Crafting in Games

April Grow <agrow_at_ucsc_dot_edu>, University of California Santa Cruz
Melanie Dickinson <mldickin_at_ucsc_dot_edu>, University of California Santa Cruz
Johnathan Pagnutti <jpagnutt_at_ucsc_dot_edu>, University of California Santa Cruz
Noah Wardrip-Fruin <nwf_at_soe_dot_ucsc_dot_edu>, University of California Santa Cruz
Michael Mateas <michaelm_at_soe_dot_ucsc_dot_edu>, University of California Santa Cruz

Abstract

This paper analyzes the design space for crafting systems within games. A crafting system is the collection of game mechanics which enable a player to create virtual objects within a game, with examples ranging from making a pickaxe or placing blocks in *Minecraft*, to combining ingredients to create items in *World of Warcraft*, or designing a city in *SimCity*. Within both game design and game journalism there is no systematic understanding of what crafting means in games — rather, the term is used in an informal "I'll know it when I see it" way, with different ideas of what constitutes "crafting" becoming conventional for different game genres. We aim to create a more systematic analysis and language for virtual crafting, using a design space approach where we identify seven features of crafting systems; any given crafting system is understood in terms of the degree and manner in which it exemplifies these features. The features we identify are: recipe definition (how well-defined are the player steps for creating a virtual object), fidelity of action (how detailed is the player-performed enactment of crafting), completion constraints (how is creation constrained by virtual "resources"), variable outcome (how much can the result vary when the player performs the same crafting actions), system recognition of outcome (to what degree do other game systems "understand" and take account of what a player has made), player expressiveness (how big is the creative space provided to the player), and progression (how do the possibilities of crafting change over time). We apply these dimensions to the analysis of 64 crafting systems across 47 example games. This more systematic analysis of the crafting system design space provides a deeper understanding of how games function as a creative and artistic medium for players, and provides a language and framework enabling scholars, critics and developers to more deeply understand existing crafting systems as well as unexplored opportunities for future crafting systems.

Introduction

Crafting in games affords a unique kind of interactive experience, only possible in playable media, and made increasingly rich with the complex computational modeling of videogames[1]. It enables players to create items and structures without the cost, effort, and destructive modification of raw materials in the real world. It provides spaces of playful experimentation and creative expression, wherein ordinary people can design and create whole buildings, cities, geographies, machines, rocketships, gardens, and creatures, without a difficult-to-acquire formal education, specialized knowledge, or extraordinary resources. Games provide a context of playfulness that removes the inhibiting pressure to make something perfect or to fulfill a practical purpose. Crafting in games often supports creativity as intrinsically pleasurable, rather than as a commodified, extrinsically-motivated way to accomplish tasks, much like the casual creators genre identified by Compton and Mateas [Compton 2015]. It also means that people can create things that would be altogether impossible in the real world. These creations then exist in the context of a game world; what players craft in a game becomes part of the game, taking on a new life, whether becoming part of a graphically modeled, physically simulated landscape, or an item that the player-character can use, wear, trade, or consume. This is all to say that crafting in games, while sharing much in common with real-world crafting or digitally augmented crafting[2], is also made significantly different by its context in playable systems.

Crafting is a creative and pleasurable alternative gaming experience — thus often speaking to different audiences than playable media centered around combat systems or resource management. While players can be creative and express themselves through game actions that do not involve creating [Wright et al. 2002], creativity that results in the existence of something new supports a creative pleasure not obtainable otherwise. *SimCity*, *The Sims*, and *Minecraft*, while vastly different in subject matter (urban planning, house design and decoration, and block placement, respectively), are all underwritten by player creation in virtual worlds, and have all been highly influential, popular-culture phenomena in their times. Much of their connected successes are in the novel experiences they enable, all primarily driven by crafting. These pioneering, crafting-focused games have transformed the playable media landscape, inspiring entirely new genres of simulation and open-world, sandbox games [Ondrejka 2004] [Rose 2014]. A new, more diverse audience of players has also been drawn to these experiences, noteworthy in the field's straight, white, male-dominated history and tenacious reputation [Kafai et al. 2008] [Shaw 2015]. *SimCity* and *The Sims* drew mainstream media headlines for their equal gender appeal [Paulk 2006] [Thompson 2003], and *Minecraft* has been deployed in diverse classrooms for experimental creation-based learning [Smaldone et al. 2016].

2

Making, creating, building — the language typically used to describe these experiences — are vague descriptions for the complex fields of creativity and design. We need to build a finer critical vocabulary around these systems to more effectively discuss them. Additionally, crafting systems are often deeply integrated into other game systems and mechanics, or operations of the game: working as the production side to a rich in-game economy, or as vehicles for player or game progression. Understanding the kinds of play crafting affords the player, or where a crafting system could fit into a larger game's context, is currently under-explored in game studies.

3

To build this understanding, we construct a taxonomy of crafting in games. Taxonomies are a common tool in game studies, often used to discuss either types of players [Bartle 1996] or games and simulations themselves [Klabbers 2003]. We take the latter, game-centric approach. Taxonomies create a shared vocabulary around complicated concepts to allow a deeper and better-organized study of those concepts, which is the aim of this article.

4

When a player performs actions similar to those of a real-life crafter, or goes about creating a virtual object using the mechanics, the system of those mechanics may be colloquially known as a crafting system by the game's designers and/or players. "Crafting" is advertised as a feature of games from *World of Warcraft* to *Minecraft* (and is even included in some of their names). However, games implement their "crafting" vastly differently; in *World of Warcraft*, crafting is highly abstracted, conducted through menus, and linked into a vast player economy of items, goods and services. Crafting in *Minecraft* is far closer to playing with LEGOs, as *Minecraft* structures are built by players placing virtual blocks on one another. These creations range from giant masterpieces to bare-bones boxes that let players survive the dangerous night. This type of *Minecraft* crafting is quite different even from several other crafting systems within *Minecraft* where players use a dedicated set of crafting menus to instantaneously create and place objects into the player's inventory. A third contrasting example would be *Kittens Game*, a text-only browser idle[3], or incremental, game which includes evolving recipes[4] and large-scale automation. A quick look at these three games shows three very different forms of design and creation within game crafting systems.

5

With these diverse examples in mind, we might roughly say that crafting in games is the thoughtful manipulation of materials by the player to create something else within the context of the game.

6

We aim to break down the above, very broad definition and focus on seven different facets of crafting systems in games: Recipe Definition, Fidelity of Action, Completion Constraints, Variable Outcome, Recognition of Outcome, Player Expressiveness, and Progression.
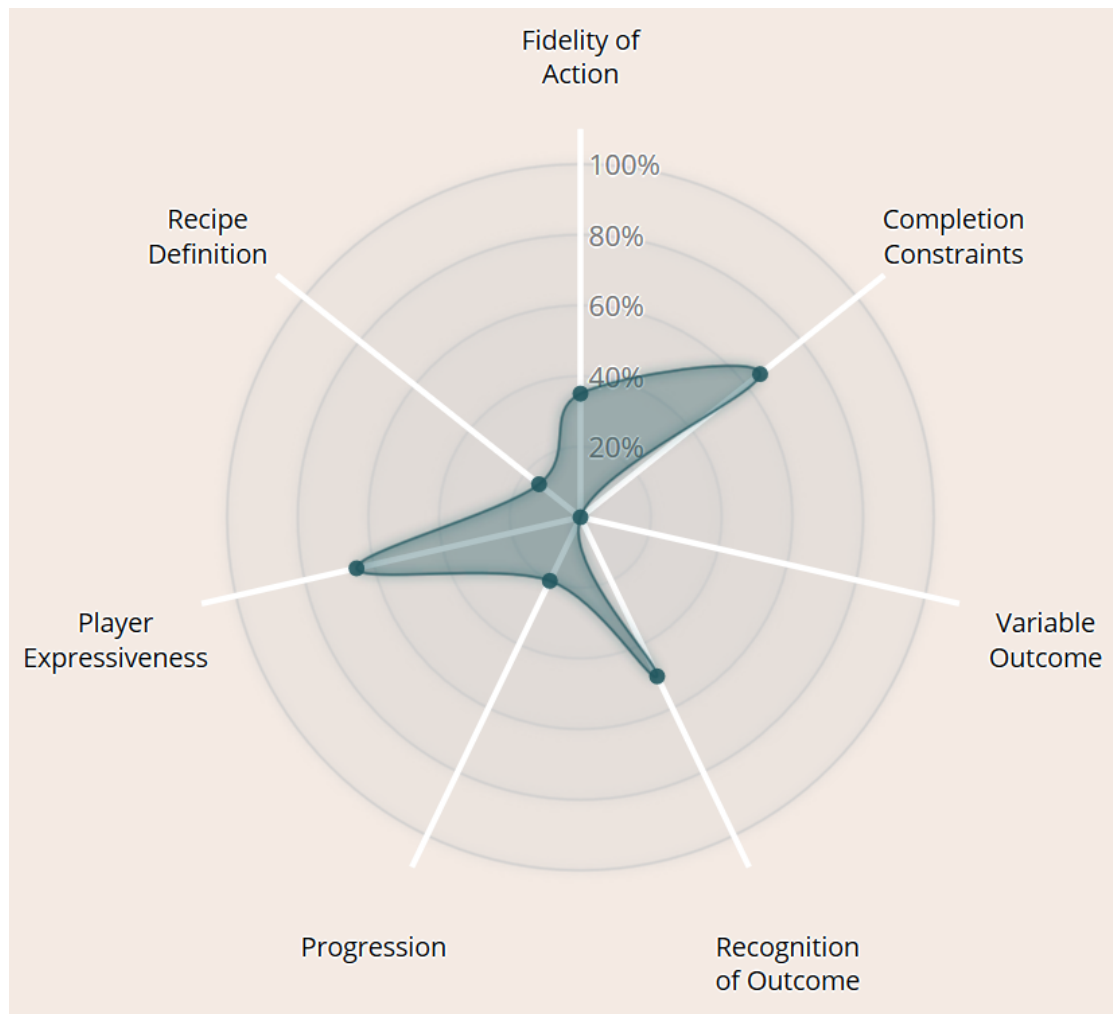
7

Figure 1. A radial graph showing all seven dimensions of crafting systems as well as an example system: House Decoration from *Animal Crossing*. The dimensions are: Recipe Definition, Fidelity of Action, Completion Constraints, Variable Outcome, Recognition of Outcome, Player Expressiveness, and Progression. All 47 games analyzed for this paper and for the accompanying visualization were given scores on each dimension.

To generate these dimensions, we started by carefully reviewing a selection of games that have clear instances of player creation. We looked at the games themselves, as well as secondary literature around these games. Because crafting is tied strongly to the real world, we also reviewed literature around non-virtual crafting. Our taxonomy is built from concepts that have been identified in other games studies work, as well as important concepts in the study of real-world crafting.

After this initial review, we struggled to relate the many and often disparate features of crafting in games. We could not find a "canonical" crafting game, nor did we find it effective to try to describe a set of categories that neatly discretizes the space into mutually-exclusive groups of games or crafting systems. We instead identified core attributes of crafting which different crafting systems had to varying degrees. The attributes were orthogonal and nonhierarchical; having one attribute didn't require or preclude a game from having any other. Drawing inspiration from past work in game studies by Jesper Juul, we created a seven-dimensional model of crafting systems that described meaningful mechanical differences within the games [Juul 2005]. This inclusive taxonomy approach, involving placing terms at an intersection of orthogonal, scalar dimensions, has proven illuminating and influential in game studies and is further discussed below. After we first drafted our dimensions, largely inspired by our survey of prototypical crafting games and literature on real-world crafting, we set about formally encoding specific games in the taxonomy one-by-one (or assigning values for them on each dimension). This process revealed some ambiguities and redundancy in the taxonomy, which we then fixed. We refined our taxonomy in this way over the course of about three months, adding to our set of sample games, discussing and plotting them on the dimensions, editing — and sometimes removing — dimensions, and repeating the

plotting process over again for all games collected thus far. We stopped when we no longer needed to refine the taxonomy to adequately capture new sample games, and when our set of sample games was sufficiently large and diverse, even capturing edge-case crafting systems. A full report of every crafting system we evaluated, their rankings on each dimension, and a brief description with justifications accompanies this document as a stand-alone visualization website, which can be found here: users.soe.ucsc.edu/~agrow/craftsystems[5]. Github access to the visualization website, minus the additional gifs, can also be found here: github.com/agrow/craftsystems.

This paper examines the role of crafting systems in games and analyzes seven design dimensions of crafting mechanics. First, we'll examine some related work to give a perspective on various frameworks in use in game studies, and then take a brief historical tour of real-life crafting and design literature. Then we'll break down each crafting dimension in detail, going over what the dimension scores represent and giving some illustrative example games at different positions along the dimension. Next, we will mention important external forces involved with a crafting system, such as social play and player communities, and how those forces interact with the design dimensions. Finally, we'll wrap up the taxonomy and invite future discussion on the topic. | 10

Related Work & Background Concepts

Game Studies

The work presented in this paper falls under the realm of game studies. As previously mentioned, the core organizational method we use stems from Jesper Juul and previous work on taxonomies as applied to game studies. | 11

Taxonomies in Game Studies

Taxonomies are a common tool in game studies as the medium has rapidly been expanding. Ellington, along with his co-authors [Ellington et al. 1982], breaks games up into three pure types (pure games, pure simulations, and pure case studies), and four hybrids (simulation games, simulated case studies, games used as case studies, and simulation games used as case studies). | 12

Callois uses a two-dimensional system to classify games [Callois 1961]. The first one is built around four categories: Agon, Alea, Mimicry and Ilinx. Agon refers competition and balance, with all players having an equal chance of success. Alea refers to luck, with players having little control over the game's outcome. Mimicry is about pretending to be someone else, and Ilinx refers to the feeling of vertigo and attempting to disrupt normal sensory patterns. Callois' second dimension is focused on the rule base: Piada, freedom for improvisation, is opposed to Ludus, rules and conventions. | 13

Shubik also breaks apart free-form games and rigid rule-based games, but additionally considers how the game teaches concepts to the player [Shubik 1983]. Klabbers builds a taxonomy of games by combining concepts from social systems theory with a semiotic theory of gaming, dividing agents in game into actors, rules and resources, each with their own syntax, semantics and pragmatics [Klabbers 2003]. | 14

Taxonomies have also been brought to bear to try to understand the people that play games, instead of games themselves. Bartle breaks players into four types: achievers, socializers, explorers and killers [Bartle 1996]. These four types sit across two axes: whether players prefer action or interaction (the first) with other players or the game world (the second). Similarly, Yee also talks about player motivations in online games [Yee 2006]. Yee breaks down player motivations into three primary categories (achievement, social and immersion) and ten subcategories. In a meta-analysis, Smith et. al. present a taxonomy of player modeling approaches, such as examining whether the player model is general or specific, or categorizes based on human reactions or game actions [Smith et al. 2011]. | 15

This list is by no means exhaustive, but gives an idea of how taxonomies are often used in games studies: to talk about games as a whole and players of particular kinds of games. In only a few cases, we can see categorizations of design approaches that can be similar to a taxonomy, such as Lewis' collection of motivational design strategies [Lewis 2014]. Individual game systems are less likely to be the focus of a taxonomy, and we focus on one particular game system: | 16

crafting.

Dimensions of Games

In the book *Half Real*, Juul evaluates seven definitions of "games" and synthesizes a single definition that he terms "the classic game model" [Juul 2005, 23]. This definition is intended to apply to all "traditionally constructed" games from different cultures and across the 5,000-year history of games. In order to accomplish this, Juul surveyed game definitions to extract concepts necessary to all games:

> [A] good definition should describe these three things: (1) the system set up by the rules of the game, (2) the relation between the game and the player of the game, and (3) the relation between the game and the rest of the world. [Juul 2005, 28]

After synthesizing seven different scholars' previous definitions of games [Juul 2005, 30], Juul derives six common features to compose his own definitional framework: fixed rules, variable outcome, valorization of outcome, player effort, player attachment to outcome, and negotiable consequences [Juul 2005, 36]. Juul uses these features as measuring sticks to determine how game-like an experience is. A *game* has all of these features, while experiences like gambling or open-ended simulations are borderline cases, and even looser experiences like storytelling or free-form play are simply not games. Some features are binary, such has alternative ending in an experience counting as not having a fixed outcome. However, whether that uncertainty is based on skill, luck, or how impactful it is on the experience of the outcome may be lost.

Those experiences that are too far off of the canonical game to count as a game still have a place on Juul's model. Readers can argue that free-form play should count as a borderline case because free-form play can spontaneously contain many of these game features. In making that argument, readers such as [Deterding et al. 2011] and [Myers 2009] are using Juul's features and critical language to dissect and analyze games regardless of where they ultimately land on the graph. Juul admits that his model "is no longer all there is to games" [Juul 2005, 53]. We understand that our model is also not all there is to crafting systems. However, Juul understands his model is an abstract tool, "an abstract platform upon which games are built, a platform that games use in different ways" [Juul 2005, 54]. We see our work, following Juul in outlining dimensions, as a new foundational and abstract tool by which to examine crafting in games.

Unfortunately, we lack solid definitions of craft systems as this paper refers to them. [Tychsen and Hitchens 2009] and [Consalvo 2009] refer to "crafting" as reference to the colloquial category of crafting professions/skills/jobs in MMOs, and [Leung et al. 2014] is a patent for a "Crafting system in a virtual environment" that attempts to define their use of "crafting systems" with the following:

> Provided herein is a crafting system that allows users or players of a website to create their own "unique" virtual items for use on the site. With the crafting systems, players can collect materials from within a virtual social environment to customize items, which the player can use to express their individuality. Crafting can include materials such as, but not limited to, textiles and related finishes, edible materials, paint, as well as clay, stones and other earthen and organic materials to create pottery. Crafted items include but are not limited to clothing, shoes, accessories, jewelry, food, beverages, dishes, ceramics, paintings or other forms of artwork and designs and furniture.
> [Leung et al. 2014]

This broad definition, while attempting to enumerate some materials and end products, describes the patent's crafting system, not crafting systems in general, nor crafting systems within games. Without clear definitions of crafting systems suitable for game studies, we surveyed a broad spectrum of games, including explicit and implicit crafting systems, to find our dimensions of a crafting system: Recipe Definition, Fidelity of Action, Completion Constraints, Variable Outcome, Recognition of Outcome, Player Expressiveness, and Progression (see Figure 1). These dimensions will be fully developed in the core of the paper, Definition and Dimensions of Crafting in Games, following our examination of real-world crafting as a basis for the cultural understanding of crafting systems in games.

The Evolution of Crafts

Operationalized and systemic representations of crafting, like those found in games, are necessarily in dialog with cultural conceptions of real-world crafting, which is a contested subject in and of itself. *To better understand what would cause a developer to declare a system a crafting system, or to build an implied crafting-like system, we examined how crafting formed and exists today in the real world*.

Paul Greenhalgh separates the history and characteristics of craft history into three threads: *decorative art*, the *vernacular*, and the *politics of work* [Dormer 1997, 25]. These attributes cover the split between art and craft, public opinion of handmade goods over time, and the connection of crafts to mercantilism and the economy, all of which have affected crafting representations in games.

Decorative vs. High Art

In the West, roughly since the Renaissance when Academies were being created, the five fine arts of painting, sculpture, architecture, music, and poetry became privileged over other artistic professions [Dormer 1997, 27]. Other arts, which Greenhalgh calls *decorative* arts, were considered by some to be too functional or made of too-cheap materials for the designation of fine art [Dormer 1997, 29]. Games often use the signifiers of high art, usually easels, paint brushes, or tubes of paint, to unequivocally represent creative or artistic endeavours, such as *The Sims* (see Figure 2). The modern resurgence of DIY (Do-It-Yourself) culture and the indie (independent) craft movement celebrate all forms of art high, low, and undiscovered [Levine and Heimerl 2008]. The wide variety of crafting systems this paper discusses aligns with the indie view and includes high art, decorative art, and creative skills that may have no real-world equivalent.

Figure 2. The icon for the "Artistic" skill in *Sims III* is a paintbrush, and is often raised when characters paint on in-game easels.

The Vernacular

After the privileging of high art, the vernacular surrounding the decorative arts of rural pre-industrial country craftsmen was unfavorable until the late 19th century, when it took on the perception of being "unpolluted" and "authentic" when compared to mass-produced goods [Dormer 1997, 31]. The common opinion of handmade items has oscillated between these favorable and unfavorable views for the last century. In *Handmade Nation*, Andrew Wanger speaks of the 60's and 70's "hippy counterculture," how 80's and 90's artists distanced themselves from hippies by focusing on galleries and museums, and today we have another DIY resurgence [Levine and Heimerl 2008, 2–3]. Debbie Stoller in *Stitch 'n Bitch* traces a more nuanced history of knitting in particular: how each generation either snubs the "silly domestic work" or takes up the needles through necessity or pride, such as in times of economic hardship or war [Stoller 2003, 13]. Some alternative terms we have heard refer to this DIY resurgence include maker practice or maker movement, hacker culture, and appropriation of handicrafts [Tanenbaum et al. 2013].

Craftivists, practitioners of a specific subculture of the maker movement, create crafts as a means of political self-expression, such as graffitiing with yarn, or "yarn bombing," in the effort to reclaim cold public spaces [Mould 2014]. Some games such as *Jet Set Radio*, a rollerblading action game about tagging graffiti and evading authorities, attempt to faithfully model transgressive behaviors. We would consider transgressive crafts within games as within the scope of this paper. The practice of taking "Grandma's crafts" — traditionally feminine crafts, such as scrapbooking, knitting, sewing, and other textile crafts, used to convey "deceivingly harmless" messages reinforcing gender, religious, political, or familial identity or ideology — and inverting, twisting, or otherwise *subverting* their usual meaning is known as subversive crafting [Winge and Stalp 2013]. *Modding*, a jargon term for modifying a game in any way outside of the

scope of the gameplay, could be considered subversive or transgressive craft if the craft in question is game creation or game authoring [Letzter 2015]. However, the scope of this paper is crafting systems modeled *within* games, and, in general, modding games themselves is a higher level of abstraction than the focus of this paper.

Many sources, including those cited previously as well as David Gauntlett's book *Making is Connecting,* speak to the current renaissance of craft knowledge and products spread throughout the internet [Gauntlett 2013]. Popular websites such as *Ravelry*, *Craftster*, and *Instructables* encourage users to share their own patterns; and commerce websites like *Etsy* or *Amazon: Handmade* exist for users to sell their handmade goods and materials. It is not surprising, then, to see crafting systems and other creative acts being integrated into more games, leveraging art, craft, DIY, and engineering visual design or theming with or without mechanics or systemic models of crafting[6]. A game's craft aesthetic can wrap around and re-influence real-life crafting, which has been reported by the developers of *Little Big Planet* [Westecott 2011]. Games that *do* make crafting playable are also often supported with a library of patterns, economies, and player strategies that mirror communities of real-life crafts (see Social Aspects below for more detail).

Craft and Creativity

The Industrial Revolution's toll on an individual's creative freedom was being felt widely enough to seed an international shift in values, the Arts and Crafts movement, across Europe, in other nations in the British empire, in North America, and later in Japan [Dormer 1997, 35–36]. This movement defined and popularized the concept of "craft" as this paper refers to it: as a creative endeavor alongside and sometimes including the fine arts [Dormer 1997, 38]. In the games we have analyzed, we consider everything from drawing to city-building and spell-crafting to fall under the umbrella of crafts.

What is creativity, then, as a necessary component of a creative endeavor? In general, many researchers of human and/or computer creativity agree that the "standard definition" is some combination of originality (or novelty, or innovation) and effectiveness (or appropriateness, or usefulness, or value) [Runco and Jaeger 2012]. Most games do not try to operationalize or evaluate creativity and instead leave it up to player interpretation. However, some games, such as those discussed in the System Recognition dimension, attempt to procedurally evaluate, model, or simulate some qualities of entities made by the player.

Related to evaluation of crafts is motivation: why do people craft, in games or otherwise? Unfortunately, to the authors' knowledge, there is not a quality source that outlines what motivates players to craft in games. We can, however, extrapolate from the reasons why people craft in general and begin to evaluate how they may or may not apply to crafting in games. Winge and Stalp did an ethnographic study of 44 crafters, primarily subversive ones, and some of their reasonings were: for necessity (or lack thereof), to develop skills (learn), to relax, to have fun, to create a supportive space, and to develop identity [Winge and Stalp 2013]. Many find it "therapeutic" to quietly reflect, or think about nothing at all, and to watch a product slowly unfold [Winge and Stalp 2013, 75]. To summarize, some craft to make a specific object or outcome, and others for the process of making regardless of the outcome. While the value of a specific object may be worth the cost of admission to create it, in the real world or in a game, the process is the more interesting motivation here. Through Csikszentmihalyi, we find a common thread to games and crafting: the flow state [Csikszentmihalyi 1996]. While not the only feeling involved in the process of crafting, the flow state — where there is a balance of challenge and skill that consumes a person's whole concentration — is shared with gaming in general, as well as crafting within the game, and is likely one major motivator for players to seek crafting systems in games.

Aesthetic Diversity

According to David Pye in *The Nature and Art of Workmanship*, workmanship can be broken into two contrasting groups, "the workmanship of risk" and "the workmanship of certainty" [Pye 1968]. The workmanship of risk is workmanship where the quality of the result is at risk during the process of making — a workman can make mistakes in the act of crafting. In the workmanship of certainty, the quality of the item is predetermined at design time, usually by automation [Pye 1968, 20].

To Pye, the quality of a craft is determined based on how closely it follows or augments a particular intended design.

Regulated workmanship is when the design and final crafting result very closely match. By definition, the workmanship of certainty is regulated. The workmanship of risk is permitted to be worse than the design (thwarting it) or better (augmenting it).

While the quality variation of the workmanship of risk may seem undesirable, Pye argues that it is desirable for producing subtle variation across created objects. Mass production (and the high degree of the workmanship of certainty required for it) "lack[s] all depth, subtlety, overtones, variegation, diversity…" [Pye 1968, 19].

We call this property "aesthetic diversity", the quality of having many variations on a theme in a collection of crafted objects. This sort of variation is often not enacted in crafting systems, in fact, crafting systems almost always fall under the workmanship of certainty. Reasons for this are discussed in the Undefined Recipes and the Variable Outcome dimensions. Automation affordances often work against aesthetic diversity, as is discussed in Automation.

Definition and Dimensions of Crafting in Games

We further develop our preliminary definition of crafting in games: *crafting* in a game is the set of mechanics and interactions by which players thoughtfully manipulate materials to create a new entity in the game world. The entity[7] might be created from scratch, or created by modifying or augmenting something that existed before crafting. The entity may be animate or inanimate, a representative of real-life crafting or some fully imaginary and virtual form of crafting only possible in the abstractions of playable media. Resources are often consumed or transformed in the creation process, but don't have to be. The mechanics and interactions which comprise crafting include actions and choices that players make during the creation process,[8] and the game system's immediate responses and mechanical interactions with those actions and choices. Players should enact these actions and choices with the intention of making something, and with the feeling that accompanies, even if they don't know precisely what the created thing will be. And finally, the creation process and the created entity are explicitly represented in the context of the game world[9].

In this definition, we aim to outline a general game experience for which the following design dimensions may be a useful analytical tool, rather than prescribe a narrow characterization of what crafting in games always is and always will be. Creating virtual objects in games happens in many genres and thematic domains, in both computational and non-computational games, and in dialog with many other game systems. The proposed seven-dimensional framework cuts across these differences, posing questions and design considerations, and mapping the experiential space in a way that a narrow definition cannot. We propose that the wide definition offered above be used as a rough guideline for what games to apply the dimensions to, in order to answer game studies and game design questions more useful than, "Is it a crafting system?"

What follows are descriptions of seven dimensions or axes that characterize and explore the space of possibilities surrounding crafting in games. While similar to Jesper Juul's dimensions of games, our dimensions will differ from his in that in his system there is a canonical concept of game as exemplified by games like chess. Games that score high on all of his dimensions converge on this canonical concept. In contrast, there is no canonical crafting system, just as there is no canonical real-life craft or way to craft. So there is no position along the dimensions that converges to a canonical crafting concept, no position along any given dimension that is implicitly preferred. Rather, each point on each dimension supports different kinds of experiences and play. Among the 64 systems surveyed and discussed here, some readers may also contest particular examples as not "crafting" systems, according to common game cultures' understandings of crafting. However, we illustrate that the framework is designed to surface useful discussion wherever players create things in virtual worlds. It is intended to serve as a lens, rather than a box, and is informed but not prescribed by the sometimes arbitrary labeling traditions of different gaming communities. We have sampled a wide enough range of different systems to both stretch the boundaries of our dimensions, as well as illustrate combinations of characteristics that have yet to be implemented as gaming experiences in order to help discover them.

It is important to note that the seven dimensions characterize individual crafting systems, not games within which crafting systems are embedded. Individual games can have multiple crafting systems. Games like *Minecraft* are so committed to crafting that they have multiple, interwoven playable models of crafting, each of which will have their own

characterization along the crafting design dimensions.

We now describe each dimension in depth. While there is no intrinsic dimension order, they are presented here roughly in order of the crafting process as enacted by the player. We begin with qualities that primarily describe the enactments of crafting: Recipe Definition, Fidelity of Action, and Completion Constraints. That is followed by what happens after that process is enacted: Variable Outcome and System Recognition of Outcome. Our list concludes with qualities of the larger game experience within which the player may have enacted crafting many times: Player Expressiveness and Progression.

37

Recipe Definition

A *recipe* is a representation of the knowledge necessary to transform a collection of game objects (ingredients or raw materials) into a new object. For example (described more below), a recipe for a pickaxe might specify two twigs and two flint as the materials necessary to create a pickaxe. Just as objects such as twigs, flint and pickaxes can be virtually possessed by a player, so too can recipes be possessed, where recipe possession means that the player now possesses the knowledge to perform the transformation from ingredients to produced object, typically represented within the game by providing this transformation as a new player action.

38

The Recipe Definition dimension refers to the flexibility of the ingredients used as input to the crafting process. At one extreme is the *strongly defined recipe* which dictates recipe ingredients precisely, leaving no room for customization or choice in ingredient specifics. This provides less control for the player, but greater control to the game creator, who can explicitly enumerate all possible craftable objects. At the other extreme are crafting systems with *undefined recipes*. Such systems use an underlying simulation to determine what happens when objects are combined to create a new object. By not enforcing specific ingredient combinations, this opens more possibilities for experimental and creative play, but reduces the game creators' control over crafting. Between these two extremes are recipes with some predefined structure that support variability in input ingredients and the resulting created object.

39

Strongly Defined Recipes

Crafting systems which only support crafting by strict recipe sit at the high end of this dimension. These recipe-based crafting systems effectively force players to follow predetermined links between input and output to craft anything. Strongly defined recipes are so pervasive in games that such systems are more commonly called "crafting" by game designers [Naasz 2013] [Steinke 2015], player communities [Item Crafting 2017], games journalists [Frank 2016], and in-game terminology (*Minecraft*) than the freeform crafting of undefined recipes.

40

Don't Starve has several crafting systems, including item crafting, crockpot cooking, and structure placement, which fall at different positions on this spectrum. But the one the game itself calls "crafting" (via in-game text), is a classic example of a strongly defined recipe-based system (Figure 3). For example, to make a pickaxe, the player needs two twigs and two flint in their inventory, opens the tools crafting menu, selects the pickaxe icon, and clicks "build." The player similarly clicks on other object icons and their "build" buttons to make torches, traps, top hats, crock pots, etc.

41

Figure 3. Crafting items and tools like pickaxes in *Don't Starve* involves selecting from a library of completely defined recipes.

The recipe for an item never changes, and never allows a range of possible resources to fulfill a single ingredient requirement. These recipes also never vary in their output. If the player selects and "builds" the "pickaxe," she will inescapably have two twigs and two flint removed from their inventory and replaced with a pickaxe. There's no notion of contextualized raw ingredients, of different kinds or shapes of wood or stone, perhaps from different parts of the virtual world, which might otherwise be visible in the texture and quality of a pickaxe they were used to create. A crafted pickaxe also isn't changed by random dice rolls, or by the experience, handiness, or personality of the player or player-character. A pickaxe, or any other item the player makes in *Don't Starve*, is the same, whether crafted by a full, healthy, lucid adventurer, or one delirious from hunger and monster attacks.

42

From a designer's perspective, limiting the otherwise large possibility spaces (as strongly defined recipes do) is practical, both reducing development complexity and tightening authorial control over a player's experience. Strongly defined recipes have an analog to node-and-link-based hypertext, especially when the result is at first unknown to the player. Performing a crafting action using the recipe is like traversing a hypertext link, where the source text is the set of input ingredients, and the destination text is the output of crafting. That the two "texts" are connected by a "crafting" relationship signals a "part-of" or "transformation" relationship to the player. While many existing games use these links functionally, they could just as easily be made more lyrical and interpretive, in the sense of Susana Tosca's *lyrical links* [Tosca 2000].

43

When the individual recipes of a virtual world are considered together, they can describe the logic and conceptual makeup of the fictional world and its inhabitants. Many of the recipes in fantasy RPGs like *World of Warcraft*, *Skyrim*, and *Dragon Age* are composed of found materials like herbs, minerals, and animal fur, which are transformed into items you'd expect from Medieval European craftspeople. Ingredients in *No Man's Sky* recipes are often atomic elements or alien machinery, which are transformed into science fiction space-travel technology. The ingredients of strongly defined recipes tell the player how their character understands and acts in the fictional world.

44

Furthermore, the most important of the strongly defined recipes of a game are often revisited many times during play, and thus solidified in players' minds and Wiki pages. This recipe-learning through repetition can solidify the meaningful connections of strongly defined recipes, and cause players to see the game world through the subjective lens of their fictional avatars.

45

Parametric Recipes

Crafting according to a semi-defined recipe links multiple ingredient sets to the same output, or links a given set of input components to multiple possible results. We call both patterns *parametric recipes*, or recipes that take parameters of input ingredients. In both cases the player has some options in enacting a recipe, either by being able to choose among multiple sets of ingredients to produce a given object in the first case, or in being able to produce multiple objects given a set of ingredients in the second.

An example of parametric recipes is the *Don't Starve* crockpot cooking system. The crockpot is a crafted structure placed in the world, which itself is a crafting station for producing food items. The player navigates to it, clicks on it, fills the four available slots with individual ingredients, clicks "cook," and waits for the result to appear in the pot. Crockpot cooking is recipe-based because each combination of four ingredients placed in the slots is linked to an output according to hand-authored connections (where order of the four ingredients doesn't matter), but the recipes are templated. Each one allows multiple different ingredient combinations for the same output. For example, the Stuffed Eggplant recipe calls for one eggplant, one vegetable food item, and two fillers (Figure 4). The eggplant requirement can be fulfilled by two different single items (eggplant and cooked eggplant), and the vegetable and filler can be fulfilled by 25 and 52 different items, respectively. In effect, Stuffed Eggplant can be produced in 2x25x52x52 = 135,200 different ways.

Figure 4. The parameterized recipe for Stuffed Eggplant in *Don't Starve*'s crockpot system, as shared on the player-run *Don't Starve* Wikia page (dontstarve.wikia.com/wiki/Stuffed_Eggplant). The second box indicates "vegetable," and the question marks indicate "filler."

Though *the dish is the same*, whether made with twigs as filler or with meat, the space of possible input gives the player potentially interesting choices to make in enacting the recipe. She can execute the recipe strategically, using the most widely available or lowest valued ingredients in the set of options. A strategic enactment of Stuffed Eggplant would be to use twigs, a low-risk, readily available, widespread resource in the world, before meat, which is usually risky, rare, and more productively used in other recipes. And the player can adapt her enactment according to what she has an abundance of at the time. But she can also enact it in support of more self-expressive play, which can be encouraged or discouraged by other aspects of the game's design and the context of crafting. For example, the player can use less efficient ingredients even when there's an option to use others, if the less efficient ingredient suits their specific play style or desired experience, particularly in terms of role-playing, perceived realism, or exploration and experimentation (see Player Expressiveness). Furthermore, defining a recipe as a space of input, rather than a single possibility, changes the way the player thinks about their crafting. The player can see resources in the world as classes of ingredients connected by abstract properties, and, similarly, interpret crafted entities as the union of abstract properties that define their input parameters.

The second kind of parametric recipe is one which represents, rather than ignores, the individuality of ingredients in an enacted recipe's result. There is still a space of possible input, like the multiple options of many *Don't Starve* crockpot recipes, but now the choice among those options is represented in the output. Consider, for example, the *Minecraft* pickaxe (Figure 5). The player arranges materials in the same, T-shaped graphical recipe to craft any pickaxe in the game. The handle is always formed by two vertical sticks, but the head can use several different kinds of blocks, with the different materials contributing different strength and durability values for the resulting pickaxe. The pickaxe recipe, in effect, is parameterized by those materials. The pickaxe isn't completely defined; the player has room to enact the recipe in different ways. But, unlike for Stuffed Eggplant, different ingredients produce observable changes in the resulting item.
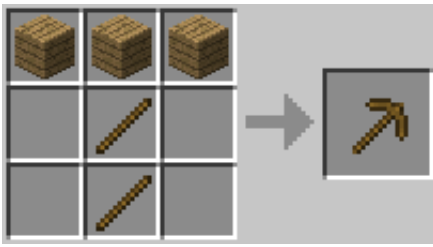
Figure 5. A *Minecraft* pickaxe is always crafted by the same, T-shaped graphical recipe, but many different materials can be used to form the head, making it parametric. The material parameters are carried through to the output in the form of its strength and durability. Diamond pickaxes last for 1562 uses and can mine all blocks, while wood pickaxes have durability 60 and can only mine basic blocks.

Considering games as a "series of interesting decisions" [Meier 2012], this kind of parametric recipe affords the player crafting decisions which are arguably more interesting and impactful than the previous form of parametric recipe. The decisions she makes in enacting a recipe are visible and significant to gameplay as long as the crafted item exists or is in some way simulated in the virtual world. A player's choices are made tangible in the very tools she uses, items she carries, and structures she adds to the world.

As the constraints of a recipe are relaxed, the closer the experience of following it resembles the freeform mode of creating described in the next subsection. And the closer the crafting experience is to this freeform, experimental mode, the greater the crafter's agency and expressive potential, the less the authorial control of the recipe-writer, and the larger the impact of the player's choices on the output and player's experience[10].

Systems with Undefined Recipes

Crafting systems without well-defined recipes sit at the low end of this dimension. Instead of explicitly mapping specific crafting inputs to specific outputs, these systems tend to specify rules by which a player can craft, with the results determined by the interaction of those rules. For example, *Minecraft* block placement (different from the grid-based crafting used to craft a pickaxe) allows players to attach solid cubical blocks to each other along a three-dimensional grid, as long as there is another solid surface in the world to attach them to.

Combining components without a recipe is more akin to real-world art-making than real-world cookbook-following. A given set of components may be combined in many different ways to produce many potential results. There is not a hard-coded link from input to output but, rather, a large space of possibility reachable from a given set of input components. Players can create things that are individual and perhaps even unique, if the possibility space is large enough. They can express themselves through crafting; they can take actions and make choices that not all — and sometimes not any — other players will make. Players can create things that surprise even the developers. For instance, because water can be poured into holes and is subject to abstract liquid mechanics in *Minecraft*, players have invented endless title pools that their avatars can ride in perpetually, without movement input from the player (Figure 6). These "AFK pools" have become popular ways to prevent servers from automatically logging players off for extended idleness (AFK standing for "Away from Keyboard"). In *Terraria*, obsidian farms and slime machines are often crafted by players to automate particular resource gathering. Freeform crafting systems often sustain large gaming communities of inventive in-game crafting. This kind of inventive play is one of the primary draws of games like *Minecraft*, *Terraria*, *Besiege*, *SimCity*, *Kerbal Space Program*, and *Draw Something*.

Especially in the AFK pool and obsidian farm examples, players are inventively crafting with the mechanics of the larger game systems in mind. One of the hallmark affordances of crafting in games is that crafted output exists in a virtual world, interacting with the other systems in the game to varying amounts (see Recognition of Outcome). It becomes part of other game systems, influencing world variables and the behaviors of other objects, while they, in turn, influence and act on the crafted item. When Recognition of Outcome is high, players need an understanding of the operations of the game system to be able to intentionally control it through crafting. This understanding, described by the *SimCity* Effect [Wardrip-Fruin 2009], is iteratively developed from player expectations, experimentation, and observations during play.

Freeform crafting allows players to test and refine their expectations of system behavior in a unique way, by making them material. Indeed, crafting in systems with undefined recipes *and* high Recognition of Outcome supports the player learning and iterative mental model-making of the *SimCity* Effect.

Figure 6. A *Minecraft* player-character riding in an AFK pool.

Low recipe definition also allows for crafted objects with individuality — that express aspects of the process and the person who created them — a quality commonly valued in real-world crafts [Pye 1968]. *Draw Something*'s drawing system is so freeform that the player is visible in what she crafts, through her virtual brush strokes, art style, and chosen subject. There is also great variation in how players draw the same item for a friend to guess. And there are often unintended flaws, which means that *Draw Something* supports the workmanship of risk, as well as aesthetic diversity, in its virtual crafting [Pye 1968]. This individuality is nearly impossible for a player to express in a strongly defined recipe.

55

Players can improvisationally create within an undefined-recipe system, but also share their inventions and borrow from others through social venues external to the game. They essentially define their own recipes that other players can follow, enabling the kind of socially embedded creativity that is characteristic of real-world crafting communities (see Social Aspects).

56

Fidelity of Action

One concept of psychology that game designers take advantage of in their trade is the idea of *affordance* as the possibility of an action on some other entity. Buttons beg to be pushed, and tea cups suggest being held by their handle. Affordances[11] are a core component in leading a player to perform particular actions within the game world: pickaxes are used for mining, and sewing machines sew fabric. Fidelity of Action represents *how detailed those player-performed actions are within the game world, how accurate those actions are to applicable real-life actions, and how embodied the player performance of those actions is*. We reiterate that there is no moral judgement or criticism of a game's design on

57

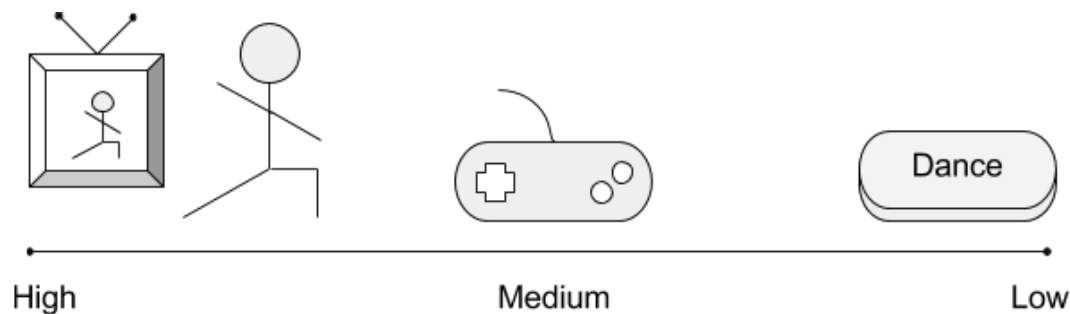how well or inaccurately a game represents their chosen affordance.

Figure 7. An illustrative example of Fidelity of Action using dancing as a subject. The closer to dancing the game demands of the player, the more physically demanding and nuanced the controls, the higher the system would score.

As an example, consider the representation of knitting in games. In real life, a knitter performing the craft uses manual dexterity to manipulate a continuous thread on two sticks called needles, which produces a knit fabric. The act of knitting has been represented in games in a wide range of fidelity that stretch across this dimension. *Knitting Simulator 2014* is an example of extremely high fidelity to knitting because it involves embodied knitting, the player physically mimicking the act of a knitter, on custom game controllers consisting of knitting needles with motion sensors. Each wiggle of the knitting needles represents stitch progress in the game. A game with less fidelity in the knitting genre would be *Crafting Mama* on the 3DS. Without any controllers representing the knitting needles, the player's embodied action while playing the game is tapping the stylus on the Nintendo DS screen, which is further away from those in *Knitting Simulator 2014* and from real-life knitting. However, the player does tap the screen rapidly to progress stitch-by-stitch, and in traditional *Mama* fashion, she judges your performance at the end, and you can even dress up Mama in your scarf. The knitting mini-game in *Crafting Mama* is still very much about the act of knitting. Games with the lowest amount of fidelity in their crafts abstract away the process and leave nothing but a progress bar and/or an end result. For example, a tailor in *World of Warcraft* (*WoW*) makes fabric, using nothing but fiber, at a click of a button. It could be a woven, knit, or even felted fabric; the player's character could be using knitting needles, a loom, a crochet hook, or some other tool. We just don't know. The fidelity of fabric-creation in the case of *WoW* is extremely low, so low that we cannot distinguish the process as knitting.

The Applicability of Fidelity

Custom arcade machines, the Nintendo Wii system with its Wii-Mote and Nunchaku, the Playstation Move and its Motion Controller, Microsoft's Kinect, and other games and systems with custom hardware such as *Rock Band* have all helped explore the space of high-fidelity experiences. These experiences have so far focused primarily on musical instruments, combat, or other physical tasks, such as requiring accurate aiming with a "gun," slicing with a "sword," or placing feet in dancing (Figure 7). Some crafts, such as knitting described previously, are also physical. However, some crafts, and especially design activities and planning, are not primarily performed via physical actions.

Accuracy of Abstractions

In *SimCity* (2013), where players craft a city, is the player a mayor, an urban planner, or an urban engineer? Real-life incarnations of these jobs cannot plop down miles of road at a click of a button, or evict a neighborhood of high-class citizens, rezone the district, and instantly create a casino in its place. However, some aspects of the city such as traffic, pollution, hospital access, and water and power systems are more deeply simulated, and the management actions with respect to these city aspects are higher fidelity. Thus the Fidelity of Action of *SimCity* is quite low in some cases, much higher in others, and so we consider it to have a middling score for this dimension as an average.

Completely Fictional Crafts

Some developers design a completely fictitious crafting system and/or crafted items for their game's mechanics, such

as spells in *CodeSpells* or items in the *Atelier* series. In these cases, we judge the fidelity of the crafting system's process by the game's descriptions of that process, as well as player's fidelity in enacting those processes. We cannot compare the accuracy of the outcome because there is not a real-life equivalent. In *CodeSpells*,[12] the player writes code to create spells for their avatar to execute, including effects over time, changes to the environment, and visual elements, such as particle effects. As the game is made of code and exists by executing code, we determined there was no greater fidelity for a digital fictional system other than programming its behaviors.

In contrast, the *Atelier* series describes the process of alchemy as partially physical. Alchemy, in this case, is a fantastical incarnation of the real-life profession, which involves mixing loosely related materials in a special cauldron to produce everything from potions, baked goods, equipment, and even living creatures. In *Atelier Totori*, the protagonist is often shown learning how to stir contents of an alchemy cauldron in a particular manner. However, while each *Atelier* game involves different alchemy minigames as the primary gameplay and progression mechanic, none involve stirring or embodied action with the cauldron. The player progresses through a series of menus, managing ingredients, item properties, and alchemical skills, but the player never learns the details of how the alchemist makes the final product. While the process of alchemy is intricate in many areas and offers extraordinary depth in its menus, it fails to fully model all the aspects of the craft it created, and so the series is slightly above average on the Fidelity dimension.

Completion Constraints

Everything that is required for an entity to be crafted is encapsulated within this dimension. For the general crafting process, resources are usually consumed, perhaps some tools are used (what we call required components), and time passes to indicate the transformation of resources to the end product. Sometimes a game might impose other restrictions on the player, such as only being able to have two professions in *World of Warcraft*, what we discuss below under miscellaneous constraints. While this paper does not address systems outside of crafting, such as prerequisite resource-gathering, the use and consumption of resources or tools heavily influences a player's crafting experience. Designers often use resources or *recipes* to gate *progression*, require crafting stations to pace gameplay, or consume player time as a function of gameplay.

The completion constraint categories are also problematic upon further inspection: there are many systems that use no resources, some games with legal tender that make differentiation between resources meaningless, and the concept of time is not consistent between systems. Given the breadth of systems and mechanics, the subjective difficulty of these constraints, and the often compounding influence of these constraints due to crafting trees, it became impossible to objectively compare the impact of the different types of constraints between systems.

Instead, we largely took a quantitative approach of how many different types of constraints were involved in the crafting process (resources, time, location, and peripherals), with the presence of miscellaneous constraints and the severity of the constraints nudging games that fell within the same number.of types of constraints (Figure 8). For example, systems with resources are more constrained than those that do not use resources, but resource systems that have a legal tender that allows for resource trade are less constrained than those resource systems without. In general, the higher the score, the more disparate conditions the player must satisfy in order to complete crafting.
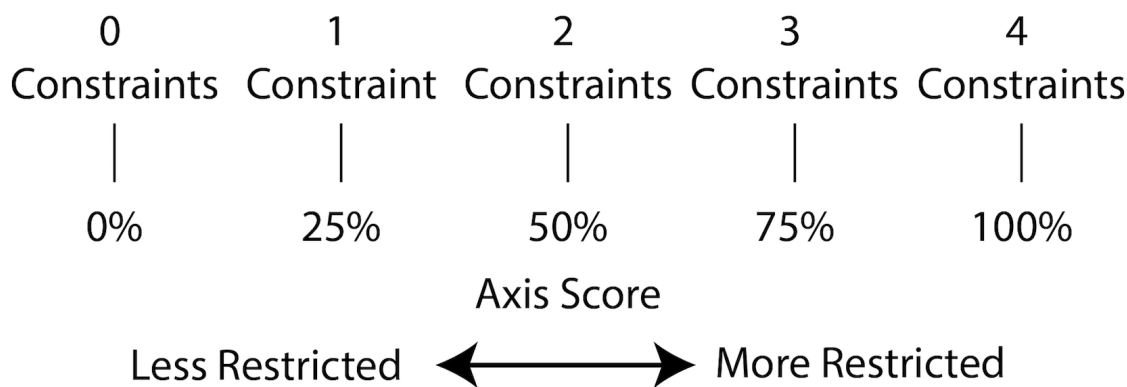
0	1	2	3	4
Constraints	Constraint	Constraints	Constraints	Constraints
│	│	│	│	│
0%	25%	50%	75%	100%

Axis Score

Less Restricted ←————————→ More Restricted

Figure 8. Our quantitative approach to this completion constraint dimension. Each main constraint category (consumed resources, present resources, time, and location) that was expressed by the system increased its overall score by 25%. The score was then modified by 5-10% per constraint if it was particularly lax or egregious (such as being a movable or symbolic location or having a currency relaxing those constraints), or if it had another notable constraint not captured by the categories (a miscellaneous constraint). If the system in our graph encapsulates multiple crafting systems with different scores, we average their score and make note of it.

Consumed Resources

Games often deal with consumed resources — currency or materials players acquire that can be spent in order to achieve certain goals in a game. Consumed resources may also include the recipe itself as a blueprint object and is actually consumed to craft an item (such as in *EVE Online*, *Lichdom* spellcrafting, or *Wildstar* gear crafting). Any prerequisite material with a limited durability, in that some part or whole of the material is removed or destroyed during crafting, counts as a consumed resource. Thinking about games in terms of how these resources flow, how they are acquired and spent, has a history in game analysis. For example, Dorman's *Machinations* is a way to describe arbitrary game mechanics in terms of resource flows [Adams and Dormans 2012]. 66

However, not all crafting systems have consumed resources. *Knitting Simulator 2014*, where the player goes through the actions of knitting, does not consume any resource. Players do not spend anything to make a new stitch. In general, when players play with crafting systems in free-play modes (where they are presented with an unlimited amount of resources that are always present), it becomes challenging to consider crafting systems as only resource transformation systems. 67

Present Resources

Materials or tools that are not consumed, but are still required to be present in an inventory or game world, are considered present resources. We bother with the distinction because present resources are a one-time need that is more flexible than consumed resources. A jeweler's kit is required to be held by jewelcrafters in *World of Warcraft* for them to create gems and jewelry, but so long as they have that item, they can craft at any location in the world (Figure 12). While economies are often deeply integrated with consumed resources, present resources, other than possibly first acquisition, are separate from the influence of currency. 68

Once the player acquires access to a present resource, it often opens up a whole new set of *recipes* in crafting *progression*. For example, the hellforge is a crafting station in *Terraria* that can only be found in the underworld (but it can be picked up and carried back to the player's base). It can smelt some of the most powerful pre-hardmode materials and can be used as a consumed resource to create the first forge able to smelt hardmode metals, helping unlock two new tiers of player strength. A crafting tree that combines consumed and present resources for a *Terraria* item is shown in Figure 9. 69

The constraint of present resources is kept separate from location constraints because, as in *Minecraft*'s or *Terraria*'s crafting stations, they can sometimes be picked up and carried with the player, functionally removing the location restriction that other systems may enforce. 70
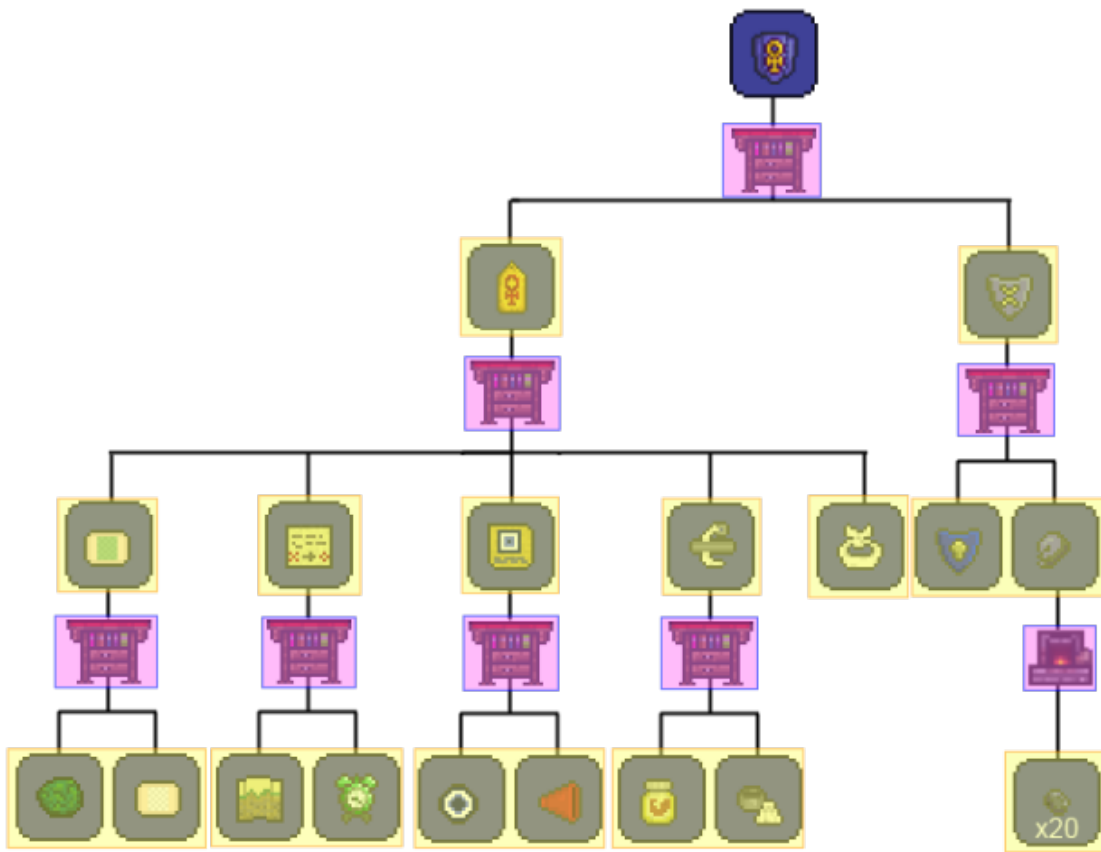
Figure 9. Marked up *Terraria* crafting tree for an Ankh Shield. The parts of the tree highlighted in yellow are consumed resources. The bookshelf-like parts of the tree highlighted in pink are present resources that must be placed in the gameworld, where the crafting actions must then be performed. In *Terraria*, some present resources are found in the gameworld, others are the result of crafting themselves. Original image distributed with the CC-BY-SA license, original attribution goes to the Wikia user Juper0.

Time

Time is a complex topic in games. According to Zagal & Mateas, there are four common temporal frames: real-world time (experienced in the physical world), gameworld time (in-game time including events around gameplay action), coordination time (coordination between player and agents, such as navigating menus), and fictive time (acknowledged sociocultural labels and event sequences) [Zagal and Mateas 2010]. Sometimes, crafting takes time in any of these temporal frames. This slows down a player's completion rate, either because crafting comprises a game in and of itself which takes time to play, or players just need to wait a certain amount of time before the crafting is finished (in real-world time or gameworld time). However, not all time is created equal. Some crafting delays take up all of the player's focus and ability to perform other tasks, while other delays are completely idle.

71

In *Final Fantasy XIV: A Realm Reborn*, a player must spend time performing a sequence of crafting actions to complete a craft, during which time the player cannot perform other gameplay actions without abandoning the crafting sequence (Figure 15). When romancing piñata animals in *Viva Piñata*, the player has a skippable cutscene, but a non-skippable minigame to play. Macro engines (discussed later in Automation) allow a player to remain attentive of other things as the performing of crafting actions is automated, but she still needs to wait for crafting to complete. In the same vein, players cannot act while waiting for a bar to fill to complete a craft in *World of Warcraft*, although this sort of delay does not require intense player focus (Figure 10).

72

Figure 10. In the *Draenor World of Warcraft* expansion, Scribes used ink in large quantities, which would take many minutes to make (15+ in this example). In the newest *Legion* expansion, Scribes use pigments directly and skip this step.

Time, when considered this way, is a hard, unskippable constraint. *Besiege* does not have this constraint, because the player can complete crafting her fantastical steampunk death machine by just clicking go. Further, *Besiege* starts players with a core 'starting' block, and considers this starting block a valid, completed craft; thus, players do not have to make any crafting decisions in any temporal frame in order to immediately begin playing the game In contrast, in *Final Fantasy XIV: A Realm Reborn*, the player must execute a series of repetitive actions in order to fill a completion bar to create a craft (Figure 15).

73

This is different from the time constraints in *Pokémon*[13] breeding. Time, as measured in the number of steps the avatar has taken in the game world, is a large delay in breeding a new Pokémon. Although the player is free to perform other in-game tasks while she waits, time only passes — at least in relation to waiting for pokémon to breed — when the player avatar moves.

74

Both these examples are different from how time is considered in *Eve Online*'s manufacturing system. Time in *Eve Online's* manufacturing system heavily involves the real-world temporal frame and is directly tied to the passage of real-world time outside the game — a manufacturing job is measured in a number of minutes, hours or even days (Figure 11). A player does not have to be playing the game for that entire amount of time. Progress on completing the manufacturing job happens regardless of the player being logged in.
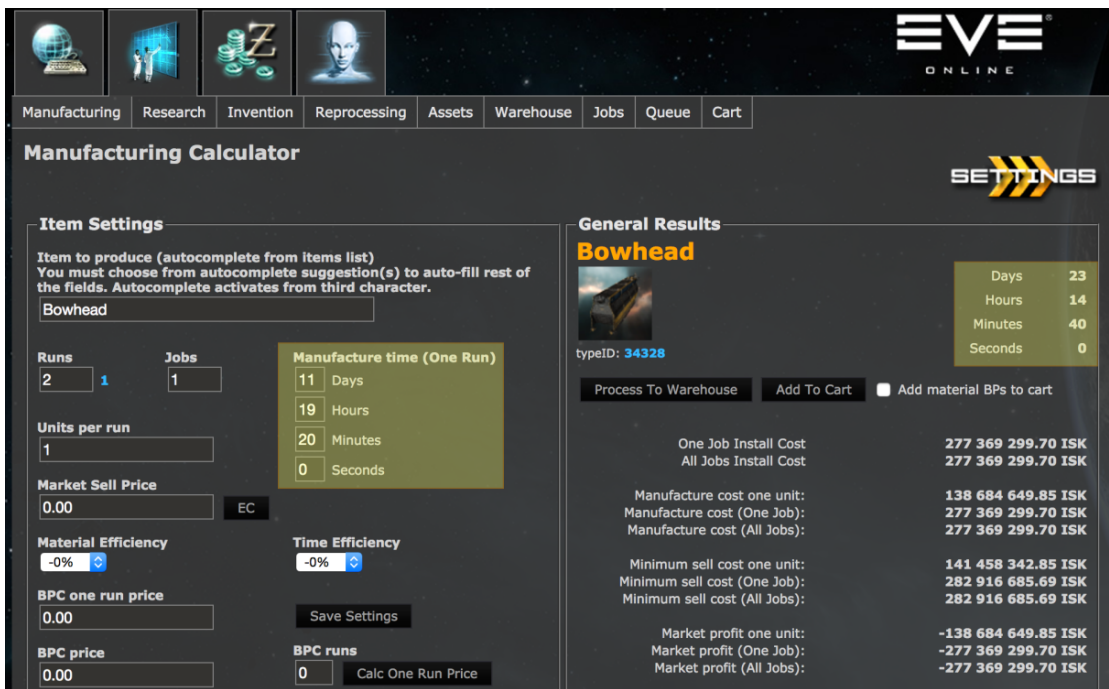
75

Figure 11. This is a screenshot of an *Eve Online* crafting calculator (accessed at http://www.eve-cost.eu/calculator). Time taken to craft an item is highlighted in yellow, showing the massive (for virtual crafting) timescales at play with crafting in Eve. The highlighted portion on the right is the time for all the runs to finish; the highlighted portion on the left is for a single ship to be created. At such large timescales, it makes sense to use the passage of real time, rather than just time when the player is actively playing or a proxy for time like steps taken.

Location

Typically, only a concern when a player is embodied, some games require a player to be in a particular gameworld location to craft items. Not all locations serve as meaningful completion constraints, however. *Kerbal Space Program* has specialized hangers where a player builds rocket ships and space planes, but a player can snap to these locations at any time (even in the middle of a rocket launch). Thus, they provide a thematic frame for construction (construction happens in hangars), but don't introduce an actual gameplay constraint within the crafting system. Systems that include more thematic and less functional location constraints receive a smaller score than those with immobile, hard location constraints. Examples of gameplay location constraints include workbenches in *Minecraft*, cooking fires in *World of Warcraft* (Figure 12), or the non-player character (NPC) craftspeople a player visits to craft weapons, armor, or gems in *Diablo III*.

Figure 12. Certain items, such as this Juicy Bear Burger in *World of Warcraft*, need to be created at a cooking fire, which is a present resource constraint. Cooking fires exist in certain spots in the world, forming a location constraint on where Juicy Bear Burgers can be created. However, players can also generate cooking fires, loosening the location constraint as compared to a forge, which are not mobile. An example of a present resource that has no physical location is the Jeweler's Kit, which must be carried in the player's inventory.

It is important to note that locations can also represent available land for construction, in games where the player is crafting some physical space (e.g., creating a pond or fencing in a garden space in a *Viva Piñata* garden). Although the player isn't really embodied in *Viva Piñata*, she still needs to maneuver the camera to an unoccupied place in the game world before being able to complete a crafting step of planting tree or building a fence.

Miscellaneous Constraints

In addition to consumed resources, present resources, time, and location, there are other miscellaneous constraints that are occasionally used in crafting systems. For example, in *World of Warcraft*, player characters are limited to two professions; each profession limits what the character can and cannot craft. Some professions have further specializations, such as *World of Warcraft* engineers being able to choose between gnomish or goblin engineering. To get around this constraint, players often have more than one character, each with a different set of professions. These alternate characters (alts for short) allow players to maximize their crafting output [Consalvo 2009, 413]. But time, location and resources account for the vast majority of constraints found in crafting systems.

Variable Outcome

Crafting systems are not always deterministic paths representable by a crafting tree. Two players can have the same tools, have the same resources, be at the same location, take identical actions, and *still* end up with two different results. We refer to this randomness as Variable Outcome.

Variable Outcome is not the same as having visible variables in a crafting system. Rather, fixed or random variables may be hidden from the player; this requires a player to explore the system in order to unveil the rules under which the gameworld operates. Players may develop a "feel" for the values a stochastic variable may take. Crafting systems often allow players to manipulate the probability distributions of their random variables, increasing the odds that they will get the outcome they want. For example, in *World of Warcraft*, whether or not crafting a particular recipe will help level up a crafting stat is random. However, certain recipes at a given cooking skill have a higher probability to give further stat increases and are color coded to show this to the player (Figure 13). By only crafting particularly colored recipes, the player is putting a thumb on the "increase crafting stat" variable and pushing it in her favor.

Figure 13. A screenshot of the player's cookbook in *World of Warcraft*. Each recipe a player knows is highlighted in a different color, which relates to the odds that preparing that recipe will increase the player's crafting skill (in this case, cooking skill). Recipes highlighted yellow give a medium chance of increasing the cooking skill. The screenshot shows how each instance of crafting does not necessarily increase the skill: orange recipes guarantee skill increase, green recipes indicate a low chance of skill increase, while gray recipes guarantee zero skill increase.

Randomness in a crafting outcome can sometimes include a complete failure to craft, or randomly assign properties to a completed item. An item can be of better or worse quality. Or the process may refund a variable amount of resources back to the crafter upon completion.

81

A common paradigm for games is to randomly apply some extra properties to a crafted item, with some of them being negative. Each act of crafting always produces something, but it might not be exactly what the player is looking for.

82

For example, when crafting a new weapon in *Terraria*, players combine various resources at a particular location in the game world. Part of this is deterministic; players will always get a particular weapon with certain default characteristics. However, the weapon also gets a random modifier that is outside the player's control. This modifier can impact the effectiveness and utility of the weapon at hand, shown in Figure 14.

83

Figure 14. The strongly defined recipe Wooden Sword from *Terraria* produces a sword with default properties. These properties are potentially changed by random modifiers. In these two examples, we see the "Legendary" modifier increasing the damage and effectiveness of the sword, while "Shameful" decreases its damage and its ability to keep enemies away from the player but still increases the sword's size

There is no way to control the random selection of a modifier; a player cannot make particular modifiers more or less likely. *Terraria* does allow for a player to reroll a modifier after a weapon has been created as many times as she likes until she happens on the desired one. For variable outcome, this is a common design choice — if a player can't have control over an aspect of crafting, at least she can retry somewhat easily in order to find the values she wants.

Variable Outcome is Uncommon

Variable outcome is not particularly common, especially in systems built for simulation. Most of the crafting systems analyzed do not have any variable outcome. The spaceship manufacturing in *Kerbal Space Program* is typical of the common case of completely deterministic crafting. Players have complete control over how the parts are placed on their spaceship, and how those parts will function. The player has absolute control over the form and function of each new spacecraft she creates. This helps reveal the underlying rules of the world to players via the crafting system. It is more difficult for players to form a mental model of the crafting system when it contains hidden variables or is intrinsically stochastic.

Variable Outcome in the Real World

Linking variable outcome back to real life crafting, it intuitively feels like a proxy for the random forces that impact successfully completing a craft in real life. Think of the random forces that impact successfully making a dish in a kitchen — a cook can accidentally add too much of an ingredient at a particular time, grab the wrong spice, or have the lid of the spice container come off while shaking some into the dish. However, Variable Outcome in game crafting systems leads to different effects than the variability in real life crafting.

In the *Workmanship of Risk* (see Aesthetic Diversity), Pye argues at length about the virtues of workmanship and how it relates to design. Items created with unregulated or free workmanship often look very different from each other, but can serve the same functional purposes. In addition, they can be of the same quality, even if they look very different. This kind of variability relates to the improvisational skill of the craftsperson given the dynamic richness of real-world situations. In contrast, variable outcome in game crafting systems often works against Pye's dream of aesthetic variability. Given the stochastic modeling of object attributes typical in crafting systems with variable outcome, the results are often not functionally equivalent — they vary along attributes with gameplay significance, making one crafted object clearly superior to the other.

Either due to internal motivation, *social pressures*, or other forces such as gameplay thresholds, players often look for

84

85

86

87

88

optimal items. These players grind until they get the optimal bit of gear, repeating the crafting attempt as many times as it takes until the outcome lines up with their definition of optimal. Variable outcome, in this case, doesn't actually improve the aesthetic diversity of various crafted items in games.

This actually makes variable outcome, in the games we analyzed, a poor proxy for the random forces that impact real-world crafting. A missed stitch, awkward tension, or a flaw in the yarn does not meaningfully change how warm a sweater is. However, this type of improvisational variability that free workmanship has been under-explored in crafting systems and is a rich future direction for crafting system design.

Progression with Randomness

Gaining influence over a random variable is a way to add progression into a crafting system, which sometimes ties this dimension to *Progression*. As players progress in the game (sometimes within the crafting system, and sometimes as part of a different system), players may get new ways to view the random variables at play or the ability to narrow the ranges on the random variables.

This is one of the primary ways players progress in the crafting system of *Final Fantasy XIV: A Realm Reborn* (*FFXIV*). *FFXIV*'s crafting system is complex, giving the player a range of crafting actions that are abstractions over ideas and concepts that ground out in real-world crafting (e.g. "Precise Touch", "Muscle Memory"). Some of these skills fill up a progress bar, and when the bar is full, the item is created (see Figure 15).

Figure 15. The progress window for an item during the crafting sequence from *Final Fantasy XIV: A Realm Reborn*, and a snippet of the crafting details relayed to the player in their chat log. The player uses crafting actions, such as the shown 'Basic Touch' and 'Careful Synthesis' to alter the progress, quality, durability, condition, and chance for high quality (HQ) outcome. Note that in the chat log these crafting actions succeed, but some actions have a chance of failure. There are three possible outcomes: failure (via 0 durability without a full progress bar, or quitting), success, and HQ success.

However, in Figure 15 we can see items also have a running durability score, which represents how many more actions can be performed before the crafting is forcibly resolved. Finally, there is a quality bar, which is the chance that, when the craft is completed, the item produced will be of high quality (better in terms of stats). In addition, randomly, the next action taken can be flagged as excellent, good, normal or poor, which impacts the rate at which the progress and quality bars fill. This random action bonus or penalty is referred to as the action's condition.

Starting crafters in *FFXIV* are at the mercy of conditions — without a few lucky successful actions while avoiding an unlucky failure, the progress bar cannot be filled and the craft fails. However, as players progress, they unlock more crafting skills which gives them more control over the crafting process. A 'poor' condition becomes less of a setback, an

'excellent' condition can be capitalized on. Likewise with the quality bar — at the start, players have very few ways of filling the quality bar, or can fill it, but at the expense of being able to fill the progress bar. As players progress, filling the quality bar becomes more feasible, and players can more consistently create items of high quality. This mastery of randomness is pleasing and powerful, and one case study into how Progression can work in a crafting system.

Recipe Definition with Randomness

While attempting to visualize the range of inputs that could be part of the Recipe Definition, Completion Constraints, and how they related to the variability of the outcome, we created Figure 16.
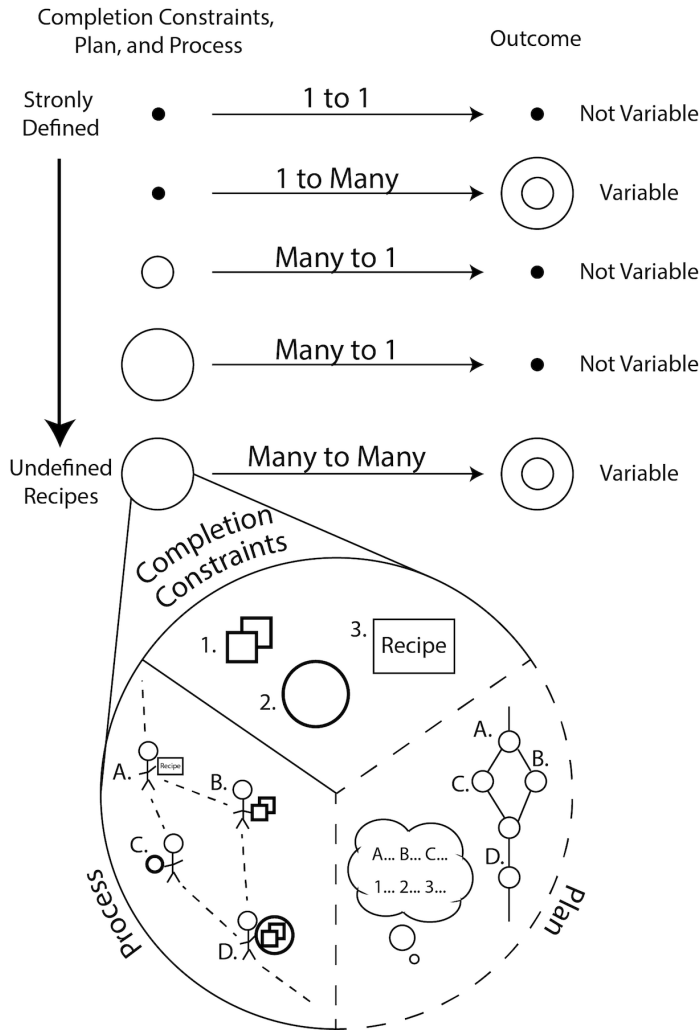
Figure 16. A generic summary of the relationship between crafting inputs and outputs, combining the concepts of different types of defined recipes, crafting inputs, player intention/design, and variable outcome. The left-hand circles start with strongly defined recipes and represent less and less defined recipes as the examples progress (their circles get bigger, representing a greater space of potential inputs). The right-hand outcome circles show that either the output results in a determined output (not variable), or a range of outputs, small or large, that result from some amount of variability. The bottom half of the diagram expands on what could be involved in crafting inputs: some amount of materials (completion constraints), the crafting process, and (optionally) an implicit or explicit plan to make or use the materials or output in a different context.

System Recognition of Outcome

System Recognition of Outcome refers to the manner and degree to which a larger game system models and represents the results of crafting. This is the reinforcement after player's crafting actions by which players know that they have crafted something; that something new exists in the virtual world as a result of their decision-making and actions. System recognition ranges from one-time, immediate graphical or textual acknowledgement, to complex behavioral modeling of the crafted item over time.

The level of system recognition of a crafting system may be estimated by answering these questions, adapted for crafting from Chris Crawford's listen-think-speak definition of interactivity [Crawford 2003].

1. How many properties of the resource constraints and crafted output does the system use to impact future game actions?
2. How much processing, or transformation, does the player perceive to be carried out on those properties? Do those properties change the final product or how the final product interacts with the greater game systems?
3. How deeply is the output of crafting represented to the player? Can the player perceive the transformation by some textual, graphical, or other signal expressed by the final product?

These questions culminate in how well a crafting system (1) listens to, (2) thinks about, and (3) speaks about what a player *is crafting* and has crafted. In Crawford's terms, it is how well the system holds up its end of the conversation with the player, after the player has "spoken." In this case, the player spoke through crafting actions, and the focus of conversation is on what the player has crafted. If the system looks at few properties of the output, processes them minimally or not at all, and presents that minimal processing directly, without impacting other game systems, it is considered to have low system recognition of outcome. If the system uses many properties of crafting output, connecting crafting output to non-trivial functions and other game systems, and presents a transformed representation and cascading consequences of the output back to the player, it has high system recognition.

Low Recognition

Among the lowest modes of recognizing player crafting is by direct, one-time acknowledgement by static text or image. In *Kittens Game*, crafting a stone slab results in a slab counter in a list of many other counters quietly increasing, or a new counter quietly appearing if it hasn't been crafted before (Figure 17). Such counter-based representation of crafting, in which the sole purpose of crafting objects is to increment counters which then serve as resources for crafting other objects (which are themselves just counters), is typical of the idle games (or incremental games) genre [Khaliq and Purkiss 2015] [Deterding 2016].

1. The system uses for recognition only a single property of a crafted item: that the item now exists. It doesn't need any other information about that item.
2. The only processing over that single property is the incrementing of a numerical inventory count of identical items that existed previously. In the context of gameplay action, the system propagates the consequence of that changed numerical value to other relevant systems (such as trade).
3. The only direct representation of the output is a changed textual counter.

The *indirect* representation of output through other game systems pushes *Kittens Game*'s crafting model higher than its otherwise minimal position on this dimension. It further recognizes player-crafted items by modeling them as ingredients in future crafting recipes, and as inputs to the game's economy, as goods that can be sold or traded.

Figure 17. Crafting outcomes in *Kittens Game* are recognized through minimal text representation and as input to future crafting and economic actions.

Medium Recognition

Contrast this representation with crafting whose output is given physical manifestation in the gameworld, beyond an inventory abstraction. In *Minecraft* crafting (versus block placement), many craftable items exist as icons in inventory slots, but may also be placed in the world or held and used by the player character. When a pickaxe is crafted, it appears as a static, 2D icon in the inventory menu, that may be dragged and dropped into different slots of a grid. It may then be selected to be equipped, at which point its 3D, voxel representation is visible in the player character's hand as she moves about the world. With pickaxe equipped, the player can click on blocks in the world, causing the pickaxe's 3D model to move and rotate with the avatar's arm, as if being swung. Each click and swing causes cracks to appear on the clicked block, growing more pronounced with each swing, until finally bursting into collectible resources if the pickaxe was crafted with a strong enough metal. Besides the all-important pickaxe, *Minecraft* crafting supports the creation of other equipable tools, weapons, armor, and several classes of placeable blocks and vehicles.

Minecraft easily has deeper recognition of crafting outcomes than *Kittens Game*, but less than games which, for instance, apply and visualize detailed physics simulations. We analyze this already complex middle case as follows:

1. The *Minecraft* crafting system recognizes several properties of its crafted items: item category (e.g., pickaxe, sword, wood plank) and several classes of ingredients, which are different for different item categories (e.g., for pickaxes, axes, and swords, the metal used matters; for wood planks, the kind of wood blocks used matters).
2. The system-observed properties of the crafted items are transformed and processed only slightly: for

stackable items, an inventory counter is increased, and for items with quality, that quality is compared to the quality of an action target to determine the action's outcome. When the player uses these different items, their properties are used in determining action outcomes (e.g., if mining with a crafted pickaxe, a block can burst into resources).

3. The crafted items are modeled as moveable graphical icons in a gridded inventory space, as 3D-modeled objects in the world. The material composition (and thus strength) of the item is 'spoken' to the player as different-colored versions of the same item corresponding to the color of the material.

High Recognition

At the highest end of this spectrum are games which save a lot of information about how an item is crafted and use that information as input to a simulation, which applies functions or rules to transform that saved information, and whose behavior may be observed during play. Such *simulation-based crafting recognition* games include *Besiege*, *Kerbal Space Program*, *SimCity*, *CodeSpells*, and *Spore*'s character creation.

In *Besiege*, players craft hyper-destructive siege machines from a large pallet of combinable engine parts, including beams, springs, flame-throwers, and motorized wheels. The resulting objects often look like they could be Medieval European war technology, but are crafted from atomic parts with a large space of potential combination. Players regularly create mechanisms never used in historical European warfare and that are in fact completely new to *Besiege* and its online community.

1. In *Besiege*, recognized properties of a crafted item consist of its types of component pieces (e.g., beams, springs, wheels), and how those pieces are arranged relative to each other in a 3D space (e.g., this particular beam is connected to this wheel in this orientation). The game system represents many physical properties of the constructed entity, excluding only information such as the exact sequence of actions or time the player took to construct it.
2. The recognized properties are transformed by putting their representation through *Besiege*'s stylized physics simulation. X- and y-positions of engine parts are modified by gravity, collision with other solid objects, input controls, and the behavior of connected parts and the type of connection.
3. The crafted item is well represented to the player. When the player completes a siege machine, they can press a key or button to "play" the scenario, which exits build mode, begins applying the physics engine to the machine as it is represented graphically, enables keyboard-controllable pieces to be controlled, and initiates the movement and targeting of tiny, defending NPC knights and archers toward the newly minted death machine.

Patterns of System Recognition

In games like *Besiege*, with particularly high outcome recognition, the core loop often emphasizes it via a separate, watch-it-run or simulation phase, following the crafting. Often, the phase makes the crafted item more traditionally playable, with movement-control or other minimal interactivity. True also of *Kerbal Space Program*, *Incredipede*, and *Spore* character creation, this affords the player the unique pleasure of seeing her creations theatrically come to life, in a performative reveal after a suspense-building crafting period. The player creates with an idea of what will happen, runs the simulation and inevitably finds her idea only partially correct, but in an understandable way, so that her idea of what will happen is updated to reflect a better understanding of the simulation. Her updated mental model is immediately actionable in the next crafting phase. The game-internal core loop is neatly translatable to the player-internal iterative model-making loop of the *SimCity* Effect [Wardrip-Fruin 2009]. Further, "live" recognition, that happens in real time, both as the item is crafted and afterward, as with *Minecraft* block placement, supports player embodiment as a character in the virtual world, and is more commonly integrated with larger gameplay experiences that aren't focused primarily on crafting. Furthermore, systems high on outcome recognition and low on recipe definition generally support the *SimCity* Effect (named for a live crafting system), regardless of the presence of a detached watch-it-run phase after crafting.

Other patterns of system recognition include graphical modeling and physics simulation (e.g., gravity, collision, fluid

physics), impact on a stats system (combat, character progression, NPC relationships), and adjacency bonuses in base or city-building (e.g. building a Theatre District next to a Natural Wonder in *Civilization 6* produces bonus culture). All of these can vary on system recognition by how flat or complex the model is.

Another interesting pattern is the use of outcome recognition to identify conditions-based recipes in an otherwise undefined recipe system. In *Terraria*, houses can be built for NPCs using the general block-placement system, anywhere in the large sandbox world. But NPCs will only become residents if particular requirements are met: blocks are placed to form a fully enclosed structure, with at least one entrance, covering at least 60 total tiles, but less than 750, containing at least one valid light source, flat surface item, and comfort item. The system will check for player structures which satisfy this *after* crafting, rather than impose the requirements before the structure can be built at all, as in strongly defined recipe crafting. This pattern is also visible in the gardening of *Viva Piñata*, in which specific clusters of player-placed items and piñata residents in a garden entice other piñatas to visit and become residents. This style of recognition enables both the creative experimentation and expression of freeform crafting as well as the authorial control of predefined recipes. When players discover these implicit recipes, without having looked them up or known about them beforehand, this style of system recognition can be a delightful and surprising meeting of minds between the player and system.

Games with undefined recipes often have high system recognition, as with *Besiege*, *Kerbal Space Program*, and *Minecraft* block placement, but this is not always the case. *Draw Something* has no recipes, yet low system recognition of the player-created drawings. The system doesn't listen and respond to player input beyond presenting the drawing on the screen as it was drawn. Instead, the finished drawings are recognized by a human partner, in an association game paradigm similar to *Pictionary* or charades. Player-recognized crafting is not mapped by this dimension, yet features importantly in Player Expressiveness.

Player Expressiveness

The dimension of player expressiveness captures the range of reasonable actions and outcomes within a crafting system suggested by the domain of the game world. "Expressiveness" is a somewhat vague term, as is "creativity" (See Craft and Creativity), but this dimension focuses on the amount of reasonable choice the player has, given the game's setting and the tools provided to them, during the process of crafting. This dimension is not trying to capture players explicitly subverting or exploiting a crafting system, as might occur in counter-play. Further, this dimension is not about the process of choosing what to craft, nor about the uses of the item after being made. For example, this dimension is about the process of customization and sewing of a shirt, not about choosing to wear the shirt (because it may be pretty or powerful), nor about how to dye, alter, or equip the shirt in a different crafting system after being completed. In short, we are ignoring the infinite combinatorial influences of outside systems as much as possible, but we acknowledge that an outside influence can inspire creative action or exploration. We do not care where inspiration comes from; this dimension only measures whether or not the player can express her inspiration successfully.

For example, in *Draw Something*, the player is given a one-word prompt to draw, and she is victorious if the other player guesses her prompt successfully. The player is free to draw anything she likes in a rainbow of colors, whether or not it supports the prompt. The user may draw an expansive beach scene, put Batman lounging in a beach chair with a coconut drink complete with tiny umbrella, and point to the little straw in the cup for the prompt "straw." While the brushes and drawing tools are extremely basic, players can use fingertips or a stylus to be as creative as they like or spend as much time on a drawing as they desire, and this results in extremely high player expressiveness.

In *Neko Atsume*, a game of "gardening" cats via putting down enticing furniture, players don't have the ability to pet, adopt, customize, or directly interact with the cats, which are common actions players may want to do with real or virtual cats. There is a lack of fulfillment of player desires in customizing their homes or interacting with their cultivated garden of adorable felines, which results in lower player expressiveness. A game world that suggests many creative options without offering the ability to capitalize on them has the lowest level of Player Expressiveness. For example, *Knitting Simulator 2014*, which emulates knitting extremely well, offers no ability to change, adjust, or customize yarn color, pattern, or stitches. All of the creative choice a crafter has when knitting is absent. The mechanics of the game distill

down to a progress bar, leaving no space for player agency and no support for player choice. In the terms of Mateas, this would be called an imbalance of material affordances (reasonable player desire) and formal affordances (game system support) [Mateas 2001], expanded in [Wardrip-Fruin et al. 2009]. In these examples, we show that high player expressiveness is a combination of a creative domain or space that begs for customization, as well as mechanical support for the player to create within that space.

Subversive Play

In real-life situations including games, open-ended play, or law, many humans have cheated, found loop-holes, and otherwise circumvented the rules or intended purpose of rules. Often, cheating leads to avoiding or skipping game challenges by amassing experience, automatically aiming at critical targets, or having unlimited health or lives. However, a subset of creative players cheat or find alternative modes of play within a given arena. Game scholars call these activities "critical" or "subversive" play [Flanagan 2009]. Game developers can try their hardest to snuff out cheaters, and often do in competitive games such as *Overwatch* or *DOTA,* where one player's cheating harms the play experience for another real-life human. Other developers may embrace and attempt to support subversive play, such as Will Wright, creator of the Sims and a fan of attaching toys to rockets [Totilo 2010]. The "motherlode" debug code, which grants the player a large sum of in-game currency, has existed in nearly every *Sims* game since *Sims II*. The *Sims* series of games has many interesting opportunities for subversive play, such as torturing and killing sims in countless ways, flirting with Death, and having babies with aliens [Reddit 2015]. Since choosing to play subversively is a direct choice of the player's, and a developer's support of those choices enables players to explore new and creative experiences, *games that explicitly support subversive play have increased player expressiveness*. This is distinct from unsupported subversive play, which, as we say above, we do not count towards player expressiveness in our analysis. There are very few examples of these games, however, and this is a rich area of potential future exploration.

Resources and Recipes

Different materials and recipes often spark new ideas or give avenues for fresh inspiration and methods of crafting. For example, many players of *Minecraft* do not immediately recognize that the positions of materials in the 2x2 and 3x3 crafting grids (discussed previously in Strongly Defined Recipes and Figures 5 & 19) are meaningful. However, once the player discovers this fact, if she doesn't already know the other recipes, there is an immediate suggested course of action discovered within the crafting grid. Regarding *Minecraft*'s other primary crafting system of block placement, different block materials can help crafters better suggest form in their low-resolution sculptures or apply other forms of art, such as pixel graphics. Not only do users have the range of naturally occurring material, such as stone or wood, but they can dye wool in a range of 16 colors. If we examine character customization in *Minecraft*, we also see that those 16 colors can be mixed or overdyed to create 12,326,391 colors[14].

Progression

Progression addresses the experience of interacting with the crafting system over time. Games that read higher on this dimension, such as those with deep crafting prerequisite trees or the leveling up of avatar crafting skills, afford a greater sense of growth and development. A crafting system with no system-underwritten progression allows any of the system's recipes or combination mechanisms to be engaged at any time during play, which enables more player self-direction and freeform experimentation.

This dimension is highly related to player skill at a particular crafting system. Crafting systems can be difficult, such as the complex calculations that go into a successful ship for *Kerbal Space Program*, or the planning and architectural design for large *Minecraft* creations. As people engage with the game systems, they master them, being able to better direct them towards a particular goal. However, this sort of progression is highly dependent on the person playing the crafting system rather than properties in the system itself. For example, a rocket engineer may build a successful ship in *Kerbal* far more quickly than the average player. Therefore, this dimension avoids addressing player skill and instead focuses on how the crafting system itself supports growth and development through engagement.

We identified five types of progression in crafting systems: leveling (improving an abstraction for crafting skill), resource

unlocking (getting new resources to use), recipe unlocking (finding new recipes), prerequisite crafting (needing to craft something before being able to craft something else), and mechanics changes (new ways to craft things).

Leveling

This type of progression is related to increasing the player-character's crafting skill. Most commonly, crafting skill is represented as an abstract number, and increasing this number is interpreted as an avatar becoming more skilled at a craft. Not all forms of leveling in a game pertain to crafting, or are relevant to crafting progression. In *Skyrim*, the player has many different stats to level that are abstractions over the avatar's skill in a variety of domains. Leveling up combat-focused skills (like one-handed weapon combat) does not impact the player's ability to craft in any way, while certain skills in *Skyrim* are centered around crafting (e.g., smithing).

There is another, less explicit form of crafting leveling. When looking at games like *Katamari Damacy* in terms of crafting, a player progresses through levels by building balls, called katamaris, to a certain size or with specific types of items. Because the level requirements of the player's katamari get more and more demanding, moving from level to level works as a form of crafting leveling, unlocking new resources to use in katamaris as well as proposing new recipes. The player may not think to make a katamari almost solely of school supplies, but the level requirements force the player to make conscious choices on what to pick up with their katamari. NPCs may comment, as a player moves into later levels or overshoots the level's requirement, that their avatar is becoming a better and better crafter, still feeding into the core idea of development and growth that is important to this part of this dimension.

Implicit or Explicit Resource Unlocking

Progression through a crafting system can work in more ways than increasing an abstraction for skill. Another common mechanism is revealing new resources for a player to use. These reveals can be explicit: a single flag that must be flipped or a condition that must be satisfied in order to reveal new resources. Often, the revealing condition will be brought to the player's attention, or the unveiling of a new resource will be announced to the player when it is available.

In other games, such as *Minecraft*, this happens implicitly by virtue of world building or level design. As players play *Minecraft*, they explore the world primarily by digging down through it. It takes tools built from rarer, deeper resources to dig through tougher, deeper kinds of rock. This is an implicit resource unlock — there is no flag that must be flipped by getting to a particular level or finishing a quest in order to unveil new kinds of ore to mine.[15] Instead, there is a natural barrier that can be overcome in a variety of ways, all of which require some time spent interacting with the game world. This natural progression through strata of the game world is systematically enforced through properties of the procedural content generator that generates Minecraft worlds.

Implicit or Explicit Recipe Unlocking

Just like resources can be explicitly or implicitly locked, so can particular recipes for crafting. This concept ties in heavily with the Recipe Definition dimension, as a game needs to have semi-defined or strongly defined recipes in order to unlock them. Explicitly locked recipes might be quest rewards, or given to the player if she achieves a certain level. Games that have implicit recipe unlocking may randomly distribute their recipes throughout the game world (like *No Man's Sky*, see Figure 18). There is nothing the game explicitly asks the player to do to unlock a particular recipe, outside of interacting with and exploring the game world.

Figure 18. A player picking up a new recipe in *No Man's Sky*. These recipes are scattered throughout the galaxy, requiring a player to explore the galaxy however she sees fit to unlock them.

This is also common in idle or incremental game crafting [Khaliq and Purkiss 2015] [Deterding 2016], and it may be explicit or implicit. New recipes in incremental games often appear as new buttons (new ways to convert resources) or tabs (entire banks of new buttons or existing functionality on old buttons). Some recipes may be implicitly unlocked by just spending a certain amount of time with the incremental game, while others are explicitly locked with a certain amount and combination of resources required.

Prerequisite Crafting

Prerequisite crafting is when there is an explicit requirement for something to be crafted before something else can be crafted. This is the type of progression in crafting trees, which are long chains of prerequisite objects that usually terminate in a single item. A good example of this are the crafting trees of *Terraria*, which often have players crafting a good many items before being able to craft a final product, as shown in Figure 12.

Games with strongly defined recipes are better equipped to have crafting trees. As the ingredients are explicitly laid out in front of the player, it is easy for a player to see how crafting one item allows for the ability to craft a second. It also often fits naturally within the game world — many defined recipes are for more refined raw materials that will be used to craft more complex objects at the roots of crafting trees.

Not all prerequisite crafting is a crafting tree; sometimes a player needs to craft an item before a particular new set of crafting mechanics are available, or a crafted item might be the way to explicitly unlock a recipe or resource.

Mechanics Changes

Sometimes games provide progression by revealing more mechanics to the player as she crafts. This is highly common in games that have multiple crafting subsystems, as they tend to be introduced gradually, waiting until a player is comfortable with one before moving on to the next. There is no requirement that this be the case, and certainly games running a god or sandbox mode reveal all of the crafting abilities to the player at once. That is, what a player may usually experience as gradual changes can be presented as not being changed, but we consider them mechanics changes as they are still comprised of different sub-systems.

In *Minecraft*, players start with a very small set of abilities. After punching down their first tree, they can create a workbench which vastly expands the set of available crafting options. As they progress, they can eventually create a brewing stand, which unlocks a different set of crafting mechanics for creating potions, and an enchantment table, for crafting enchanted items (Figure 19).
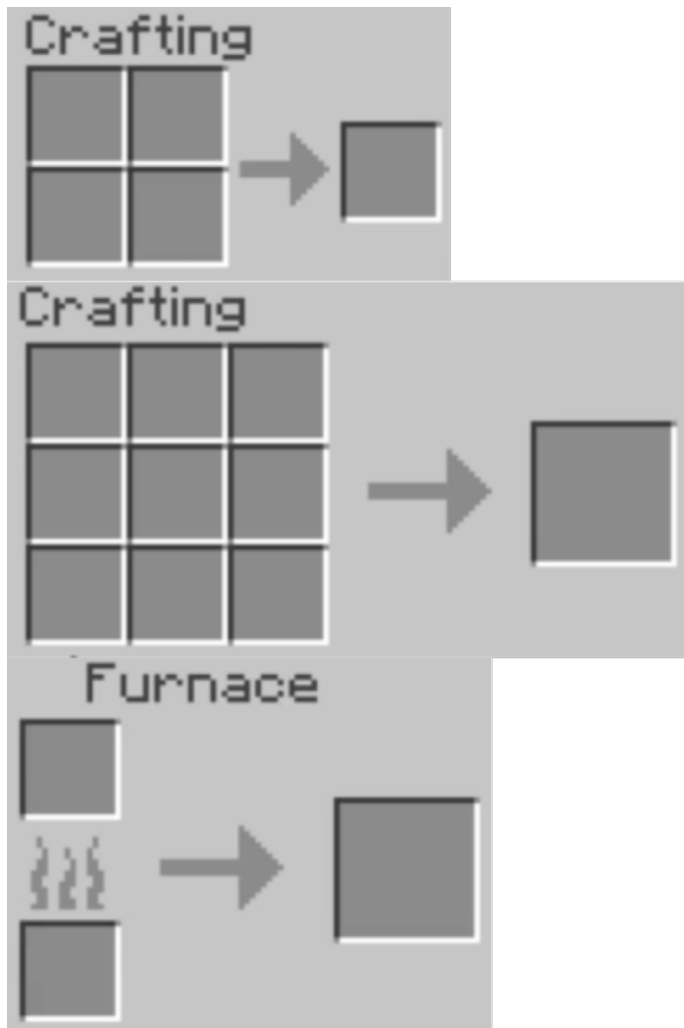
Figure 19. Some of the various crafting stations in *Minecraft*. Each interface has its own mechanics, which allows players to make more items and also gives a sense of mastery.

In *Final Fantasy XIV*, the crafting professions are individually leveled. At level 13, players learn "Observe" which is a skill that does literally nothing. At this higher level, and with this skill, players are prompted to learn about item conditions, which they have had no control over and no reason to care about up until now (Figure 15). Players learn how to optimize quality gains when the condition of items is "excellent" and use "Observe" when the condition is "poor," minimizing quality loss.

Goals in Crafting

Progression is often tethered to goal-setting and what a player wants to achieve. For example, a player who wants to

become the best crafter as quickly as possible will likely craft different items than one who wants to craft for the highest amount of profit. People may follow long, winding and very different paths through crafting trees to get vastly different final items. Tying into Social Aspects, as part of progression, crafters may specialize, opening up new mechanics, recipes and/or resources, but being gated out of other mechanics, recipes and/or resources.

A unique case of a crafting system as a driver for narrative progression, *Dark Cloud*'s town-building puzzle heavily motivates the player toward one specific goal of restoring the town to how it used to be. The player is free to place buildings, bits of road, and water wherever she likes, and to fill the houses with whomever and whatever she likes. However, for example, if the protagonist's main window doesn't face eastward and if the house doesn't include a keg, chimney, upstairs storage, Renee (the protagonist's mother), a llama, and a stray cat, then the player cannot receive the second party member, Xiao. Without the proper party members, dungeon bosses and mechanics can be extremely difficult, if not impossible. Progression is gated by not just crafting, but by building the town to be precisely how the NPCs desire. This is also an interesting case study for Player Expressiveness as the player must navigate a tension between her aesthetic crafting goals and the systematically-enforced desires of NPCs.

Each dimension, on its own, covers just a facet of the full experience of a crafting system. In some cases, we've discussed how interplay between dimension scores affords different ways to view a crafting system. This is only a small part of how different dimension settings work together to describe the experience of crafting in a game. We hope that future work can dive deeper into dimension interplay.

Beyond the Dimensions

With all the dimensions presented in this paper, there is still much more that remains to be explored within the broad definition of crafting systems we outlined earlier. However, the following sections, while important to many crafting systems, were too narrow or too complex and multifaceted to classify as a common attribute representable with a single numerical value. However, in the games we analyzed, the following concepts of automation and social involvement offered notable parallels to real-world crafting.

Automation

Crafting systems are sometimes focused on the single production of an item, letting players imbue that item with meaning and care. However, some games let players mass produce items, or let large chunks of the system run automatically with no player input for extended periods of time. In short, some games give players the ability to automate their crafting systems.

Automation starts to push a crafting system into a different experience than one often associated with real-world hand-crafting. Going from an intimate, item-by-item experience in games like *Cooking Mama* where each step of the crafting process is a rich minigame, to a large-scale operation, powered by automated actions or helper systems like a redstone powered[16] mining operation in *Minecraft,* changes how the player crafts. We can look at both of these styles in terms of how various positions on the dimensions help afford automation.

Automation affordances score low in the Fidelity of Action dimension. High fidelity crafting systems often have highly detailed, complex, (virtually) embodied actions that take time and focus for a player to complete. Automation pushes against this, as it tries to afford an experience that's focused on producing and handling crafting outcomes at a large scale, rather than embodied action. A way to look at large scale crafting is that a crafter is creating an operation, rather than the items. Massively multiplayer online game (MMOs) crafting systems often invoke similar feelings, as Robert Rice points out, "More often than not the process is unrewarding and more like a highly repetitive mass production and manufacturing of items" [Rice 2006, 139]. We do not make any claims about MMO crafting feeling rewarding or not, but do agree that MMO crafting often has crafters create large quantities of similar items.

It is unsurprising, then, that MMOs allow for macroing. Macroing can afford automation while maintaining a higher degree of player fidelity. A macro is a small program, built within game provided tools, that can automate some character actions. As a defense against botting, or having entirely autonomously controlled agents within a game, it is

rare to see program flow control structures or loops as part of a game-provided toolkit for macroing. However, some games have historically afforded loops and if statements in their macro systems; for example, *Star Wars Galaxies* had self-callable macros, which affords repetition like loops through recursion. With a rich macroing language, a game can have a higher degree of fidelity while still allowing some players to automate the process.

Games with strongly defined recipes (see Recipe Definition) often support automation. Because a recipe has a known, fixed number of input components, recipes can be made in batches. Fixed recipes also help with macro design and creation, as the actions that lead to success are listed to the player and encoded in the system. However, even games with no recipe definition can afford automation. 136

Minecraft block generators are player-crafted machines that produce resources for players to perform more block-placement crafting (Figure 20). The machines themselves are created by taking advantage of *Minecraft*'s physics engine and placing particular blocks in particular locations (*Terraria* has similar player creations, see Out-of-Game, below). Although this doesn't afford automation of the actual crafting process (in terms of block placement), it still works as an inroad into working with crafting at a large scale. It also does take some crafting know-how and skill to set up a block generator in the first place. 137

Figure 20. A running cobblestone block generator in *Minecraft*. This sort of work does not automate the crafting process, unlike macro engines, but shows how freeform crafting can lead to creations that 'craft' on their own, or automate other parts of the game.

Variable Outcome is problematic when it comes to automation, especially when we reflect back on real-world crafting. On one hand, games that read high on this dimension often work as motivation for why a player might want to automate the crafting process. A player searching for an optimal item in a game with a lot of randomness in crafting might need to 138

craft the same item 20, 50 or even 100 times. It is not hard to imagine why such a player might want autonomous tools. On the other hand, Variable Outcome thwarts the regulation that is associated with mass production, automation and the workmanship of certainty. Crafting systems with high variable outcome have large fluctuations in outcome quality even with automated ways to go about the crafting process. Crafting systems high in variable outcome have a reason to add automation affordances to their crafting systems to help mediate the monotony of searching for a highly desirable outcome. However, this process of using automation to burn through bad results is a different experience from how automation functions in real-world crafting, where automation and the workmanship of certainty prevent particular bad results from ever occurring.

So far, we have only discussed the production of a large quantity of in-world items. Along with automating the crafting process to allow for players to produce a large quantity of items, games also afford a player to inspect, handle, and quantize a large amount of crafted output. These sorts of systems are a bit beyond the scope of this paper, but it is worth remembering that crafting systems do not exist in a vacuum. [139]

Economic systems like *World of Warcraft*'s auction house, combined with third party analysis tools (e.g., wowuction.com) that measure things like the amount of a particular item currently being sold, its local average sale price versus a historical sale price and the amount of currency currently in circulation, allow players to make informed decisions about the large number of items a macro-wielding worker may create. [140]

Social Aspects

Community is a fundamental component of not only the history of crafting, but also the modern resurgence of crafts and crafting in games. However, the wide range of amount and types of social involvement in different games, both in-game and out-of-game, made the concept too awkward to include as its own dimension of crafting systems. The social examples we use here are directly applicable to the embedded crafting systems within the games cited and may be generalized to other games or genres. [141]

In-Game

Crafting systems with in-game social involvement necessarily need some amount of multiplayer connection, either as a MMO-type game or with the ability to share in-game entities with other players. Within these games, players can make commissions of other players personally, as well as trade their crafted goods in in-game auction houses, or simply collect or curate their favorite designs. For example, MMOs such as *World of Warcraft* often make use of consumable materials like potions that improve the player's abilities for a short time and are needed regularly by active players. *WoW* also makes use of communal improvements, such as cauldrons of flasks from the *Burning Crusade* and *Legion* expansions, or food carts from the *Mists of Pandaria* expansion and "feasts" from the *Wrath of the Lich King* expansion onward. [142]

In some MMOs with many professions, different professions require materials from each other, and a single player may need to make another character with that profession, or be forced to buy or trade for materials with other players. Some players of MMO games that include a free market economy spend all their time manipulating the market or taking advantage of changes in the ever-updating game. Players may choose to make their money mass-producing ingredients for other crafters (see Automation), or buying all items of a type and owning part of the market. Crafters in real life struggle with pricing and selling their goods as well, whether in-person during craft fairs or on websites such as *Etsy* (see The Evolution of Crafts). [143]

There are a few interesting single-player examples either modeling social behavior for crafters or cleverly sharing in-game entities without embodied trade or currency at all. *Recettear: An Item Shop's Tale* is a game about crafting an item shop, including gathering your own stock items, curating your shop's atmosphere (by changing the walling and flooring), and haggling with customers. Customers have a reputation with the player character which increases their budget, so short-term discounts may lead to long-term profits. Not many games bother with modeling these social behaviors, which leaves this space very under-explored. Another game, *Spore*, managed to share its wide variety of creatures between players without a connection between them by saving creature data in a PNG image. The [144]

Sporepedia hosts, among other items, a gallery of images that any *Spore* player can grab and upload as a fully functioning creature using only 8KB of data. However, to avoid intentional hacking, the creators of *Spore* obfuscated how the data was encoded within the image.

Out-of-Game

There is no limit on how players connect outside of games. Many games have entries in *GameFAQs*, a website with fan-made walkthroughs, guides, maps, save files, Q&A, and a forum where fans can post polls or pose deeper questions for discussion. *Reddit* is another common communal area for unstructured game questions and strategy discussion. More popular games have either company-hosted or fan-curated Wiki pages, such as Sporepedia, which include official-looking guides with images and hyperlinks defining everything from game concepts to crafting trees and character stats. Fan-curated content may also include community strategies and sometimes exploits that the creators of the game may monitor for future updates.

In general, these websites offer players a means to share details about a game's crafting system, such as *Minecraft*'s crafting trees, where often no such resource exists in-game. Players can also offer player-made recipes for complex structures, such as *Terraria*'s obsidian farm, which generates a highly-valued resource every player wants but is difficult to obtain. The sharing of recipes and techniques is an exact parallel to real-life crafts. There is a massive amount of craft tutorial videos on *YouTube*, as well as crafting pattern sharing or hosting websites such as *Craftster*, *Ravelry*, and *Craftsy* (see The Evolution of Crafts).

In-and-Out-of-Game

The rise of *Facebook* games in the 2000's rocked the gaming industry, where *Farmville* brought Zynga and casual games to the forefront. Using psychological skinner box techniques like those described as "dark patterns" by Lewis [Lewis 2014], *Farmville* reached critical mass by motivating its players to share, poke, prod, and harass their real-life friends for in-game rewards. The more active friends you had, the more materials you had to craft with, and the better you were at the game. Many people made multiple accounts or simply made a separate account for gaming altogether, collecting hundreds and thousands of fellow gamers as friends. *Farmville*'s model of mixing out-of-game viral marketing and in-game rewards became the foundation of a whole new genre of games, many of which include crafting, such as *Castleville*.

Both automation and social play go beyond just crafting, showcasing an instance of a separate system to afford different play experiences in a crafting system (automation) and an instance of of how crafting can work with other game systems to afford different play experiences (social play). These lines are fuzzy, as automation often ties into in-game economies, which are primarily a social phenomenon. We hope that this section motivates future inspection in game crafting. This is not a complete listing of all ideas that touch on crafting, but showcase some concepts around potential future discussion on crafting in games.

Conclusion

Within games journalism and design, there is not a strong definition of what crafting is or means in games. We've proposed a definition of crafting and shown a seven-dimensional framework that highlights important features of crafting across a myriad of games and systems. In Recipe Definition we separated games around the notion of a recipe and how free-form the crafting experience may be. For the Fidelity of Action dimension, we examined how detailed and performative player actions are, and how closely they correspond to any real-world crafting equivalents. We enumerated the types of constraints a game puts on completing a craft in Completion Constraints. We addressed randomness and the amount of control a player has over the outcome of a crafting attempt in Variable Outcome. We analyze how richly a system understands and models the outcome of a crafting attempt in System Recognition of Outcome. In Player Expressiveness, we examined the range of creative actions available in crafting domains, and how they are supported by material and formal affordances. Finally, for the Progression dimension, we enumerated the types of advancement the player may achieve from interacting with the crafting system over time.

Each of these dimensions is important in its own right, and we have shown through many examples that there is no one 150 set of scores on these dimensions that represent a 'canonical' or ideal crafting system. This framework is remarkably flexible, providing a useful lens for games that might not fall under the traditional umbrella of crafting, but which have crafting elements. We looked at various crafting-like actions in games, from placing blocks in *Minecraft* to designing spacecraft in *Kerbal Space Program*. Although not covered in detail in this paper, we've even looked at crafting-like systems in borderline cases of crafting, ranging from link exploration in *With Those We Love Alive* to katamari accretion in *Katamari Damacy*. The affiliated web visualization showcases these extra examples.

We hope that this paper helps build a critical vocabulary around crafting systems, and helps start discussions and shed 151 light on this common concept in games.

Games Cited

5th Cell. (2012). *Scribblenauts Unlimited*. [Android, Wii U, Nintendo 3DS, iOS, Microsoft Windows], Nintendo, Warner Bros. Interactive Entertainment.

Angel, R. (1985). *Pictionary.* [Board Game], Angel Games Inc, Hasbro.

Barone, E. (2016). *Stardew Valley.* [PlayStation 4, Xbox One, Nintendo Switch, Microsoft Windows, Linux, macOS], Chucklefish Games.

Bethesda Game Studios. (2006). *The Elder Scrolls IV: Oblivion.* [Microsoft Windows, PlayStation 3, Xbox 360], Bethesda Softworks.

Bethesda Game Studios. (2011). *Skyrim.* [Microsoft Windows, Playstation 3, Xbox 360, Playstation 4, Xbox One], Bethesda Softworks.

Bethesda Game Studios. (2015). *Fallout IV.* [Microsoft Windows, PlayStation 4, Xbox One], Bethesda Softworks.

BioWare. (2011-). *Star Wars: The Old Republic.* [Microsoft Windows], Electronic Arts.

BioWare. (2014). *Dragon Age: Inquisition.* [Microsoft Windows, PlayStation 3, PlayStation 4, Xbox 360, Xbox One], Electronic Arts.

Blizzard Entertainment. (1998). *StarCraft.* [macOS, Microsoft Windows, Nintendo 64], Blizzard Entertainment.

Blizzard Entertainment. (2002). *Warcraft III.* [Mac OS, Microsoft Windows], Blizzard Entertainment.

Blizzard Entertainment. (2004-). *World of Warcraft.* [Microsoft Windows], Blizzard Entertainment.

Blizzard Entertainment. (2012). *Diablo III.* [Microsoft Windows, macOS, PlayStation 3, Xbox 360, PlayStation 4, Xbox One], Blizzard Entertainment.

Blizzard North. (2000). *Diablo II.* [Microsoft Windows, Classic Mac OS, macOS], Blizzard Entertainment.

bloodrizer. (2014). *Kittens Game.* [Web Browser].

Carbine Studios. (2014). *Wildstar.* [Microsoft Windows], NCSOFT.

CCP Games. (2003-). *EVE Online.* [Microsoft Windows, macOS], CCP Games.

Cooking Mama Ltd. (2006-2009). *Cooking Mama.* [Nintendo DS, iOS], Taito Corporation. Series.

Cooking Mama Ltd. (2010). *Crafting Mama.* [Nintendo DS], Square Enix.

EasyGameStation. (2007). *Recettear: An Item Shop's Tale.* [Microsoft Windows], Carpe Fulgur.

Game Freak. (1996-). *Pokémon Red*, *Pokémon Blue*, *Pokémon Green*, *Pokémon Yellow: Special Pikachu Edition*, *Pokémon Gold*, *Pokémon Silver*, *Pokémon Crystal*, *Pokémon Ruby*, *Pokémon Sapphire*, *Pokémon FireRed*, *Pokémon LeafGreen*, *Pokémon Emerald*, *Pokémon Diamond*, *Pokémon Pearl*, *Pokémon Platinum*, *Pokémon HeartGold*, *Pokémon SoulSilver*, *Pokémon Black*, *Pokémon White*, *Pokémon Black 2*, *Pokémon White 2*, *Pokémon X*, *Pokémon Y*, *Pokémon Omega Ruby*, *Pokémon Alpha Sapphire*, *Pokémon Sun*, *Pokémon Moon*. [Gameboy Color, Gameboy Advance, Nintendo DS, 3DS Virtual Console], Nintendo, The Pokémon Company. Series.

Gust. (2010). *Atelier Totori.* [PlayStation 3, PlayStation Vita], Gust.

Gust, Amazing, Banpresto, Bothtech, E3 Staff, GMO Mobile, Hyde, Idea Factory, Otomate, Rideon Japan, Tose. (1997-2016). *Atelier.* [PlayStation, PlayStation 2, PlayStation 3, PlayStation 4, PlayStation Network, PlayStation Portable, PlayStation Vita, PlayStation TV, WonderSwan Color, Game Boy Advance, Nintendo DS, Nintendo 3DS, Dreamcast, Sega Saturn, Microsoft Windows, Android, EZ, GREE, i, iOS, S!], Tecmo Koei, Gust. Series.

Harmonix, Pi Studios. (2007). *Rock Band.* [Xbox 360, PlayStation 3, PlayStation 2, Wii], MTV Games, Electronic Arts.

Hello Games. (2016). *No Man's Sky.* [PlayStation 4, Microsoft Windows], Hello Games.

Hit-Point. (2014). *Neko Atsume.* [iOS, Android].

Klei Entertainment. (2013). *Don't Starve.* [Microsoft Windows, macOS, Linux], Klei Entertainment.

Konami. (2005). *Lost in Blue I.* [Nintendo DS], Konami.

Level-5. (2000-2001). *Dark Cloud.* [PlayStation 2], Sony Computer Entertainment.

Marvelous Entertainment. (1996-2016). *Harvest Moon.* [Super Nintendo Entertainment System, Game Boy, Game Boy Advance, Game Boy Color, Nintendo 3DS, Nintendo 64, Nintendo DS, GameCube, PlayStation, PlayStation 2, PlayStation Portable, Wii], Natsume. Series.

Matrix Software. (2007). *Lost in Blue II.* [Nintendo DS], Konami.

Matrix Software. (2007). *Lost in Blue III.* [Nintendo DS], Konami.

Maxis. (2008). *Spore*. [Microsoft Windows, macOS, iOS], Electronic Arts.

Maxis. (2013). *SimCity*. [Microsoft Windows, macOS], Electronic Arts, Maxis.

Maxis, The Sims Studio. (2010). *The Sims 3.* [Microsoft Windows, macOS], Electronic Arts.

Maxis, The Sims Studio. (2014). *The Sims 4.* [Microsoft Windows, macOS], Electronic Arts.

Mojang. (2011-). *Minecraft.* [Microsoft Windows, macOS, Linux, Android, iOS, Windows Phone, Xbox 360, Xbox One, PlayStation 3, PlayStation 4, PlayStation Vita, Raspberry Pi, Universal Windows Platform, Wii U, Nintendo Switch, Apple TV], Mojang, Microsoft Studios, Sony Computer Entertainment.

Monomi Park. (2016). *Slime Rancher.* [Xbox One, Microsoft Windows, Linux, macOS], Monomi Park.

Multi-Dimensional Games. (2015). *CodeSpells.* [Windows, macOS], ThoughtSTEM.

Namco. (2004). *Katamari Damacy.* [PlayStation 2], Namco.

Nintendo. (2001-2016). *Animal Crossing.* [Nintendo 64, iQue Player, GameCube, Nintendo DS, Wii, Nintendo 3DS, Wii U], Nintendo. Series.

Northway Games. (2012). *Incredipede*. [Android, iOS, Adobe Flash Player, Microsoft Windows, Linux]. Northway Games.

OMGPop. (2015). *Draw Something.* [iOS, Android, Windows Phone], OMGPop.

PopCap Games. (2009). *Plants vs. Zombies.* [Web Browser, Microsoft Windows, macOS, iOS, Xbox Live Arcade, PlayStation Network, Nintendo DS, DSiWare, Bada, Android, Windows Phone, PlayStation Vita, BlackBerry Tablet OS, BlackBerry 10], PopCap Games.

Porpentine. (2014). *With Those We Love Alive.* [Web Browser].

Rare. (2006). *Viva Piñata.* [Xbox 360, Microsoft Windows], Microsoft Game Studios.

Re-Logic, Engine Software. (2011). *Terraria.* [Microsoft Windows, Xbox 360, PlayStation 3, PlayStation Vita, iOS, Android, Windows Phone, PlayStation 4, Xbox One, Nintendo 3DS, Wii U], 505 Games, Headup Games.

Smilebit. (2000). *Jet Set Radio*. [Dreamcast, Java ME, Game Boy Advance, Microsoft Windows, Playstation 3, Xbox 360, Playstation Vita, iOS, Android], Sega.

Sonic Team. (1998). *Sonic Adventure.* [Dreamcast, GameCube, Microsoft Windows], Sega.

Sonic Team. (2001). *Sonic Adventure II.* [Dreamcast, GameCube, PlayStation 3, Xbox 360, Microsoft Windows], Sega.

Sony Online Entertainment, Electronic Arts. (2003-2011). *Star Wars Galaxies.* [Microsoft Windows], LucasArts.

Spiderling Studios. (2015). *Besiege.* [Windows, macOS, Linux], Spiderling Studios.

Squad. (2015). *Kerbal Space Program*. [Linux, Microsoft Windows, macOS, PlayStation 4, Xbox One, Wii U], Squad.

Square. (2002). *Kingdom Hearts.* [PlayStation 2], Square.

Square Enix. (2010). *Final Fantasy XIV.* [Microsoft Windows], Square Enix.

Square Enix Product Development Division 1. (2005). *Kingdom Hearts II.* [PlayStation 2], Square Enix.

Stone, K. & DaRienzo, G. (2014). *Knitting Simulator 2014*. [Custom Hardware].

Xaviant. (2014). *Lichdom: Battlemage.* [Microsoft Windows].

Zynga Dallas. (2011-2016). *CastleVille.* [Adobe Flash], Zynga.

Notes

[1] While videogames and other computational media enable different kinds of systems modeling and moment-to-moment interaction patterns, crafting also appears in physical and tabletop games. These are equally applicable to our model of crafting systems.

[2] Digitally augmented crafting refers to traditional, non-virtual crafting aided by digital tools (e.g., software for designing knitting patterns).

[3] Idle games are a recent genre of games without much game studies research. See [Khaliq and Purkiss 2015] and [Deterding 2016] for more details.

[4] It requires progressively more ingredients to craft an item in *Kittens Game* every time that item is crafted.

[5] Back-up hosting for this visualization can be found at: http://www.creweltech.com/craftsystems/ and https://users.soe.ucsc.edu/~agrow/craftsystems/

[6] Games that only contain crafting in their theme or visual style, rather than player actions, are not considered to have crafting systems, and are therefore not explored as examples for our crafting systems framework.

[7] For more information on the use of entity and its context in game worlds, see [Zagal et al. 2007]

[8] The creation process does not include, for example, player gathering of ingredients beforehand.

[9] Representation within the game world removes game modding from consideration as crafting in games.

[10] According to Wardrip-Fruin et. al., agency can be considered as a result of not just the player, but the player and the game [Wardrip-Fruin et al. 2009]. In this case, a relaxed recipe does not necessarily increase agency without the game supporting and enabling that freedom with mechanics.

[11] For more information on affordances in games, including game narrative, see [Mateas and Stern 2006]

[12] *CodeSpells* is in beta as of the writing of this paper and is subject to radical change. For example, the designers may remove the ability to look at the underlying code behind the visual language before release.

[13] This is true from when *Pokémon* breeding was introduced in *Pokémon Gold/Silver* up until *Pokémon X/Y*. *Pokémon Sun/Moon* has a sauna, where players can leave eggs for breeding. Hatch time, for eggs in the sauna, is measured in real-world time, much like *Eve Online* manufacturing.

[14] Statistic and equations that calculate color from minecraft.gamepedia.com/Dye

[15] This is broadly true for *Minecraft*, until players decide to go to the Nether, which has an explicit locking mechanism of building a Nether portal. The Nether has several unique resources in it.

[16] Redstone is a special type of brick in *Minecraft* that allows for simple logic gates to be encoded in-game. This forms the basis for incredibly complex automatic *Minecraft* machinery.

Works Cited

Adams and Dormans 2012 Adams, E., & Dormans, J. *Game mechanics: advanced game design.* New Riders. (2012).

Bartle 1996 [Bartle 1996] Bartle, R. "Hearts, clubs, diamonds, spades: Players who suit MUDs." *Journal of MUD research.* (1996).

Callois 1961 Caillois, R. *Man, play, and games.* University of Illinois Press. (1961).

Compton 2015 Compton, K., & Mateas, M. "Casual Creators." ICCC. (2015).

Consalvo 2009 Consalvo, M. "There is no magic circle." *Games and culture.* (2009).

Crawford 2003 Crawford, C. "Chris Crawford on game design." New Riders. (2003).

Csikszentmihalyi 1996 Csikszentmihalyi, M. *Flow and the psychology of discovery and invention.* New York: Harper Collins. (1996).

Deterding 2016 Deterding, S. "Progress Wars: Idle Games and the Demarcation of "Real" Games." In joint *DiGRA* and *FDG* (2016).

Deterding et al. 2011 Deterding, S., Dixon, D., Khaled, R., & Nacke, L. "From game design elements to gamefulness: defining gamification." In *Proceedings of the 15th international academic MindTrek conference: Envisioning future media environments ACM.* (2011): 9-15.

Dormer 1997 Dormer, P. *The culture of craft.* Manchester University Press. (1997).

Ellington et al. 1982 Ellington, H., Addinall, E., Percival, F. *A handbook of game design.* Kogan Page. (1982).

Flanagan 2009 Flanagan, M. *Critical Play: Radical game design.* MIT Press. (2009).

Frank 2016 Frank, A. "Here's how The Division's next update is changing crafting and loot drops." *Polygon.* Vox Media. (2016).

Gauntlett 2013 Gauntlett, D. *Making is connecting.* John Wiley & Sons. (2013).

Item Crafting 2017 "Item Crafting." TV Tropes. (Accessed 2017).

Juul 2005 Juul, J. *Half-real: Video games between real rules and fictional worlds.* MIT press. (2005).

Kafai et al. 2008 Kafai, Y. B., Heeter, C., Denner, J., & Sun, J. Y. *Beyond Barbie [R] and Mortal Kombat: New Perspectives on Gender and Gaming.* MIT Press (BK). (2008).

Khaliq and Purkiss 2015 Khaliq, I., & Purkiss, B. "A study of interaction in idle games & perceptions on the definition of a game." In *Games Entertainment Media Conference (GEM)*, IEEE (2015).

Klabbers 2003 Klabbers J. H. "The gaming landscape: a taxonomy for classifying games and simulations." In DiGRA Conference (2003).

Letzter 2015 Letzter, R. "Online communities are changing video games to make them better, weirder, and much more wonderful." *Business Insider.* (2015).

Leung et al. 2014 Leung, F., Borst, K. J., & Ganetakos, J. B. U.S. Patent No. 8,836,719. Washington, DC: U.S. Patent and Trademark Office. (2014).

Levine and Heimerl 2008 Levine, F., & Heimerl, C. *Handmade nation: The rise of DIY, art, craft, and design.* Princeton Architectural Pr. (2008).

Lewis 2014 Lewis, C. *Irresistible Apps: Motivational Design Patterns for Apps, Games, and Web-Based Communities.* 1st ed. Apress, Berkely, CA, USA. (2014).

Mateas 2001 Mateas, M. "A preliminary poetics for interactive drama and games." *Digital Creativity,* 12(3). (2001): 140-152.

Mateas and Stern 2006 Mateas, M., & Stern, A. "Interaction and narrative." *The game design reader: A rules of play anthology*, 1 (2006): 642-669.

Meier 2012 Meier, S. "Interesting Decisions." Firaxis Games. Game Developer's Conference (GDC). (2012).

Mould 2014 Mould, O. "Tactical urbanism: The new vernacular of the creative city." *Geography Compass.* (2014): 529-539.

Myers 2009 Myers, D. "In search of a minimalist game." *Breaking New Ground: Innovation in Games, Play, Practice and*

Theory. Proceedings of DiGRA. (2009): 1-4.

Naasz 2013 Naasz, J. "Crafting Success: A Deeper Look at CastleVille's Crafting System." Zynga. Game Developer's Conference (GDC). (2013).

Ondrejka 2004 Ondrejka, C. "Escaping the gilded cage: User created content and building the metaverse." NYL Sch. L. Rev. 49. (2004): 81.

Paulk 2006 Paulk, C. "Signifying play: the Sims and the sociology of interior design." *Game Studies*, 6(1). (2006).

Pye 1968 Pye, D. *The nature and art of workmanship.* Cambridge UP. (1968).

Reddit 2015 "What is the worst thing you've ever done in The Sims series?" Reddit. (2015).

Rice 2006 Rice, R. "MMO Evolution." Lulu.com. (2006).

Rose 2014 Rose, M. "Infinifactory and the next generation of the 'Minecraft genre.'" *Gamasutra.* (2014).

Runco and Jaeger 2012 Runco, M. A., & Jaeger, G. J. "The standard definition of creativity." *Creativity Research Journal*, 24(1). (2012): 92-96.

Shaw 2015 Shaw, A. *Gaming at the edge: Sexuality and gender at the margins of gamer culture.* Univ Of Minnesota Press. (2015).

Shubik 1983 Shubik, M. "Gaming: A state-of-the-art survey." In I. Ståhl (ed) *Operational Gaming*, Oxford (1983), pp. 15-22.

Smaldone et al. 2016 Smaldone, R. A., Thompson, C. M., Evans, M., & Voit, W. "Teaching science through video games." *Nature Chemistry.* (2016).

Smith et al. 2011 Smith, A. M., Lewis, C., Hullett, K., Smith, G., & Sullivan, A. *An inclusive taxonomy of player modeling.* University of California, Santa Cruz, Tech. Rep. UCSC-SOE-11-13. (2011).

Steinke 2015 Steinke, Matthew. "Witchcraft: The Alchemy of a Crafting-Based Economy." CD PROJEKT RED. Game Developer's Conference Europe. (2015).

Stoller 2003 Stoller, D. *Stitch'n bitch: The knitter's handbook.* Workman Publishing. (2003).

Tanenbaum et al. 2013 Tanenbaum, J.G., Williams, A.M., Desjardins, A. & Tanenbaum, K. "Democratizing technology: pleasure, utility and expressiveness in DIY and maker practice" In *SIGCHI Conference on Human Factors in Computing Systems.* ACM. (2013): 2603-2612.

Thompson 2003 Thompson, C. "The Sims: Suburban Rhapsody." *Psychology Today.* (2003).

Tosca 2000 Tosca, S. P. "A pragmatics of links." In *Proceedings of the eleventh ACM on Hypertext and hypermedia.* ACM. (2000): 77-84.

Totilo 2010 Totilo, S. "Barbie Doll Destroyer Will Wright Visits Toy Fair, Hints Toy Plans." Kotaku. Gizmodo. (2010).

Tychsen and Hitchens 2009 Tychsen, A., & Hitchens, M. *Game Time Modeling and Analyzing Time in Multiplayer and Massively Multiplayer Games. Games and Culture*, 4(2). (2009): 170-201.

Wardrip-Fruin 2009 Wardrip-Fruin, N. *Expressive Processing: Digital fictions, computer games, and software studies.* MIT press. (2009): 299-352.

Wardrip-Fruin et al. 2009 Wardrip-Fruin, N., Mateas, M., Dow, S., and Sali, S. "Agency Reconsidered." *Proceedings of DiGRA.* (2009).

Westecott 2011 Westecott, E. "Crafting play: little big planet." *Loading...*, 5(8). (2011).

Winge and Stalp 2013 Winge, T.M. and Stalp, M.C. "Nothing says love like a skull and crossbones tea cozy: Crafting contemporary subversive handcrafts." *Craft Research*, 4(1). (2013): 73-86.

Wright et al. 2002 Wright, T., Boria, E., & Breidenbach, P. "Creative player actions in FPS online video games: Playing Counter-Strike." *Game studies*, 2(2). (2002): 103-123.

Yee 2006 Yee, N. "Motivations for play in online games." *CyberPsychology & behavior.* 9(6) (2006): 772-775.

Zagal and Mateas 2010 Zagal, J. P., & Mateas, M. "Time in video games: A survey and analysis." *Simulation & Gaming*, 41(6), (2010): 844-868.

Zagal et al. 2007 Zagal, J. P., Mateas, M., Fernández-Vara, C., Hochhalter, B., & Lichti, N. "Towards an ontological language for game analysis." *Worlds in Play: International Perspectives on Digital Games Research.* (2007): 21.