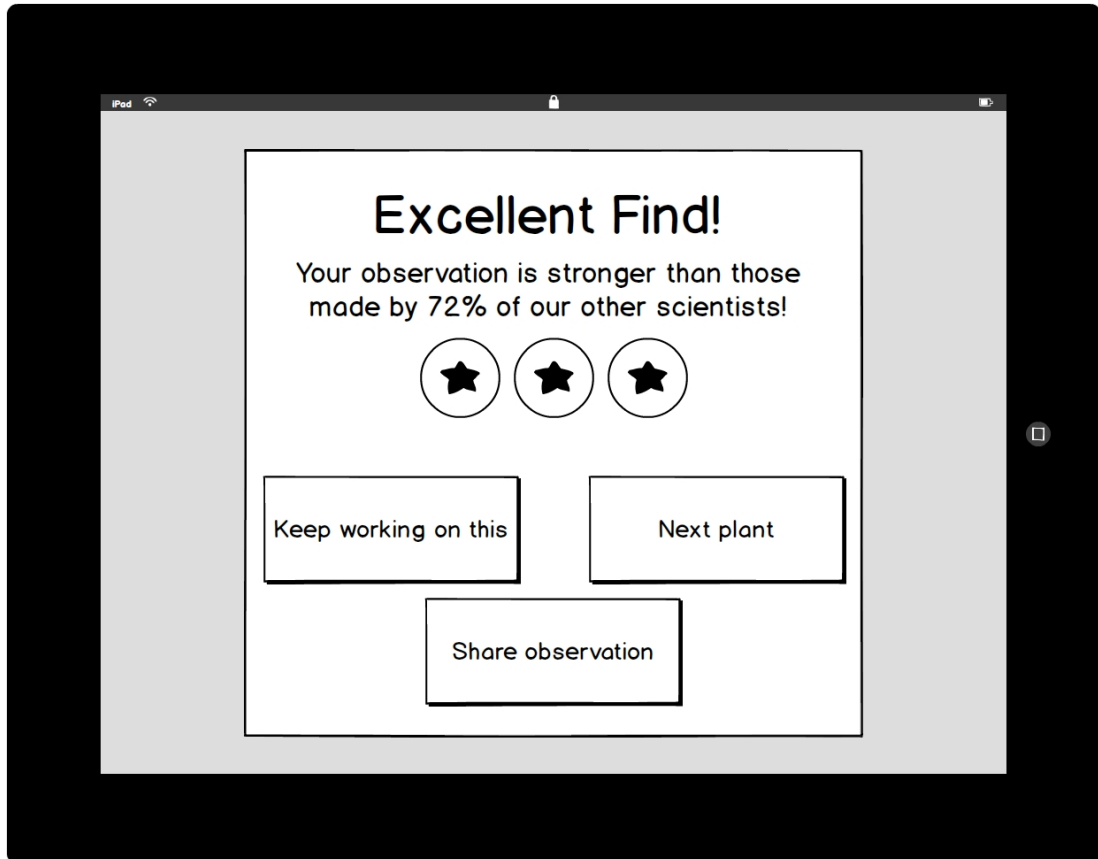
Figure 13.9: When players make a strong observation, they are PRAISED for doing well, but the wording deliberately tries to avoid being controlling, and instead just provide feedback for players. When players do better than 50% of the rest of the player population, they are informed about this. This allows them to gauge how well they are doing, without needing to resort to a LEADERBOARD view. Answers can be BROADCASTED to others to help them out with difficult plants, and perhaps inspire new answers.
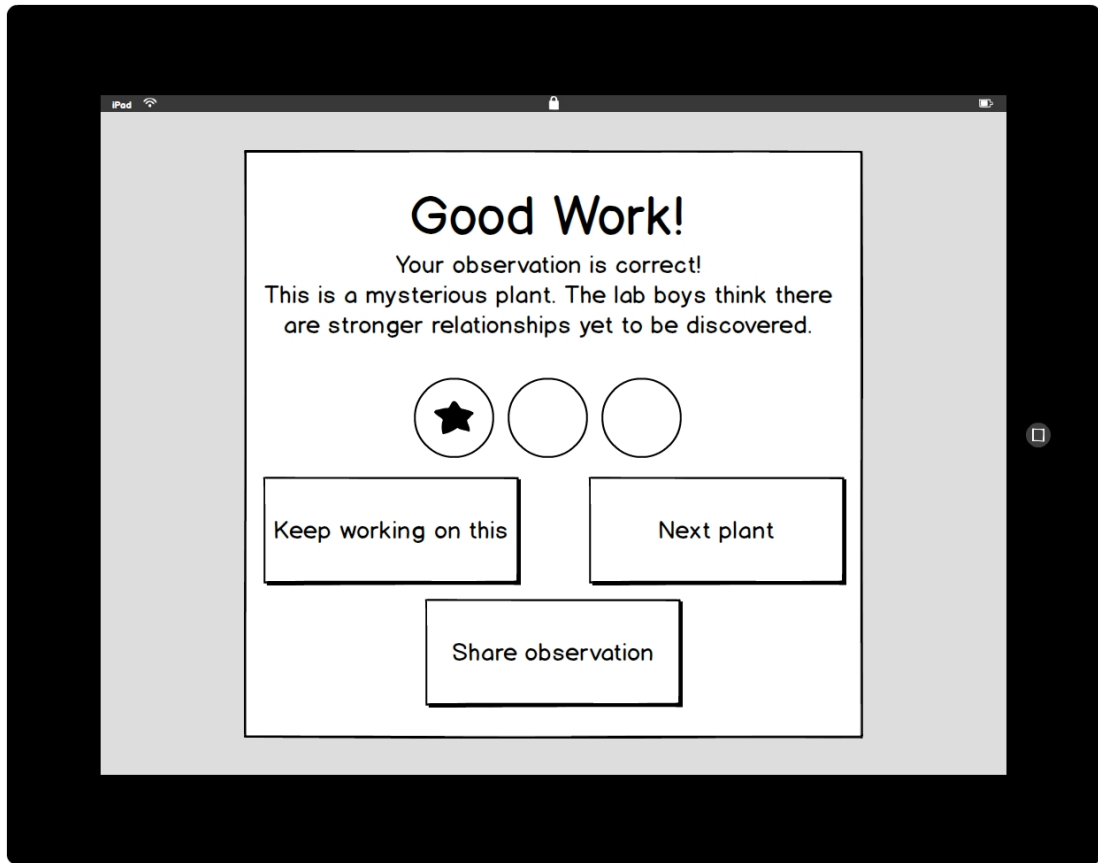
Figure 13.10: Weak answers are more troublesome to word without being controlling. Phrases like "You can do better!" or "Keep it up!" could undermine *Independence*, but showing just how weak the answer is with a percentage is demoralizing. The player is simply informed that there are other relationships that could be found, without being instructed that *she* should be the one to do it.
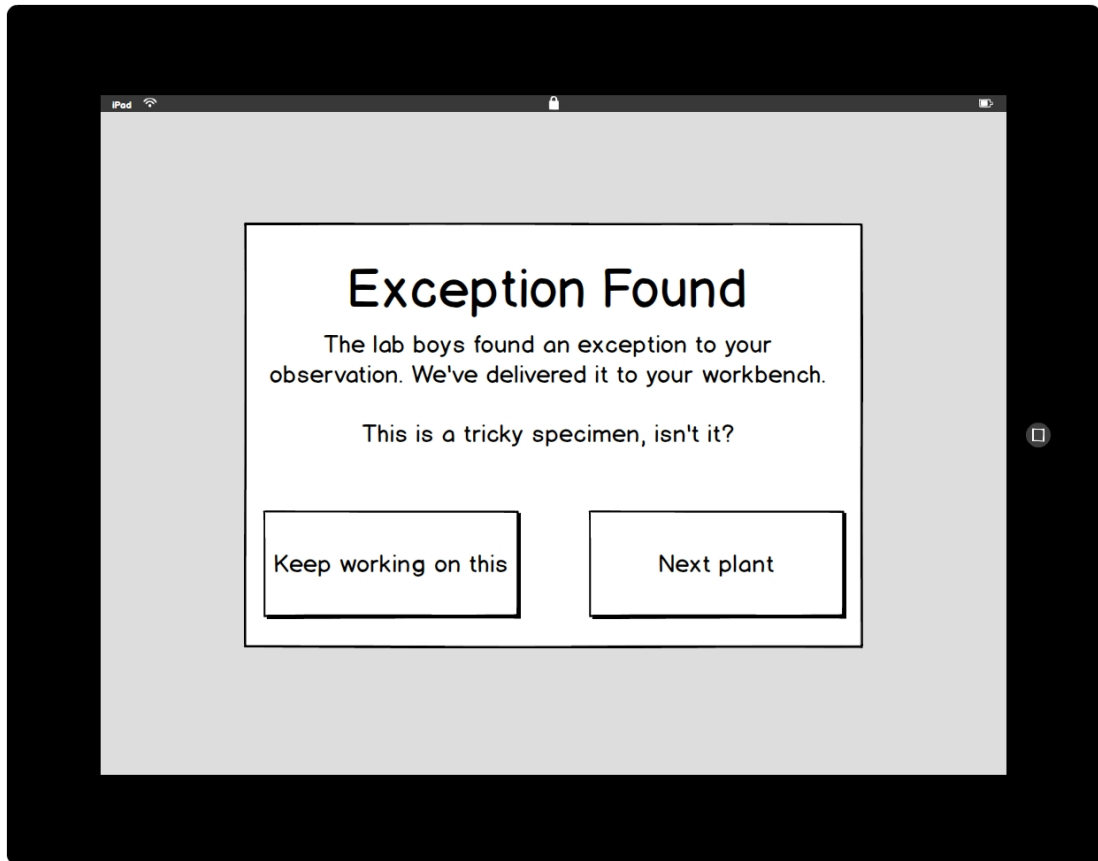
Figure 13.11: As with the weak answer, the temptation to put in a controlling call to action when the player answers incorrectly is high. Instead, the message acknowledges the possibility that the player is struggling, and validates those feelings by blaming the plant. This hope is this creates a feeling of *Acceptance*, even when the player isn't performing well.
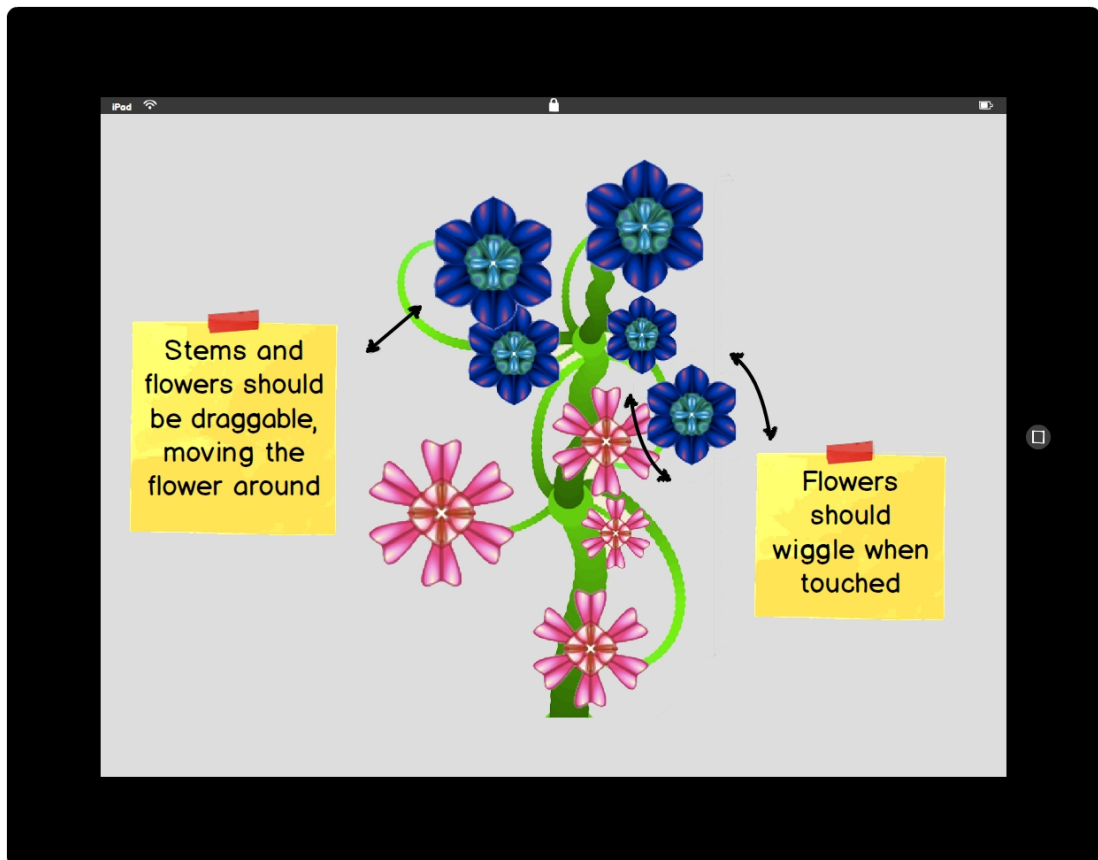
Figure 13.12: To create a feeling of *Acceptance*, PRAISE is used throughout the interface. Making the plant tactile, by having the stems and flowers respond to user input, ensures that users feel like the interface is listening to their input.
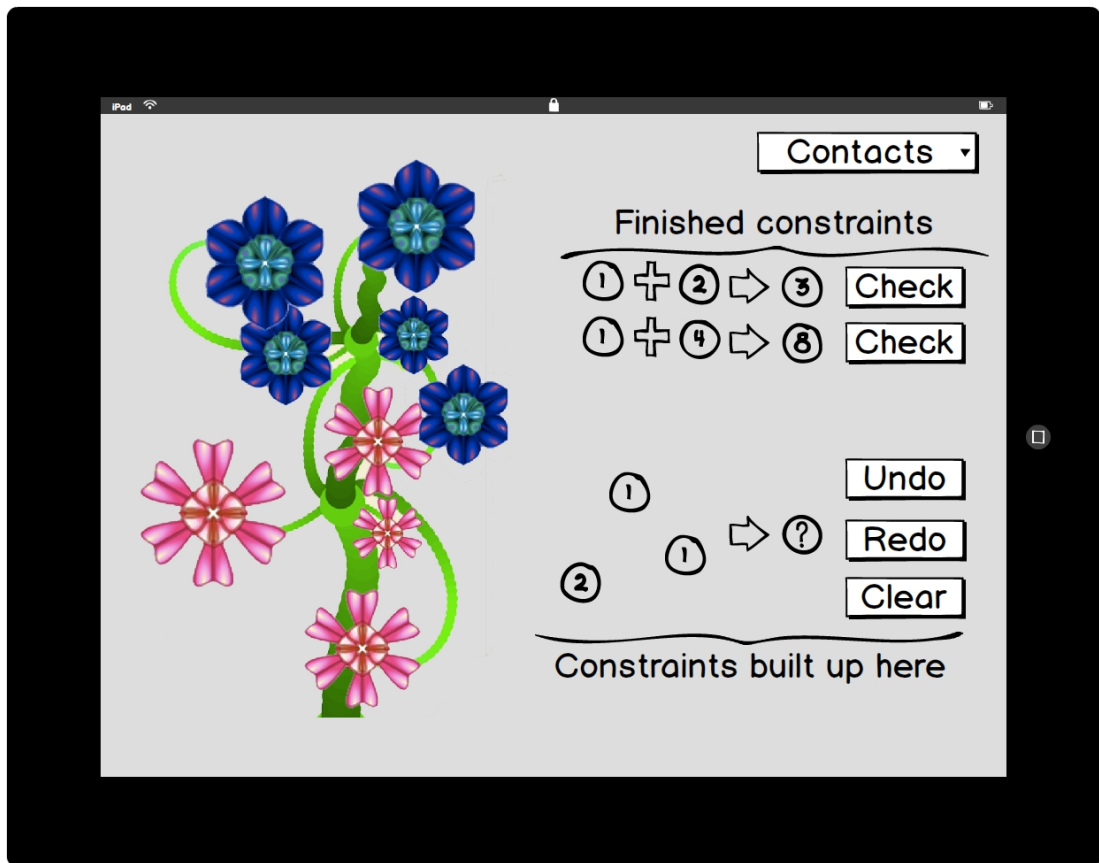
Figure 13.13: As users build possible answers, a clear UNDO button is provided to allow them to explore possible ideas.
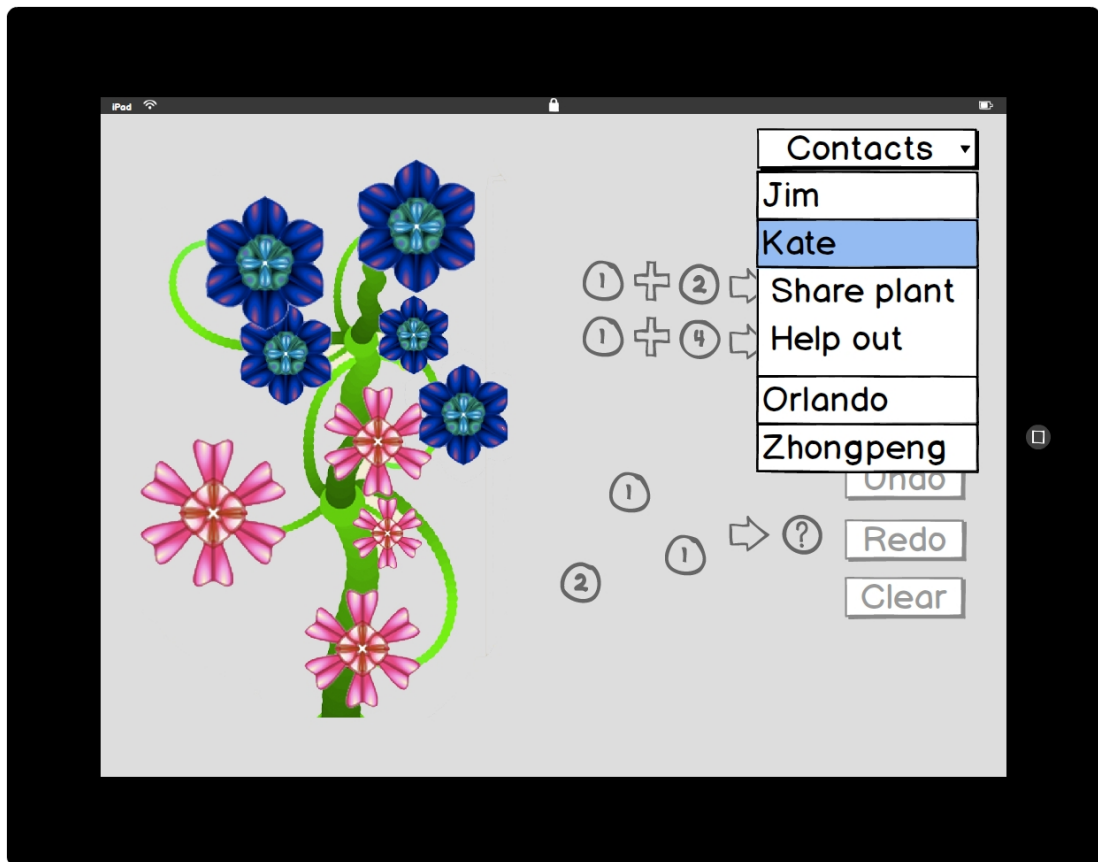
Figure 13.14: The CONTACT LIST lets users quickly find their friends to BROADCAST the plant they are working on. They can also see friends in need of help, and can work on their current plant, sending back a possible answer.
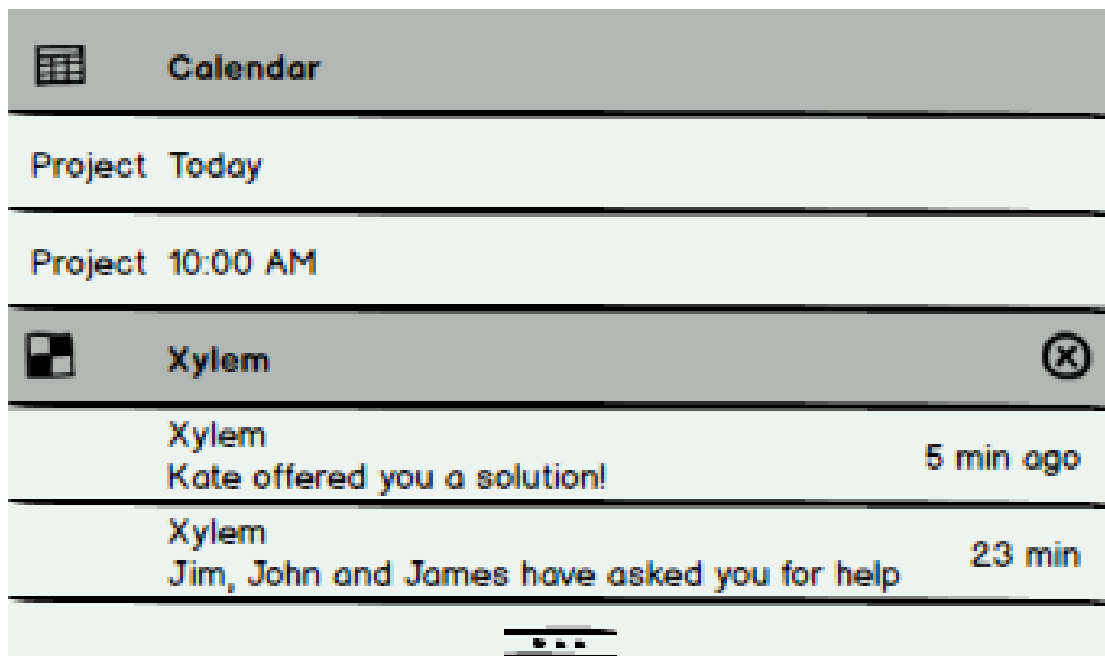
Figure 13.15: As the Growth of the garden should provide enough prompting for users to return to the application, Notifications are only used to inform players that their friends are trying to get in contact. The second notification combines three separate notifications into one. It does this by collecting notification requests, then sending them as a batch on an interval (which could be user-specified), so the user is not spammed with multiple notifications leading to Dark Pattern: Interaction by Demand. The messages themselves are worded informationally, to avoid using controlling language that may give the impression of Dark Pattern: Social Pyramid Scheme.

Chapter 14

Conclusions

14.1   Summary of contributions

In this dissertation, twenty-seven motivational design patterns were presented, across gameful, social, interface and information categories. These patterns were discovered by utilizing the new framework provided by motivational user stories, looking for patterns that fit into the "As a <user>, I want <something> so that <Reiss desire is met>" paradigm. Formulating user stories in this way allows designers to focus on which desires they wish their motivational software to meet. Motivational user stories bring motivational ideas to the forefront, ensuring that motivational desires, not just functionality, is a key part of the design process.

The patterns presented in the library are not only a description of common design idioms in motivational software, but contain supporting data from motivational theory, behavioral psychology and behavioral economics to help us understand why certain patterns work. The library is not just a description of what is, but contains descriptions of what can be, utilizing psychology to help point out and improve poor implementations.

The pattern library is useful for a broad swathe of people. For inexperienced

designers, the pattern library offers a convenient "cookbook" that can be used to build motivational software quickly. For more experienced designers with experiential knowledge in this area, the library's description of the background psychology that supports their work will help them better understand the implications of the patterns they use. For academics and others who study software, the library can be used to pinpoint design decisions, and perhaps infer how those decisions affect the user base.[1]

Eight motivational dark patterns were also presented. Patterns such as these are often avoided in academic literature for fear of appearing subjective. To counter such a fear, the eight patterns are clearly framed as creating a loss of user resources that those users did not consent to, and as creating a reduction in their Reiss motivations. Scholars have been hesitant to label behaviors as "questionable" (or worse), but we must do so if we are to improve the current design environment. Explaining what is bad, and giving clear reasoning why, is a key contribution of this dissertation. For designers, the takeaway is that such short-term thinking inevitably harms users' long term motivational satisfaction. Users will begin to understand and recognize these dark patterns, and they will learn to avoid them, or otherwise give their consent to participate in a manipulative environment. This idea is presented as *manipulation literacy*, a new theory contributed by this dissertation. If users give their consent, designers are acting ethically. Without that consent, the use of dark patterns is dissuaded, and other possible strategies were presented.

Four case studies were used to show how the pattern library could be used as a means of analyzing current software, but also as a means of generating entirely new software designs. These case studies showed how broadly applicable the library

---

[1]The library even serves to illuminate what software *means* to users. SEARCH, for example, is not just useful to lower annoyance, speeding up frustrating tasks. SEARCH also provides an important safety net. Many people find the idea of the theft of their physical photo albums deeply traumatizing. Photos that are misplaced on our computers are essentially lost as well, and SEARCH is a means to avoid that trauma. Our lives live on our computers, and motivational design patterns brings this, and what that means for our core well-being, into sharp focus.

is, helping explain how *Khan Academy* and *Tiny Tower* are built, and create their motivational draw. Aspects of *Khan Academy*'s gameful patterns were criticized for being too simplistic, while *Tiny Tower*'s use of Dark Pattern: Interaction by Demand was singled out as an unnecessary addition to the game. This shows that the pattern library can even help to improve such popular applications as these. Two further case studies were performed to show how the pattern library can help generate new software, one creating a fictional voting web site called *Votester*, the other adding patterns to *Xylem*, a research game already in development. The library helped make easy work of both case studies, effectively guiding design ideas to fruitful conclusions.

14.2 Future work

14.2.1 Expansion of the pattern library

While the twenty-seven patterns here appear to provide good coverage of many of the designs in use today, there are undoubtedly more to discover, and more will be generated as software continues to evolve. While approaching the staggering 253 patterns that Alexander (1977) found seems unlikely, a solid library of around sixty to one hundred patterns is within the realm of possibility. The beauty of pattern libraries is that adding more does not diminish those that are already there. Continued work on this front can only bring positive outcomes.

Expanding the pattern library is not simply a matter of academic completeness.[2] Rather, it is an implicit announcement that the software we create must be designed with the motivational requirements of the user in mind. Expansion of the library underlines just how much software designers today, knowingly or not, lean upon motivation theory. And with the expansion of the library comes more utility and understanding to designers of software in the future.

---

[2]Given prototype theory's rejection of objectivism, completeness may not even exist.

One other benefit of continuing to work on the pattern library will be better understanding from computer science about what one may term "the effects": the well-known experiments from psychology research that have garnered a name such as the "Zeigarnik effect" or the "endowment effect." Throughout the discovery of these patterns, time and again the effects have offered strong experimental insight into why certain patterns work the way they do. The finite time-scale with which this dissertation was written means there are many more effects which could simply not be analyzed and considered. A wealth of knowledge exists in the effects, and there are certainly many more patterns to be found with more effects that can be used to explain them.

14.2.2 Further research into motivational dark patterns

When talking to game designers during the research process, it seemed clear that many of them had more knowledge about dark patterns than they were willing to discuss. While some musty tome, scrawled with various demonic symbols, labelled "Motivational Dark Patterns," doesn't exist, there does seem to be some folk knowledge of particular unsavory tactics that has been spread via word-of-mouth.[3] This information should be gathered and preserved while it is still possible to do so. The framework provided by motivational dark patterns can give name and form to this folk knowledge, making the ideas accessible to all. Game designers are particularly of interest, as the games industry, led by companies such as Zynga, moved quickly to novel monetization strategies. In turn, these strategies needed support from the design side to encourage users to hand over their time or money.

From a disinterested academic view, the loss of this knowledge would do a disservice to science. These companies performed A:B testing on the scale of millions of users, a scale which academia will struggle to match. Such experiments may not

---

[3]While some work would likely be covered under Non-Disclosure Agreements, the folk knowledge would likely not be.

tell us why users act in a way they do, but they certainly would provide many insights into what aspects of dark patterns were effective. With that knowledge in hand, the psychology literature presented in this dissertation and from knowledge from further afield can provide the *why* that A:B testing can't.

From the point of view of those who are questioning the ethical aspects of dark patterns, this knowledge would be useful in ascertaining why certain dark patterns were being utilized and formulating consumer-friendly patterns that can be used instead, or creating educational campaigns to increase the motivational literacy of different audiences to attempt to nullify the effectiveness of these dark patterns.

14.2.3   Focus on mobile computing

The growth of mobile continues unabated. As a case in point, *Facebook* is now more often accessed by mobile platforms than the desktop, with Mark Zuckerberg proclaiming that it's now a "mobile company" (Olanoff 2013). While the patterns in the library are largely platform-agnostic, is is likely that mobile will introduce patterns that are entirely separate from other platforms. Mobile deserves, and perhaps requires, its own focus, distinct from studies of other platforms.

Analyzing mobile is not simply a function of its popularity; although this is a good reason.[4] Mobile is a core platform for motivational computing, and it is no accident that the very first exposure to motivational software presented to the reader in this dissertation is a smartphone. The omnipresence of Internet access within our reach allows us to satiate our motivational desires very quickly. We can pop open a game to satiate *Vengeance*, send a tweet to feel more *Social Contact* or search *Wikipedia* to

---

[4]Academia doesn't tend to accept that the popularity of something is justification enough for its study. However, while popularity in and of itself is not worthy of study, it indicates that there is something valuable to be learned: why are people dropping their desktop computers for iPads? Why do first-person shooters sell millions of copies year after year? That something is popular but deemed unworthy of study is indication that what is worthy of study has been missed, not that it is not there at all.

satisfy our *Curiosity*. And we can do all these things in seconds. It's tempting to think of motivational software as being a natural fit for mobile. Perhaps it is more instructive to think of mobile as being a natural fit for motivational software; that it is motivational software that is driving the growth of mobile.

This viewpoint will likely require a more holistic view of motivational software than just the artifact itself. While extra-application patterns such as NOTIFICATIONS and META-AREA are considered, an analysis of mobile patterns should analyze interactions methods as well. The way that users will interact with Google Glass, for example, will be different than how they interact with their smartphones. Those interactions modify what motivational desires users can expect to be fulfilled, and how much time that fulfillment will take. The change in interaction could also modify some patterns to be dark patterns, and vice versa. For example, our need for *Order* might make a TASK QUEUE untenable on a Google Glass-like heads-up display. A constant nagging that something is being left undone could upset many users in a way they would not be if the TASK QUEUE was part of a desktop interface. The ability for mobile to so dramatically alter patterns is a strong indication that it needs to be studied explicitly.

Motivational design patterns offer an excellent means of describing the needs that mobile platforms intend to meet. As mobile is still relatively new, and popular wearable computing has only just begun, motivational design patterns brought in at this early stage could help guide the evolution of mobile along a more focused path than the experiential trial-and-error we see today.

14.2.4 Broadening education

There is little time in computer science undergraduate programs to explore topics outside the realm of computing.[5] We teach students how to be engineers and give them the tools to build things, but spend very little time on teaching them *what* to

---

[5]This is especially true in countries like the UK, where bachelors programs are only three years long.

build or *why.* In years past, this was not such an important issue. It was enough to find a task that took too long, be it in the analog or digital worlds, identify how computers could help, then build a program so the task can be done faster. Motivational software is not about such tasks. Motivational software is about humans, engaging them, and understanding what they need at an emotional level.

All motivational software designers need to be journeymen psychologists, just as game designers are. Both must focus on the user/player, and as Schell (2008, p. 4) writes, "[A game designer's] goal is to make a human being happy. [She] must understand the workings of the human mind or [she] is designing in the dark." To properly address this, we need to consider making psychology classes a key part of computer science in the same way we consider mathematics.[6] Mathematics helps students with the *how* to build, and psychology helps students with the *what.* In a world where multi-million dollar empires can be started by individual students designing motivational software in dorm rooms,[7] computer science teaching must adapt. The growth of video game design programs provides an excellent indication that computer science departments are broadening. In the short term, one can imagine game design lecturers offering individual classes in the "Psychology of users" for all computer science majors. In the longer term, stronger collaboration between computer science and psychology, perhaps via the current partnerships in human-computing interaction, could lead to more holistic programs than we have now.

---

[6]Computer science is not necessarily supposed to be about the building of software, recalling the famous Dijkstra quote "Computer Science is no more about computers than astronomy is about telescopes." However, it would require a very extreme viewpoint to completely reject the idea that computer science undergraduate programs are not currently geared towards preparing their students for working in a job where they create software.

[7]Yes, this is Facebook.

14.3 Conclusion

In this dissertation, the motivational design patterns have been presented. Research from motivational theory, behavioral psychology and behavioral economics, more seldom seen in computer science literature than it should be, has been used. New ideas were required to properly deal with motivational software design. In many ways, all of this work can be seen as foundational, only barely hinting at what could be to come.

The joy of foundational work takes two parts. The first is the feeling that something that was previously invisible has now been brought to light. Many of the patterns that designers are using, whether they knew it or not, have strong connections to our motivational needs. Making these connections has made explicit what was once implicit, and that makes design *better*. We can now say, with some confidence, why a certain pattern works. We can decide what patterns should be used for a certain product. We can understand how best to implement a pattern. Having such knowledge written down is not just an academic exercise, but one which benefits everyone. This dissertation was written with the designer as the intended audience, and hopefully that audience does find this.

The second joy of foundational work is the promise of what is to come. The patterns here are only a reflection of what we have now. How much better could the needs of users be met if we truly took the time to understand what motivates them, rather than resorting to 'gamification' and 'social'? What software would we make? What patterns would we use? It is exciting to speculate about evolution, or even revolution, that could occur when motivational design patterns are thought about deeply by many people, rather than just one.

Motivational design patterns provide the language we need to discuss the future of motivational software. The conversation can now begin.

Appendix A

Loss-Aversion in Gamers Experiment

Questionnaire

Agreement to participate

You are being invited to participate in a research study to ascertain how people react to this game. It should take about 5 minutes. You must be 18 or over to participate.

You'll fill out a short survey on the next page, then return to the game and play it. You will not experience any risk or discomfort from the questions or the game.

You will not receive any benefit from participating in this research.

The game collects data about your game choices, and will link your survey to that game automatically. Your data is stored without identifying you in any way. The records will only be accessed by Chris Lewis and Professor Jim Whitehead.

You may contact Chris Lewis at 831-234-3387 any time you have questions about the research. If you have questions about your rights as a research subject, you may contact the Office of Research Compliance Administration at (831) 459-1473 or orca@ucsc.edu.

Your participation in this research is voluntary, and you will not be penalized or lose benefits if you refuse to participate or decide to stop.

By continuing this survey, you assert that you are 18 or over, and that you understand and agree to everything in this statement.

*1. Agreement

○ I certify that I am 18 or older and agree to the statement above

Main survey

✱2. How old are you (numerals only please, eg. 19)?

Age

✱3. What is your main job or college major?

○ Agriculture, Agriculture Operations, and Related Sciences

○ Architecture and Related Service

○ Area, Ethnic, Cultural, and Gender Studies

○ Biological and Biomedical Sciences

○ Business, Management, Marketing, and Related Support Services

○ Communication, Journalism, and Related Programs

○ Communications Technologies/Technicians and Support Services

○ Computer and Information Sciences and Support Services

○ Construction Trades

○ Education

○ Engineering

○ Engineering Technologies/Technicians

○ English Language and Literature/Letters

○ Family and Consumer Sciences/Human Sciences

○ Foreign languages, literatures, and Linguistics

○ Health Professions and Related Clinical Sciences

○ History

○ Legal Professions and Studies

○ Library Science

○ Mathematics and Statistics

○ Military Technologies

○ Multi/Interdisciplinary Studies

○ Natural Resources and Conservation

○ Parks, Recreation, Leisure, and Fitness Studies

○ Philosophy and Religious Studies

○ Physical Sciences

○ Psychology

○ Public Administration and Social Service Professions

○ Social Sciences

○ Visual and Performing Arts

*4. How many years have you played video games?

	Never	1-4	5-10	10+
Years	○	○	○	○

*5. Would you describe yourself as a gamer?

○ No

○ Yes

6. What are your favorite game genres? (Optional, and it's OK to approximate, many games don't easily fit into genres)

☐ Adventure

☐ Strategy

☐ Simulation (including driving)

☐ Puzzle

☐ Action-adventure

☐ Music

☐ Shooter (First or Third person)

☐ Sports

☐ Action (including platformers, arcade games)

☐ Role-playing (RPG)

*7. What slice of your gaming time do you use to play strategy games?

	Never	Sometimes	Half	Often	I only play strategy games
Amount	○	○	○	○	○

*8. How many matches of Starcraft 1 or 2 have you played before?

	Never	Less than 5 games/matches	5 to 20 games/matches	20+ games/matches
Single-player	○	○	○	○
Multiplayer	○	○	○	○

*9. Have you played Starcraft 1 or 2 in the last 30 days?

	Yes	No
Single-player	○	○
Multiplayer	○	○

Bibliography

A., Manila (2011). "Why is Smaller Often Better When it Comes to Online Communities?" In: *Communispace*. URL: http://blog.communispace.com/learn/why-is-smaller-often-better-when-it-comes-to-online-communities/.

Abowd, Gregory D and Alan J Dix (Dec. 1992). "Giving undo attention." In: *Interacting with Computers* 4.3, pp. 317–342. ISSN: 09535438. DOI: 10.1016/0953-5438(92)90021-7.

Aidan, Julian (2013). "Reading into Reddit." In: *The Michigan Daily*. URL: http://www.michigandaily.com/arts/1reddit-b-side-lede31?page=0,0.

Alexander, Christopher (1977). *A Pattern Language*. Ed. by S Ishikawa and M Silverstein. Vol. 21. 2. Oxford University Press. Chap. 52, p. 1171. ISBN: 0195019199. DOI: 10.1145/1529966.1529967.

Ambler, Scott W. "Introduction to User Stories." In: *Agile Modeling*. URL: http://www.agilemodeling.com/artifacts/userStory.htm.

Anders, George (2012). "How LinkedIn Has Turned Your Resume Into A Cash Machine." In: *Forbes*. URL: http://www.forbes.com/sites/georgeanders/2012/06/27/how-linkedin-strategy/.

Andersen, Erik, Yun-En Liu, Richard Snider, Roy Szeto, Seth Cooper, and Zoran Popović (June 2011). "On the harmfulness of secondary game objectives." In: *Proceedings of the 6th International Conference on Foundations of Digital*

Games - FDG '11. New York, New York, USA: ACM Press, pp. 30–37. ISBN: 9781450308045. DOI: `10.1145/2159365.2159370`.

Andreoni, James (1990). "Impure Altruism and Donations to Public Goods: A Theory of Warm-Glow Giving?" In: *Economic Journal* 100.401, pp. 464–77.

Antin, Judd (2012). *What Lies Beneath: The Foundations of Motivation in the Age of Social Media*. Tech. rep. CITRIS. URL: `http://www.youtube.com/watch?v=Rr1gvReCfZQ`.

Arguello, Jaime, Brian S. Butler, Lisa Joyce, Robert Kraut, Kimberly S. Ling, and Xiaoqing Wang (Apr. 2006). "Talk to me." In: *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (CHI '06)*. New York, New York, USA: ACM Press, p. 959. ISBN: 1595933727. DOI: `10.1145/1124772.1124916`.

Ariely, Dan (2010). *Predictably Irrational, Revised and Expanded Edition: The Hidden Forces That Shape Our Decisions*. Harper Perennial; 1st edition, p. 384. ISBN: 0061353248.

Ariely, Dan and Michael I Norton (Jan. 2009). "Conceptual consumption." In: *Annual Review of Psychology* 60, pp. 475–99. ISSN: 0066-4308. DOI: `10.1146/annurev.psych.60.110707.163536`.

Attwood, Jeff (2013). "Civilized Discourse Construction Kit." In: *Coding Horror*. URL: `http://www.codinghorror.com/blog/2013/02/civilized-discourse-construction-kit.html`.

Axelrod, Robert and William Donald Hamilton (1981). "The Evolution of Cooperation." In: *Science*. New Series 211.4489, pp. 1390–1396. ISSN: 00368075. DOI: `10.1126/science.7466396`.

B., Sean (2013). "Why was my email leaked?" In: URL: `https://forums.dropbox.com/topic.php?id=97303\&page=2\#post-530452`.

Barras, Adam (2012). "Managing app lifecycle so your apps feel "always alive"." In: *Windows 8 App Developer Blog*. URL: `http://blogs.msdn.com/b/windowsappdev/archive/2012/04/10/managing-app-lifecycle-so-your-apps-feel-quot-always-alive-quot.aspx`.

Baumeister, R F, E Bratslavsky, M Muraven, and D M Tice (1998). "Ego depletion: is the active self a limited resource?" In: *Journal of Personality and Social Psychology* 74.5, pp. 1252–65. ISSN: 00223514. DOI: `10.1037/0022-3514.74.5.1252`.

Beizer, Boris (1990). *Software Testing Techniques*. 2nd ed. Van Nostrand Reinhold, p. 550. ISBN: 0442206720.

Berdichevsky, Daniel and Erik Neuenschwander (May 1999). "Toward an ethics of persuasive technology." In: *Communications of the ACM* 42.5, pp. 51–58. ISSN: 00010782. DOI: `10.1145/301353.301410`.

Bergman, Ofer, Steve Whittaker, Mark Sanderson, Rafi Nachmias, and Anand Ramamoorthy (Dec. 2010). "The effect of folder structure on personal file navigation." In: *Journal of the American Society for Information Science and Technology* 61.12, pp. 2426–2441. ISSN: 1532-2882. DOI: `10.1002/asi.v61:12`.

Berlinger, Joshua (2012). "A Few More Ways That Supermarkets Mess With Your Minds." In: *Business Insider*. URL: `http://www.businessinsider.com/the-psychology-behind-supermarkets-2012-11`.

Björk, Staffan (2011). "Encouraged Return Visits." In: *Game Design Patterns 2.0*. URL: `http://gdp2.tii.se/index.php/Encouraged_Return_Visits`.

Björk, Staffan and Jussi Holopainen (Dec. 2004). *Patterns in Game Design*. 1st ed. Charles River Media. ISBN: 1584503548.

Boag, Paul (2012). "Are you giving your users some positive feedback?" In: *Smashing Magazine*. URL: `http://www.smashingmagazine.com/2012/07/17/are-giving-users-positive-feedback/`.

Bogost, Ian (2005). "Procedural Literacy: Problem solving with programming, systems and play." In: *Journal of Media Literacy* 52.1-2, pp. 32–36.

— (2011). "Gamification is Bullshit." In: *Wharton Gamification Symposium*. URL: http://www.bogost.com/blog/gamification_is_bullshit.shtml.

— (2012). "The Cigarette of This Century." In: *The Atlantic*. URL: http://www.theatlantic.com/technology/archive/2012/06/the-cigarette-of-this-century/258092/.

boyd, danah (Aug. 2012). "The Politics of "Real Names"." In: *Communications of the ACM* 55.8, p. 29. ISSN: 00010782. DOI: 10.1145/2240236.2240247.

Brickman, P and D T Campbell (1971). "Hedonic relativism and planning the good society." In: *Adaptation level theory A symposium*. Ed. by M H Appley. Academic Press, pp. 287–302.

Brignull, Harry (2010a). "Dark Patterns: dirty tricks designers use to make people do stuff." In: *Harry Brignull's 90 Percent of Everything*. URL: http://www.90percentofeverything.com/2010/07/08/dark-patterns-dirty-tricks-designers-use-to-make-people-do-stuff/.

— (2010b). "UX Brighton Presentation on Dark Patterns." In: *UX Brighton*. Brighton. URL: http://www.90percentofeverything.com/2010/09/13/ux-brighton-presentation-on-dark-patterns/.

— (2011). "Dark Patterns: Deception vs. Honesty in UI Design." In: *A List Apart*. URL: http://www.alistapart.com/articles/dark-patterns-deception-vs.-honesty-in-ui-design/.

— (2012). "Friend Spam." In: URL: http://wiki.darkpatterns.org/Friend_Spam/index.html.

Brown, William J., Raphael C. Malveau, Hays W. "Skip" McCormick, and Thomas J. Mowbray (1998). *AntiPatterns: refactoring software, architectures, and projects in crisis*. Wiley, p. 309. ISBN: 0471197130.

289

Brun, H (2012). "Big Battlefield 3 PS3 day: Game update, Rent a Server functionality, and shortcut items now live!" In: *Battlefield Official Blog*. URL: http://blogs.battlefield.com/2012/03/ps3-update-live/.

Butler, Brian S. (Dec. 2001). "Membership Size, Communication Activity, and Sustainability: A Resource-Based Model of Online Social Structures." In: *Information Systems Research* 12.4, pp. 346–362. ISSN: 1047-7047. DOI: 10.1287/isre.12.4.346.9703.

Cameron, J. (Jan. 2001). "Negative Effects of Reward on Intrinsic Motivation–A Limited Phenomenon: Comment on Deci, Koestner, and Ryan (2001)." In: *Review of Educational Research* 71.1, pp. 29–42. ISSN: 0034-6543. DOI: 10.3102/00346543071001029.

"Can the can" (2008). In: *The Economist*. URL: http://www.economist.com/node/12630201?story_id=12630201.

Carvalho, Daniel (2009). "What Have We Achieved." In: *Daniel Carvalho*. URL: http://danielcarvalho.com/articles/what-have-we-achieved/.

Chance, Paul (1998). *Learning and Behavior*. 4th ed. Wadsworth, p. 623. ISBN: 053434691X.

Chiesa, Mecca (1994). *Radical Behaviorism: The Philosophy and the Science*. Cambridge Center for Behavioral Studies, p. 241. ISBN: 0962331147.

Church, Doug (July 1999). "Formal Abstract Design Tools." In: *Gamasutra*. URL: http://www.gamasutra.com/view/feature/3357/formal_abstract_design_tools.php.

Cialdini, Robert B. (2008). *Influence: Science and Practice (5th Edition)*. Pearson, p. 272. ISBN: 0205609996.

Cohn, Mike (2004). *User stories applied: for agile software development*. Addison-Wesley Professional, p. 268. ISBN: 0321205685.

Constantine, Larry L and Lucy A D Lockwood (2002). "Instructive Interaction : Making Innovative Interfaces Self-Teaching." In: *User Experience* 1.3, pp. 1–15.

Crumlish, Christian and Erin Malone (2009). *Designing Social Interfaces: Principles, Patterns, and Practices for Improving the User Experience.* O'Reilly Media, Inc., p. 520. ISBN: 1449379222.

Csíkszentmihályi, Mihály (1975). *Beyond boredom and anxiety: The experience of play in work and games.* Jossey-Bass, p. 231. ISBN: 0875892612. URL: `http://www.worldcat.org/title/beyond-boredom-and-anxiety-the-experience-of-play-in-work-and-games/oclc/317667075\&referer=brief\_results`.

— (1997). *Finding Flow: The Psychology of Engagement With Everyday Life.* Basic Books, p. 181. ISBN: 0465024114.

Curtis, Tom (2011). "Tiny Tower, Dead Space Lead Apple's Game Of The Year Picks." In: *Gamasutra.* URL: `http://www.gamasutra.com/view/news/39047/Tiny_Tower_Dead_Space_Lead_Apples_Game_Of_The_Year_Picks.php`.

Dash, Anil (2011). "If your website's full of assholes, it's your fault." In: *Anil Dash.* URL: `http://dashes.com/anil/2011/07/if-your-websites-full-of-assholes-its-your-fault.html`.

Dawkins, Richard (1976). *The Selfish Gene.* Oxford: Oxford University Press.

"Dead or Alive 4 Achievements." In: URL: `http://www.xbox360achievements.org/game/dead-or-alive-4/achievements/`.

Deci, E L, R Koestner, and R M Ryan (Dec. 1999). "A meta-analytic review of experiments examining the effects of extrinsic rewards on intrinsic motivation." In: *Psychological bulletin* 125.6, 627–68; discussion 692–700. ISSN: 0033-2909.

Deci, Edward L and Richard M Ryan (1985). *Intrinsic Motivation and Self-Determination in Human Behavior.* Ed. by Plenum Press. Vol. 17. Perspectives in social psychology 2. Plenum Press, p. 371. ISBN: 0306420228. DOI: `10.2307/2070638`.

Deterding, Sebastian, Dan Dixon, Rilla Khaled, and Lennart Nacke (Sept. 2011). "From game design elements to gamefulness: defining "gamification"." In: *Proceedings of the 15th International Academic MindTrek Conference on Envisioning Fu-*

ture Media Environments (MindTrek '11). New York, New York, USA: ACM Press, p. 9. ISBN: 9781450308168. DOI: 10.1145/2181037.2181040.

Deturding, Sebastian (2011). "A Quick Buck by Copy and Paste." In: *Gamification Research Network.* URL: http://gamification-research.org/2011/09/a-quick-buck-by-copy-and-paste/.

Direkova, Nadya (2011). "Game On: 16 Design Patterns for User Engagement." In: *SxSW.* Google. URL: http://www.slideshare.net/gzicherm/nadya-direkova-game-on-16-design-patterns-for-user-engagement.

Doman, James (2012). "James Doman's answer to What is the definition of "personalization"?" In: URL: http://www.quora.com/What-is-the-definition-of-personalization/answer/James-Doman.

Dunbar, Robin (2011). *How Many Friends Does One Person Need? Dunbar's Number and Other Evolutionary Quirks.* Harvard University Press, p. 312. ISBN: 0674059328.

Email: The Easiest Way to Improve Retention. Tech. rep. Segment.io. URL: https://segment.io/academy/email-is-the-easiest-way-to-improve-retention.

F., Forrest (2013). "Why was my email leaked?" In: *Dropbox Forums.* URL: https://forums.dropbox.com/topic.php?id=97303\#post-530264.

Facebook Reports First Quarter 2013 Results (2013). Tech. rep. Facebook. URL: http://investor.fb.com/releasedetail.cfm?ReleaseID=761090.

Festinger, L. (May 1954). "A Theory of Social Comparison Processes." In: *Human Relations* 7.2, pp. 117–140. ISSN: 0018-7267. DOI: 10.1177/001872675400700202.

Fogg, B.J. (2003). *Persuasive Technology: Using Computers to Change What We Think and Do.* Morgan Kaufmann. ISBN: 0080479944.

Folmer, Eelke, Martijn van Welie, and Jan Bosch (Feb. 2006). "Bridging patterns: An approach to bridge gaps between SE and HCI." In: *Information and Software*

Technology 48.2, pp. 69–89. ISSN: 09505849. DOI: `10.1016/j.infsof.2005.02.005`.

Foursquare (2012). "Hi! I want to learn more about Foursquare!" In: *YouTube*. URL: `http://youtu.be/7bcWle-UA-M`.

Francis, Tom (2008). "Murder Incorporated." In: *PC Gamer*. URL: `http://www.computerandvideogames.com/180867/features/murder-incorporated/`.

Fung, Timothy K F (2008). "Banking with a Personalized Touch: Examining the Impact of Website Customization on Commitment." In: *Journal of Electronic Commerce Research* 9.4, pp. 296–309.

Gaider, David (2013). "On Fandom and Toxic Environments." In: *The Bittersweet Thing*. URL: `http://dgaider.tumblr.com/post/39544291251/on-fandom-and-toxic-environments`.

Galperin, Eva (2011). *2011 in Review: Nymwars*. Tech. rep. Electronic Frontier Foundation. URL: `https://www.eff.org/deeplinks/2011/12/2011-review-nymwars`.

Gamma, Erich, Richard Helm, Ralph Johnson, and John M. Vlissides (Nov. 1994). *Design Patterns: Elements of Reusable Object-Oriented Software*. 1st ed. Addison-Wesley Professional. ISBN: 0201633612.

Gittins, J. C. and P. Whittle (1979). "Bandit Processes and Dynamic Allocation Indices / Discussion of Dr. Gittins' Paper." English. In: *Journal of the Royal Statistical Society. Series B (Methodological)* 41.2, pp. 148–177. ISSN: 00359246. URL: `http://www.jstor.org/stable/2985029`.

Gobry, Pascal-Emmanuel (2011). "Boom! CityVille Zooms Past 100 Million Users." In: *Business Insider*. URL: `http://www.businessinsider.com/boom-cityville-zooms-past-100-million-users-2011-1`.

Goetz, Thomas (2011). "Harnessing the Power of Feedback Loops." In: *Wired* 19.07. URL: `http://www.wired.com/magazine/2011/06/ff_feedbackloop/`.

Goffman, Erving (1959). *The Presentation of Self in Everyday Life.* Doubleday, p. 259.

Goldberg, Jackie (2013). *Designing Daily Habits: Personalization and Patterns.* Tech. rep. Sunnyvale, CA, USA: Yahoo!

"Grinding - Giant Bomb." In: *Giant Bomb.* URL: http : / / www . giantbomb . com / grinding/92-250/.

"Grinding - Wikipedia." In: *Wikipedia.* URL: http : / / en . wikipedia . org / wiki / Grinding_(video_gaming).

"Grinding - WoWWiki." In: *WoWWiki.* URL: http://www.wowwiki.com/Grinding.

Hafner, Katie (1997). "The Epic Saga of The Well." In: *Wired.* URL: http : //www . wired.com/wired/archive/5.05/ff_well_pr.html.

Haggbloom, Steven J, Renee Warnick, Jason E Warnick, Vinessa K Jones, Gary L Yarbrough, Tenea M Russell, Chris M Borecky, Reagan McGahhey, John L Iii Powell, Jamie Beavers, and Emmanuelle Monte (2002). "The 100 most eminent psychologists of the 20th century." In: *Review of General Psychology* 6.2, pp. 139–152. ISSN: 10892680. DOI: 10.1037//1089-2680.6.2.139.

Hamari, Juho (2011). "Perspectives from behavioral economics to analyzing game design patterns: loss aversion in social games." In: *Proceedings of the CHI 2011 Social Games Workshop.* ISBN: 9781450302685.

— (2012). "Defining Gamification." In: *Hamari, J.* URL: http://juhohamari. com/post/33158604130/gamification.

Hamari, Juho and Veikko Eranti (2011). "Framework for Designing and Evaluating Game Achievements." In: *DiGRA 2011.* Utrecht, Netherlands.

Hampton, Keith N., Lauren Sessions Goulet, Cameron Marlow, and Lee Rainie (2012). *Why most Facebook users get more than they give.* Tech. rep. Pew Internet Project.

Harrigan, Kevin A., Karen Collins, Michael J. Dixon, and Jonathan Fugelsang (May 2010). "Addictive gameplay." In: *Proceedings of the International Academic*

Conference on the Future of Game Design and Technology (Futureplay '10). New York, New York, USA: ACM Press, p. 127. ISBN: 9781450302357. DOI: `10.1145/1920778.1920796`.

Hecker, Chris (Aug. 2010). "Achievements Considered Harmful?" In: URL: `http://chrishecker.com/Achievements_Considered_Harmful`.

Hergenhahn, B. R. (2001). *An Introduction to the History of Psychology*. 4th ed. Wadsworth / Thomson Learning, p. 650.

Hernandez, Patricia (2012). "This Year's Biggest Shooters Remind Me Why Multiplayer Unlocks Suck." In: *Kotaku*. URL: `http://kotaku.com/5965612/this-years-biggest-shooters-remind-me-why-multiplayer-unlocks-suck`.

Hodapp, Eli (2012). "Zynga Shamelessly Rips Off 'Tiny Tower' With Canadian Release of 'Dream Heights'." In: *TouchArcade*. URL: `http://toucharcade.com/2012/01/24/zynga-shamelessly-rips-off-tiny-tower-with-canadian-release-of-dream-heights/`.

Holiday, Ryan (2013). "How Facebook Gets Away With Being Broken On Purpose." In: *BetaBeat*. URL: `http://betabeat.com/2013/03/how-facebook-gets-away-with-being-broken-on-purpose/`.

Holkins, Jerry and Mike Krahulik (2004). "Green Blackboards (And Other Anomalies)." In: *Penny Arcade*. URL: `http://www.penny-arcade.com/comic/2004/03/19`.

Hsee, Christopher K, Fang Yu, Jiao Zhang, and Yan Zhang (2003). "Medium maximization." In: *Journal of Consumer Research* 30.1, pp. 1–14.

"Humar the Pridelord." In: *WoWWiki*. URL: `http://www.wowwiki.com/Humar_the_Pridelord`.

Hunicke, Robin (2009). "The UX of Game/Play." In: *UX Week 2009*.

"I hate you, archaic and useless Halo save system" (2012). In: *NeoGAF*. URL: `http://www.neogaf.com/forum/showthread.php?t=504209`.

"Issue 6994 - khanacademy - a way to cheat." In: *Khan Academy Issue Tracker*. URL: `https://code.google.com/p/khanacademy/issues/detail?id=6994`.

Jacobs, Melinda (2012). "Click, click, click, click. Zynga and the gamification of clicking." In: *G—A—M—E Games as Art, Media, Entertainment* 1.1. ISSN: 2280-7705. URL: `http://www.gamejournal.it/click-click-click-click-zynga-and-the-gamification-of-clicking/`.

Jakobsson, Mikael (2011). "The Achievement Machine: Understanding Xbox 360 Achievements in Gaming Practices." In: *Game Studies* 11.1. URL: `http://gamestudies.org/1101/articles/jakobsson`.

James, Stan (2009). "The dance of interface and user expectations." In: *wanderingstan*. URL: `http://wanderingstan.com/2009-11-15/the-dance-of-interface-and-user-expectations`.

Järvinen, Aki (2012). "Free to Play, Tricky to Design." In: *Proceedings of the Northern Game Summit*. Kajani, Finland. URL: `http://www.slideshare.net/gameswithoutfrontiers/free-to-play-tricky-to-design`.

Jones, Quentin, Gilad Ravid, and Sheizaf Rafaeli (June 2004). "Information Overload and the Message Dynamics of Online Interaction Spaces: A Theoretical Model and Empirical Exploration." In: *Information Systems Research* 15.2, pp. 194–210. ISSN: 1047-7047. DOI: `10.1287/isre.1040.0023`.

Juul, Jesper (2009). *A Casual Revolution: Reinventing Video Games and Their Players*. MIT Press, p. 252. ISBN: 0262013371. URL: `http://www.jesperjuul.net/casualrevolution/`.

K, R* (2012). "Taking Aim at Cheaters in Max Payne 3." In: *Rockstar Newswire*. URL: `http://www.rockstargames.com/newswire/article/35441/taking-aim-at-cheaters-in-max-payne-3.html`.

Kahneman, Daniel (2011). *Thinking, Fast and Slow*. Farrar, Straus and Giroux, p. 512. ISBN: 1429969350.

Kahneman, Daniel, Jack L Knetsch, and Richard H Thaler (1990). "Experimental Tests of the Endowment Effect and the Coase Theorem." In: *Journal of Political Economy* 98.6. Ed. by Colin F Camerer, George Loewenstein, and Matthew Rabin, pp. 1325–1348. ISSN: 00223808. DOI: `10.1086/261737`.

Kalucki, John (2011a). "John Kalucki's answer to Twitter: On Twitter, what do you call the person being followed?" In: URL: `http://www.quora.com/Twitter-1/On-Twitter-what-do-you-call-the-person-being-followed/answer/John-Kalucki?srid=u5oM\&st=ns`.

— (2011b). "On Twitter, what do you call the person being followed?" In: URL: `http://www.quora.com/Twitter-1/On-Twitter-what-do-you-call-the-person-being-followed/answer/Dennis-Shiao/comment/445855?srid=u5oM\&st=ns`.

Karau, Steven and Kipling Williams (1993). "Social Loafing: A Meta-Analytic Review and Theoretical Integration." In: *Journal of Personality and Social Psychology* 65.4, pp. 681 –706.

Katz, Michael L. and Carl Shapiro. "Systems Competition and Network Effects." In: *Journal of Economic Perspectives* 8.2, pp. 93–115. URL: `http://ideas.repec.org/a/aea/jecper/v8y1994i2p93-115.html`.

Kelling, George L. and James Q. Wilson (1982). "Broken Windows." In: *The Atlantic*.

Kelly, Tadhg (2012). "Everything You'll Ever Need To Know About Gamification." In: *TechCrunch*. URL: `http://techcrunch.com/2012/11/17/everything-youll-ever-need-to-know-about-gamification/`.

Kincaid, Jason (2009). "Facebook Makes Baby Steps Towards Its Twitter-Like 'Follow' Feature." In: *TechCrunch*. URL: `http://techcrunch.com/2009/07/25/facebook-makes-baby-steps-towards-its-twitter-like-follow-feature/`.

Kivetz, Ran, Oleg Urminsky, and Yuhuang Zheng (Feb. 2006). "The Goal-Gradient Hypothesis Resurrected: Purchase Acceleration, Illusionary Goal Progress, and Customer Retention." In: *Journal of Marketing Research* 43.1, pp. 39–58. ISSN: 0022-2437. DOI: `10.1509/jmkr.43.1.39`.

Knowles, Jamillah (2012). "Foursquare updates its Android app to bring back points and add categorization for venue images." In: *The Next Web*. URL: `http://thenextweb.com/apps/2012/10/10/foursquare-updates-android-app-to-bring-back-points-and-add-categorization-for-venue-images/`.

Koenig, Andrew (1998). "Patterns and Antipatterns." In: *The Patterns Handbook: Techniques, Strategies and Applications*, pp. 383–390.

Kohn, Alfie (1999). *Punished by Rewards: The Trouble with Gold Stars, Incentive Plans, A's, Praise, and Other Bribes*. Mariner Books; 2 edition, p. 416. ISBN: 0618001816.

Kosinski, M., D. Stillwell, and T. Graepel (Mar. 2013). "Private traits and attributes are predictable from digital records of human behavior." In: *Proceedings of the National Academy of Sciences*, pp. 1218772110–. ISSN: 0027-8424. DOI: `10.1073/pnas.1218772110`.

Koster, Raph (2003). "Small Worlds." In: *Game Developers Conference*.

— (Nov. 2004). *A Theory of Fun for Game Design*. 1st ed. Paraglyph Press. ISBN: 1932111972.

Kottke, Jason (2008). "Does the broken windows theory hold online?" In: *kottke.org*. URL: `http://kottke.org/08/12/does-the-broken-windows-theory-hold-online`.

Krug, Steve (2009). *Don't Make Me Think: A Common Sense Approach to Web Usability (2nd Edition) (Voices That Matter)*. New Riders.

Kuchera, Ben (2012). "Double Fine stumbles with Middle Manager of Justice, a free-to-play game that uses time against you." In: *The PA Report*. URL: `http://`

penny-arcade.com/report/editorial-article/double-fine-stumbles-with-middle-manager-of-justice-a-free-to-play-game-tha.

Lakoff, George (1987). *Women, fire, and dangerous things: what categories reveal about the mind*. University of Chicago Press, p. 614. ISBN: 0226468046.

Lazzaro, Nicole (2004). *Why We Play Games: Four Keys to More Emotion Without Story*. Tech. rep. XEODesign.

LeBlanc, Marc (2009). "8 Kinds of Fun." In: *8KindsOfFun.com*.

Lepper, Mark R, David Greene, and Richard E Nisbett (1973). "Undermining children's intrinsic interest with extrinsic reward: A test of the "overjustification" hypothesis." In: *Journal of Personality and Social Psychology* 28.1, pp. 129–137. ISSN: 00223514. DOI: 10.1037/h0035519.

Levy, Steven (2012). "Has Google Popped the Filter Bubble?" In: *Wired*. URL: http://www.wired.com/business/2012/01/google-filter-bubble/.

Lewis, Chris, Noah Wardrip-Fruin, and Jim Whitehead (May 2012). "Motivational game design patterns of 'ville games." In: *Proceedings of the International Conference on the Foundations of Digital Games (FDG '12)*. New York, New York, USA: ACM Press, p. 172. ISBN: 9781450313339. DOI: 10.1145/2282338.2282373.

"List of countries by population." In: *Wikipedia*. URL: http://en.wikipedia.org/wiki/List_of_countries_by_population.

Liszkiewicz, A. J. Patrick (2010). "Cultivated Play: Farmville." In: *berfrois*. URL: http://www.berfrois.com/2010/10/cultivated-play-farmville/.

Longo, Matthew R and Stella F Lourenco (Apr. 2007). "Spatial attention and the mental number line: evidence for characteristic biases and compression." In: *Neuropsychologia* 45.7, pp. 1400–1407. ISSN: 0028-3932. DOI: 10.1016/j.neuropsychologia.2006.11.002.

Lotufo, Rafael, Leonardo Passos, and Krzysztof Czarnecki (June 2012). "Towards improving bug tracking systems with game mechanisms." In: *2012 9th IEEE*

Working Conference on Mining Software Repositories (MSR). IEEE, pp. 2–11. ISBN: 978-1-4673-1761-0. DOI: 10.1109/MSR.2012.6224293.

Lowenstein, George and Peter Ubel (2010). "Economics Behaving Badly." In: *New York Times*.

Luban, Pascal (2012). "Designing Freemium Titles for Hardcore Gamers." In: *Gamasutra*. URL: http://www.gamasutra.com/view/feature/183307/designing_freemium_titles_for_.php.

Ludford, Pamela J., Dan Cosley, Dan Frankowski, and Loren Terveen (Apr. 2004). "Think Different." In: *Proceedings of the 2004 Conference on Human Factors in Computing Systems (CHI '04)*. New York, New York, USA: ACM Press, pp. 631–638. ISBN: 1581137028. DOI: 10.1145/985692.985772.

Lundgren, Sus and Staffan Björk (May 2012). "Neither playing nor gaming." In: *Proceedings of the International Conference on the Foundations of Digital Games - FDG '12*. New York, New York, USA: ACM Press, p. 113. ISBN: 9781450313339. DOI: 10.1145/2282338.2282363.

Lynley, Matthew (2013). "Homies and Besties: Searching With Slang Reveals Power of Facebook's Graph Search." In: *The Wall Street Journal*. URL: http://blogs.wsj.com/digits/2013/02/20/homies-and-besties-searching-with-slang-reveals-power-of-facebooks-graph-search/?mod=.

Lyons, Dan (2012). "Mark Cuban: Facebook Is Driving Away Brands - Starting With Mine." In: *ReadWrite*. URL: http://readwrite.com/2012/11/13/mark-cuban-facebooks-sponsored-posts-are-driving-away-brands.

Madrigal, Alexis (2011). "Why Facebook and Google's Concept of 'Real Names' Is Revolutionary." In: *The Atlantic*. URL: http://www.theatlantic.com/technology/archive/2011/08/why-facebook-and-googles-concept-of-real-names-is-revolutionary/243171/.

Madrigan, Jamie (May 2011). "The Psychology Of Microsoft Points." In: *Gamasutra*. URL: `http : / / www . gamasutra . com / view / news / 34589 / Analysis _ The _ Psychology_Of_Microsoft_Points.php`.

— (2012). "The Psychology Of... High Scores." In: *Edge*, pp. 94–9.

Malone, T W and M R Lepper (1987). "Making learning fun: A taxonomy of intrinsic motivations for learning." In: *Aptitude learning and instruction*. Ed. by Richard E Snow and Marshall J Farr. Vol. 3. Aptitude, learning, and instruction: Volume III Conative and affective process analyses 3. Lawrence Erlbaum Associates. Chap. Vol. 3, pp. 223–253. ISBN: 0898597218.

Malone, TW (1981). "Toward a theory of intrinsically motivating instruction." In: *Cognitive science* 4. URL: `http://www.sciencedirect.com/science/article/pii/S0364021381800171`.

Mamykina, Lena, Bella Manoim, Manas Mittal, George Hripcsak, and Björn Hartmann (2011). "Design Lessons from the Fastest Q & A Site in the West." In: *Proceedings of the 2011 Annual Conference on Human Factors in Computing Systems (CHI '11)*. CHI '11. ACM Press, pp. 2857–2866. ISBN: 9781450302289. DOI: `10.1145/1978942.1979366`. URL: `http://portal.acm.org/citation.cfm?doid=1978942.1979366`.

Markowitz, Harry (1952). "The Utility of Wealth." In: *The Journal of Political Economy*, pp. 151–158.

Markus, M. L. (Oct. 1987). "Toward a "Critical Mass" Theory of Interactive Media: Universal Access, Interdependence and Diffusion." In: *Communication Research* 14.5, pp. 491–511. ISSN: 0093-6502. DOI: `10.1177/009365087014005003`.

Maslow, Abraham H (1943). "A Theory of Human Motivation." In: *Psychological Review* 50.4, pp. 370–396.

— (1970). *Motivation and Personality*. Harper & Row.

Mateas, Michael (Jan. 2005). "Procedural literacy: educating the new media practitioner." In: *On the Horizon* 13.2, pp. 101–111. ISSN: 1074-8121. DOI: `10.1108/10748120510608133`.

Mattheij, Jacques (2011). "The Unofficial Hacker News FAQ." In: *Jacques Mattheij*. URL: `http://jacquesmattheij.com/The+Unofficial+HN+FAQ`.

McCartney, Scott (2012). "$100 to Fly Through the Airport." In: *The Wall Street Journal*. URL: `http://goo.gl/JzNUZ`.

McGonigal, Jane (2011). *Reality Is Broken: Why Games Make Us Better and How They Can Change the World*. Penguin Press, p. 400. ISBN: 1594202850.

McLean, Ryan (2008). "My Messy Desktop." In: *Flickr*. URL: `http://www.flickr.com/photos/saaby/2179348509/`.

Medler, Ben (2011). "Player Dossiers: Analyzing Gameplay Data as a Reward." In: *Game Studies* 11.1. URL: `http://gamestudies.org/1101/articles/medler`.

Meisler, Bernard (2012). "Why Are Dead People Liking Stuff On Facebook?" In: *ReadWrite*. URL: `http://readwrite.com/2012/12/11/why-are-dead-people-liking-stuff-on-facebook`.

Miller, Patrick (2012). "When does effective free-to-play design become an ethical matter?" In: *Gamasutra*. URL: `http://www.gamasutra.com/view/news/179264/When_does_effective_freetoplay_design_become_an_ethical_matter.php`.

Milton, Ohmer, Howard R. Pollio, and James A. Eison (1986). *Making sense of college grades: why the grading system does not work and what can be done about it*. Jossey-Bass Publishers, p. 287. ISBN: 0875896871.

Moyer, R S and T K Landauer (1967). "Time required for judgements of numerical inequality." In: *Nature* 215.5109, pp. 1519–1520. URL: `http://www.nature.com/nature/journal/v215/n5109/abs/2151519a0.html`.

Navarick, Doug (2002). *Behavioral vs Cognitive Views of Learning*. Tech. rep. California State University, Fullerton.

Neuberg, Matt. "Mountain Lion is (Still) a Quitter." In: *TidBITS*. URL: http://tidbits.com/article/13174.

— (2011). "TidBITS Lion Is a Quitter." In: *TidBITS*. URL: http://tidbits.com/article/12398.

Nielsen, Jakob (2006). *Participation Inequality: Encouraging More Users to Contribute*. Tech. rep. Nielsen Norman Group. URL: http://www.nngroup.com/articles/participation-inequality/.

Olanoff, Drew (2013). "For The First Time, Facebook's 680M Mobile DAUs Surpassed Desktop DAUs — Zuck Says It's Now A "Mobile Company"." In: *TechCrunch*. URL: http://techcrunch.com/2013/01/30/for-the-first-time-facebook-mobile-daus-exceeded-web-daus-in-q4-2012/.

Oliver, Pamela E and Gerald Marwell (1988). "The Paradox of Group Size in Collective Action: A Theory of the Critical Mass. II." In: *American Sociological Review* 53.1, pp. 1–8. ISSN: 00031224. DOI: 10.2307/2095728.

Pariser, Eli (2011). *The Filter Bubble: How the New Personalized Web Is Changing What We Read and How We Think*. Penguin Group, p. 304. ISBN: 1101515120.

Parramore, Lynn (2010). "Eli Pariser on the future of the Internet." In: *Salon*. URL: http://www.salon.com/2010/10/08/lynn_parramore_eli_pariser/.

Pereira, Chris (2012). "Will Moderation Render Wii U's Miiverse Ineffectual?" In: URL: http://www.1up.com/news/wii-u-miiverse-moderation.

Pink, Daniel H. (2011). *Drive: The Surprising Truth About What Motivates Us*. Penguin Group US, p. 272. ISBN: 1101524383.

Porter, Joshua (2010). *Designing for the Social Web*. Peachpit Press, p. 192. ISBN: 013208953X.

Portes, Alejandro (1998). "Social Capital: Its Origins and Applications in Modern Sociology." In: *Annual Review of Sociology* 24.1. Ed. by Eric L Lesser, pp. 1–24. ISSN: 03600572. DOI: 10.1146/annurev.soc.24.1.1. URL: http://www.annualreviews.org/doi/abs/10.1146/annurev.soc.24.1.1.

Preece, Jenny (Mar. 1998). "Empathic communities: reaching out across the Web." In: *Interactions* 5.2, pp. 32–43. ISSN: 10725520. DOI: 10.1145/274430.274435.

Prelec, D. and G. Loewenstein (July 1991). "Decision Making Over Time and Under Uncertainty: A Common Approach." In: *Management Science* 37.7, pp. 770–786. ISSN: 0025-1909. DOI: 10.1287/mnsc.37.7.770.

Radoff, Jon (2011a). *Game On: Energize Your Business with Social Media Games*. John Wiley & Sons, p. 432. ISBN: 1118089308.

— (2011b). "Gamification, Behaviorism and Bullshit." In: *Jon Radoff's Internet Wonderland*. URL: http://radoff.com/blog/2011/08/09/gamification-behaviorism-bullshit/.

Recreating an Activity. Tech. rep. Google. URL: http://developer.android.com/training/basics/activity-lifecycle/recreating.html.

Reeve, Johnmarshall (2005). *Understanding Motivation and Emotion*. Vol. 3. John Wiley & Sons, p. 556. ISBN: 0470392231.

Reiss, Steven (2002). *Who Am I?: 16 Basic Desires that Motivate Our Actions Define Our Persona*. Penguin, p. 280. ISBN: 0425183408.

— (2004). "Multifaceted Nature of Intrinsic Motivation: The Theory of 16 Basic Desires." In: *Review of General Psychology* 8.3, pp. 179–193. ISSN: 10892680. DOI: 10.1037/1089-2680.8.3.179.

— (Jan. 2005). "Extrinsic and intrinsic motivation at 30: Unresolved scientific issues." In: *The Behavior Analyst* 28.1, pp. 1–14. ISSN: 0738-6729.

— (2008). *The Normal Personality: a new way of thinking about people*. Cambridge University Press, p. 201. ISBN: 0521881064.

Rigby, S (2004). *Player Motivational Analysis: A model for applied research into the motivational dynamics of virtual worlds.* Tech. rep. Rochester, NY: Motivation Research Group, University of Rochester.

Rigby, Scott and Richard Ryan (2011). *Glued to Games: How Video Games Draw Us In and Hold Us Spellbound.* ABC-CLIO, p. 196. ISBN: 0313362254.

Ryan, Richard, Valerie Mims, and Richard Koestner (1983). "Relation of reward contingency and interpersonal context to intrinsic motivation: A review and test using cognitive evaluation theory." In: *Journal of Personality and Social Psychology* 45.4, pp. 736 –750.

Ryan, Richard M and Edward L Deci (2004). "Overview of Self-Determination Theory: An Organismic Dialectical Perspective." In: *Handbook of Self-Determination Research.* Ed. by Richard M Ryan and Edward L Deci. University Rochester Press. ISBN: 1580461565.

Ryan, Richard M, C Scott Rigby, and Andrew Przybylski (2006). "The Motivational Pull of Video Games: A Self-Determination Theory Approach." In: *Motivation and Emotion* 30.4, pp. 344–360. ISSN: 01467239. DOI: 10.1007/s11031-006-9051-8.

Sadun, Erica (2012). "Local notification spam: Devs, please don't do that." In: *The Unofficial Apple Weblog.* URL: http://www.tuaw.com/2012/07/16/local-notification-spam-devs-please-dont-do-that/.

Schell, Jesse (Aug. 2008). *The Art of Game Design: A book of lenses.* Morgan Kaufmann. ISBN: 0123694965.

— (2011). *When Games Invade Real Life.* Tech. rep. URL: http://www.slideshare.net/jesseschell/when-games-invade-real-life.

Schiffman, Noah and Suzanne Greist-Bousquet (1992). "The effect of task interruption and closure on perceived duration." In: *Bulletin of the Psychonomic Society* 30.1, pp. 9–11.

Schlack, Julie Whittes (2011). *The "64% Rule:" What Real Customer Engagement Looks Like*. Tech. rep. Communispace.

Schwartz, Barry (2004). "The Tyranny of Choice." In: *Scientific American Mind* 44.29.

Scotto (2012). "Hey, Remember That Weird Handheld Neo-Geo Thing? It Actually Exists." In: *Giant Bomb*. URL: `http://www.giantbomb.com/articles/hey-remember-that-weird-handheld-neo-geo-thing-it-/1100-4038/`.

Singer, Marcus G. (Oct. 1963). "The Golden Rule." In: *Philosophy* 38.146, pp. 293–314. ISSN: 1469-817X.

Smith, Kevin (2012). "Foursquare Is Taking On Yelp Head-On In Mobile." In: *Business Insider*. URL: `http://www.businessinsider.com/foursquare-taking-on-yelp-2012-11`.

Social Media Report 2012 (2012). Tech. rep. Nielsen. URL: `http://blog.nielsen.com/nielsenwire/social/2012/`.

Sorens, Neil (2007). "Rethinking the MMO." In: *Gamasutra*. URL: `http://www.gamasutra.com/view/feature/1583/rethinking_the_mmo.php`.

Speiser, Mike (Sept. 2009). "What We Can Learn About Pricing From Menu Engineers." In: *GigaOm*. URL: `http://goo.gl/4HrJE`.

State of the Appnation – A Year of Change and Growth in U.S. Smartphones. Tech. rep. Nielsen. URL: `http://blog.nielsen.com/nielsenwire/online_mobile/state-of-the-appnation--a-year-of-change-and-growth-in-u-s-smartphones/`.

State Preservation and Restoration (2013). Tech. rep. Apple. URL: `http://goo.gl/X6oaD`.

Sterling, Jim (2009). "Heavy Rain has no replay value (but that's a good thing?)" In: *Destructoid*. URL: `http://www.destructoid.com/heavy-rain-has-no-replay-value-but-that-s-a-good-thing--146940.phtml`.

Sternberg, Rober Jeffrey (2009). *Cognitive Psychology*. Cengage Learning, p. 657.

Sternberg, Robert J., Jeff Mio, and Jeffery Scott Mio (2009). *Cognitive Psychology*. Cengage Learning, p. 657. ISBN: 049550629X.

Street, Greg. "Ghostcrawler: "Blizzard doesn't listen to..."" In: *Twitter*. URL: `https://twitter.com/Ghostcrawler/status/280032566612987905`.

Suler, John (2004). "The Online Disinhibition Effect." In: *Cyberpsychology and Behavior* 7.3, pp. 321–326.

Sullivan, Anne (2012). "The Grail Framework: Making Stories Playable on Three Levels in RPGs." PhD thesis. University of California, Santa Cruz, p. 243.

Suvorov, Anton (2003). "Addiction to Rewards." In: *Mimeo* October, pp. 1–47.

Sweeny, Patrick J (2012). "Is Foursquare's Bubble About To Burst?" In: *Forbes*. URL: `http://www.forbes.com/sites/ciocentral/2012/11/23/is-foursquares-bubble-about-to-burst/`.

Thackara, John (May 2001). "The design challenge of pervasive computing." In: *Interactions* 8.3, pp. 46–52. ISSN: 10725520. DOI: `10.1145/369825.369832`.

Thaler, Richard (1994). *The Winner's Curse: Paradoxes and Anomalies of Economic Life*. Princeton University Press, p. 240. ISBN: 0691019347.

"The Core App Design" (2012). In: URL: `http://developer.apple.com/library/mac/\#documentation/General/Conceptual/MOSXAppProgrammingGuide/CoreAppDesign/CoreAppDesign.html`.

Thompson, Clive (Aug. 2007). "Halo 3: How Microsoft Labs Invented a New Science of Play." In: URL: `http://www.wired.com/gaming/virtualworlds/magazine/15-09/ff\_halo?currentPage=all`.

— (2009). "Clive Thompson on the Taming of Comment Trolls." In: *Wired* 17.04.

Tidwell, Jenifer (2010). *Designing Interfaces*. O'Reilly Media, Inc., p. 576. ISBN: 1449302734.

"Tiny Tower." In: *Metacritic*. URL: `http://www.metacritic.com/game/ios/tiny-tower`.

Tokunaga, Robert S (Jan. 2011). "Friend me or you'll strain us: understanding negative events that occur over social networking sites." In: *Cyberpsychology, behavior and social networking* 14.7-8, pp. 425–32. ISSN: 2152-2723. DOI: 10.1089/cyber.2010.0140. URL: http://online.liebertpub.com/doi/abs/10.1089/cyber.2010.0140?journalCode=cyber.

Topolsky, Joshua (2013). "I used Google Glass: the future, but with monthly updates." In: *The Verge.* URL: http://www.theverge.com/2013/2/22/4013406/i-used-google-glass-its-the-future-with-monthly-updates.

Toxboe, Anders (2011). *Persuasive Patterns.* URL: http://persuasive-patterns.com/.

Turkle, Sherry (1985). *The Second Self: Computers and the Human Spirit.* Simon & Schuster, p. 362. ISBN: 0671606026.

Walsh, James (1998). *You Can't Cheat an Honest Man: How Ponzi Schemes and Pyramid Frauds Work... And Why They're More Common Than Ever.* Silver Lake Publishing, p. 354. ISBN: 1563431696.

Wardrip-Fruin, Noah (2012). *Technology in Education: How Will it Change the Game?* Tech. rep. The Churchill Club. URL: http://goo.gl/FNbsW.

Watson, John Broadus (1919). *Psychology from the standpoint of a behaviorist.* Lippincott, p. 429.

Weber, Bethany J and Gretchen B Chapman (2005). "Playing for peanuts: Why is risk seeking more common for low-stakes gambles?" In: *Organizational Behavior and Human Decision Processes* 97.1, pp. 31–46. ISSN: 07495978. DOI: 10.1016/j.obhdp.2005.03.001. URL: http://linkinghub.elsevier.com/retrieve/pii/S0749597805000361.

Weber, Richard (1992). "On the Gittins Index for Multiarmed Bandits." In: *The Annals of Applied Probability* 2.4, pp. 1024–1033. ISSN: 10505164. DOI: 10.1214/aoap/1177005588.

"What are energy points?" In: *Khan Academy*. URL: `http://khanacademy.desk.com/customer/portal/articles/936549-what-are-energy-points-`.

Whittaker, Steve, Ofer Bergman, and Paul Clough (2010). "Easy on that trigger dad: a study of long term family photo retrieval." English. In: *Personal and Ubiquitous Computing* 14.1, pp. 31–43. ISSN: 1617-4909. DOI: `10.1007/s00779-009-0218-7`.

"Who is harmed by a "Real Names" policy?" In: *Geek Feminism Wiki*. URL: `http://geekfeminism.wikia.com/wiki/Who_is_harmed_by_a_"Real_Names"_policy?`.

Wigdor, Daniel, Sarah Williams, Michael Cronin, Robert Levy, Katie White, Maxim Mazeev, and Hrvoje Benko (Oct. 2009). "Ripples: utilizing per-contact visualizations to improve user interaction with touch displays." In: *Proceedings of the 22nd annual ACM symposium on User interface software and technology (UIST '09)*. New York, New York, USA: ACM Press, p. 3. ISBN: 9781605587455. DOI: `10.1145/1622176.1622180`.

Wittgenstein, Ludwig (1953). *Philosopical Investigations*.

Wohn, Donghee Yvette, Cliff Lampe, Rick Wash, Nicole Ellison, and Jessica Vitak (Jan. 2011). "The "S" in Social Network Games: Initiating, Maintaining, and Enhancing Relationships." In: *2011 44th Hawaii International Conference on System Sciences*. IEEE, pp. 1–10. ISBN: 978-1-4244-9618-1. DOI: `10.1109/HICSS.2011.400`.

"Woody Harrelson Reddit AMA" (2012). In: *Know Your Meme*. URL: `http://knowyourmeme.com/memes/events/woody-harrelson-reddit-ama`.

Wroblewski, Luke (2008). "SxSW 2008: Filching Design." In: *LukeW*. URL: `http://www.lukew.com/ff/entry.asp?658`.

Yee, Nick (2006). "Motivations for Play in Online Games." In: *CyberPsychology & Behavior* 9.6, pp. 772–775.

Yelp Elite Squad. URL: http://www.yelp.com/elite.

Zagal, José P, ed. (2012). *The Videogame Ethics Reader*. 1st. Cognella Academic Publishing, p. 328.

Zagal, José P, Staffan Björk, and Chris Lewis (2013). "Dark Patterns in the Design of Games." In: *Proceedings of the International Conference on the Foundations of Digital Games (FDG '13)*.

Zagal, Jose P, Michael Mateas, Clara Fernández-Vara, Brian Hochhalter, and Nolan Lichti (2005). "Towards an Ontological Language for Game Analysis." In: *DiGRA 2005*. Vol. 2005. Citeseer, pp. 3–14.

Zichermann, Gabe and Christopher Cunningham (2011). *Gamification by Design: Implementing Game Mechanics in Web and Mobile Apps*. O'Reilly Media, Inc., p. 208. ISBN: 1449315399.