

Well Played

**a journal on video games,
value and meaning**

volume 1 number 1 2011



<http://etc.cmu.edu/etcpress>

Minecraft, Beyond Construction and Survival

Sean C. Duncan

Armstrong Professor of Interactive Media
School of Education, Health, and Society +
Armstrong Institute for Interactive Media Studies
Miami University, Oxford, OH 45056
sean.duncan@muohio.edu

"We'll keep releasing expansions and keep the game alive, but there needs to be some kind of final version that you can point at and say, 'I did this!'... I'm not sure why I feel a need to have something to call the final version if we're just going to keep updating it, but it just feels wrong to never have reached some kind of goal. Having the game constantly be under development also seems to confuse the press." - Markus "notch" Persson in *Game Developer*, Feb. 2011

Minecraft has been one of the most unusual success stories in gaming in recent memory — within less than two years, it went from being one of many small, independent games released and discussed in an online indie game development community (in this case, the TIGSource forums; Persson, 2011b) to becoming a world-wide phenomenon that has earned its creator accolades such as the Independent Game Festival *Seumas McNally Grand Prize* award in 2011 and, by some accounts, millions of dollars in revenue (Lynley, 2011). *Minecraft* is a game that seems to have struck a chord with gamers in a relatively short period of time, yet is one that has changed significantly from release to release, as can be seen by the wry comment above by the game's primary designer, Markus "notch" Persson.

Minecraft is an alluringly moving target to try to pin down, and so in order to assess how it is “well-played” — well-designed and iteratively well-redesigned, in this particular case — we need to think more broadly about the approach Persson (and his company Mojang Specifications) have taken toward the development of the game in addition to its formal game mechanics. That is, we can’t assess only the design of the game itself, but need to take into account the shifting goals of the game’s designers, what players do with the game, and what the interactions between designer and players mean for the game’s evolution. What makes *Minecraft* “work” is a fascinating mix of the game’s aesthetic sensibility, its mechanics, its development history, and the creative activities of its players.

To get a better sense of the whole experience of *Minecraft*, let’s delve into the approach taken with its design, the way the game plays, but also the novel uses that players (in some cases, other game designers) have put *Minecraft* toward. In this paper, I will begin by outlining the game, briefly tracking its development history through the Alpha and Beta development stages¹. Then, I’ll isolate the two key player activities within the game — *construction* and *survival* — and show how the game’s success can be attributed to the interrelations and tensions between these two activities. Finally, I’ll discuss how *Minecraft*’s tensions between construction and survival have led it to be seen increasingly more as a gaming *platform*, one which is overtly afforded by the game’s design and which has led to exciting experiments in games for learning, game play as an instructional space, and games as playgrounds for the exploration of artistic goals.

Many Minecraft(s)

Minecraft is developed in Java, and runs on Macs and Windows machines alike, with versions in the works for both Android and iOS devices. On the game’s official website and sole distribution hub, <http://minecraft.net>, over 11,000,000 unique users have registered accounts, of which more than 25% (over

3,000,000) have purchased the game at the time of this paper's writing. One of the most popular and significant independent computer games of recent years, *Minecraft* has proven to be a compelling (and sometimes addictive) experience for many players.

Minecraft features several modes of play — a “*Minecraft* Classic,” offered on minecraft.net for free (the original version of the game), a single-player mode, and a mode in which players have access to multi-player, shared *Minecraft* servers. For the bulk of this paper, I will discuss the game in its single-player (not “Classic,” not multi-player) modes. The single-player game's design illustrates some of the key tensions in the game's mechanics, and, I argue, the game's successes are most clearly seen through the tension between the default survival mode and other activities within the game. Additionally, much of the game's development through the Alpha and Beta stages have focused on the single-player experience.

For either the single- or the multi-player game, the first step for the player is to create a world to inhabit. Before the game deposits the player in the game space, a three-dimensional world must be created by *Minecraft*, procedurally generated before the game is fired up for the first time (not dissimilar from one of Persson's inspirations, the complex simulation game *Dwarf Fortress*). After the intricate landscapes and biomes of one's *Minecraft* world are created before the first play, the player is deposited at a spawn point (often, it seems, at the edge of a beach). One of the first things that a player notices is the “primitive” default graphics set, presenting the world as a collection of meter-square blocks, from tree leaves to coal-infused stone to the clouds floating overhead. See Figure 1, below, for an example of the view from a starting spawn point in *Minecraft* — a pleasant morning on a sandy beach, with virtually no instruction as to what to do next.



Figure 1. A typical “opening scene” in *Minecraft*, with the player spawning on a beach in a newly-generated world.

Note that, in Figure 1, there are a number of easily-recognizable gaming interface elements at the bottom of the screen. There are hearts — usually indicative of health in first-person games and third-person adventure games (e.g., *The Legend of Zelda* series; Chess, in preparation). Below the hearts, there are a number of empty “slots” — in many first-person games, a location where one would pick and choose between a variety of weapons. The game’s visual aesthetic extends to the game’s representation of the player, with the block on the right side of the screen being an image of the player’s right hand/arm.

Beyond simply appearing “blocky,” the game’s uniformity of meter-square elements is a visual allusion to LEGO™, and suggests a space in which the player is given free rein to create whatever he or she wishes from the pieces provided. And though this is technically true (the game affords a great deal of construction), doing so is certainly not evident nor feasible during one’s first moments within the game. In these first experiences within a new *Minecraft* world, the player is simply ...on a beach, with no clear idea of what he or she can do within the world, what the goal is for the player, what dangers might be present in the game, not to mention there is nothing in terms of instruction or guidance built into the game’s interface. The game is enticingly

quiet at this stage (both in terms of auditory and visual information), perhaps leading a timid player into confusion, an uninformed player into a sense of complacency, or an adventurous player into sense that this simulation of a blocky world is truly open for player exploration.

Unsurprising for gamers in the 2010s, *Minecraft* players rarely seem to have had much difficulty with the problem of “what to do next” after firing up the game. The game’s open sandbox is as inviting to many players as it is intimidating to some, and the procedural generation of a world has drawn many players (such as myself) immediately into exploration mode, rooting around the world to explore the highest peaks and deepest caverns one can find. YouTube instructional videos, online tutorials, and collections of player knowledge in a collaborative Wiki (Minepedia; <http://minecraftwiki.net>) have also all served to guide novice players into the next steps, all of which help to form the basics of the game. Starting by literally punching trees (and other objects) with one’s bare, blocky hand, the player then stockpiles wood, stone, and other building blocks of the world. These items, then, can be recombined using the game’s crafting interface to first create a workbench — allowing a larger, more complex crafting interface — and then more complex items out of simple, basic components (see Figure 2, below).



Figure 2. The basic *Minecraft* crafting interface. The player here creates a wooden pickaxe from wood planks and sticks, using a crafting bench.

Thus, we see that the creativity (the “LEGO™ set” analogy) is not just something afforded by the game’s elements, but is something integral for a player to proceed in creating *anything* within the game. And, more importantly, as one quickly discovers when the game’s square (see Figure 1) sun completes its arc across the sky, the skillful recombination of items is a necessary part of the basic game. For *Minecraft* is not simply an architectural simulator, but a game in which the player must make *protective* structures against the number of monsters that arrive out of the night — spiders, skeletons, zombies, and the famous “Creeper” (see Figure 3 below) that has been emblematic of the game in many ways. Wood, sand, coal, stone, diamond, and so on each serve a purpose, and as the player progresses, he or she learns to create a stronger pickaxe, to lay miles of tracks and minecarts to more efficiently move ore around the world, to make torches to illuminate dark crevasses, and to recombine building

blocks (stone, iron, glass) into more complex pieces that may help build a fortress to protect against the monsters of the night.



Figure 3. The Creeper, *Minecraft's* most feared and most damaging monster.

The world of *Minecraft* is thus simultaneously a recombinatory, private virtual world for creative purposes and also for survival purposes — nothing in the game tells you that you need to create large, elaborate structures, but the game does quickly encourage you to make *something*. To avoid dying (and losing all of one's on-body possessions, starting over at the spawn point), the player needs to create structures and armor to survive the nightly onslaught. Admittedly, this is a relatively simple challenge; one could interpret the goal of the game as being simply “don't die” and to make a simple building to hide in for the length of the night (seven minutes in real time). But, as part of *Minecraft's* brilliance is in the balance of these creative and survival elements, players rarely simply “wait out” the night, taking the opportunity to dig, uncover new materials, and craft increasingly complex objects.

So, then, it seems that the tension between construction and survival may help us to understand the unique appeal of this

game. In thinking about how *Minecraft* is designed, one needs to first distinguish between several stages of *Minecraft*'s development — stages in which the survival elements were first not included and then later built into the game — and then tease out the significant differences between the game's designed mechanics and players' experiences. With its continuous updating and revision, there have been many *Minecrafts*, more than the simple "Alpha" and "Beta" labels indicate, and we need to understand how the game has evolved to accommodate both creative construction activities and the survival elements that typify its default settings.

Construction vs. Survival

"Waterfall is dead, long live agile!" (Persson, 2011b).

Though the balance between construction and survival is one that characterizes the default single-player (and many multi-player) versions of the game, it was not always the case. In the earliest versions of the game, now labeled "*Minecraft* Classic" and playable for free via the game's official website, *Minecraft* emphasized creation without the survival elements of the game. The earliest versions of the game were understandably its most rough, but were also released to the general public at a very early stage of development.

Persson rejects a "waterfall" model (e.g., Royce, 1970), in which relatively-rigid stages of software development follow one after another, without the flexibility to create an appropriate solution to a changing problem or changing needs of the software's users. Persson proudly exclaims "long live agile!", indicating his preference, instead, for *agile software development*, a model in which *customer collaboration* is an explicit element (Agile Alliance, 2001). And, in this regard, "agile" is an understatement in describing the development of *Minecraft*, with its quick succession of updates, the use of players as live testers of the game, and the open conversations about the game's design that Persson himself has had with players. *Minecraft* is a game in

which the players are not simply consumers, but are active in the development of the game as it has changed.

With over 9800 tweets, and over 290,000 followers on Twitter at the time of this paper's writing, Persson (or "@notch" on Twitter) has amassed a relatively large following for an independent game designer on just this one form of social media. Using Twitter (and his Tumblr, "The World of Notch"; <http://notch.tumblr.com>) to disseminate ideas about future features in the game (e.g., turbines, "adventure mode," etc.), Persson has attempted to involve the players of the game in its development. This has, of course, not come without difficulties — widely reported in October, 2010 was the distributed denial of service attack that brought down *Minecraft*'s multiplayer functionality. Upset because of a perceived lack of updates by Persson and Mojang, users on 4chan demanded that Persson start "providing ... customers with the updates that [Persson promised] them," taking down multi-player functionality (Crecente, 2010). Some angry players went well beyond most reasonable definitions of "customer collaboration", with a powerful and vocal minority of players expressing their concerns about the game's development through aggressive means. Though Persson has sometimes been unable to meet the demands of *Minecraft*'s player base, his disposition has been "agile" from the start, releasing the first game to the public only a week after he had begun development on it.

As a result, it should not be much of a surprise that many of the game's most distinctive elements were worked through with players quite early in the process. Quickly after developing a "sandbox," construction-based game (what's now *Minecraft* Classic) without a clear conflict present for the player to overcome, it was clear that more needed to be added. As a domain for creativity, *Minecraft* Classic was evocative and interesting, but it was lacking in impetus — there needed to be something more to drive the player's actions. As Persson himself stated on *Minecraft*'s "About the game" page:

“I strongly believe that all good stories have a conflict, and that all good games tell a good story regardless of if it's pre-written or emergent. Free building mode is fine and dandy, but for many people it will ultimately become boring once you've got it figured out. It's like playing a first person shooter in god mode, or giving yourself infinite funds in a strategy game.. a lack of challenge kills the fun.” (Persson, 2011b).

Acknowledging that there needed to be some kind of conflict within the game, Persson worked to add something to spur on the player beyond just the construction of objects within a virtual space. The early, key development of “survival mode” provided players with the “challenge” that Persson was looking for, while also giving the players’ construction activities increased consequence in terms of the goal structures of the game. Adding the survival mode turned the game from a simulation of a virtual space into a *game* with a set of short-term, designer-imposed goals, albeit some that do not remain compelling for most players after a degree of experience with the game.

To be clear, the survival mode of *Minecraft* is just a default setting and many advanced players disable it (playing on a “Peaceful” setting) in order to focus on construction alone. But, by including constraints that propelled players to mine, recombine elements of the game, and construct in order to avoid consequences, *Minecraft’s* “sandbox” gained a compelling structure. It’s a minimal structure, yes, and one that does not impose any specific form of construction on the player, but it serves as an impetus to explore more parts of the world, to dig deeper into the world’s underbelly, and to make increasingly complex objects.

In a brief *Gamasutra* exploration of the game, Margaret Robertson identified one of the key ways that *Minecraft*

successfully ties construction to survival, and vice versa. She stated:

[*Minecraft* enforces] play imperatives which take you through the first few hours of play. It means that when the sandbox possibilities do start to open up — of building and exploring (I'm told it would take six years of real time to walk around a full *Minecraft* world) you are deeply embedded into the world. You have a skill-set, a sense of ownership and belonging, which fuel you through the challenge of free, creative play. And that's crucial, because free, creative play is actually quite a grueling prospect, full of the pain and effort of making and losing. (Robertston, 2010, pg. 3).

That is, the game uses survival mode as a way to push the player through the earliest stages of the game, and to build a sense of immersion within the world. As the sun begins to set on the first night in a matter of minutes, the player is often scrambling to build a pickaxe, find coal (necessary to make torches), and either build a simple house or carve a sanctuary out of a rock face. Unlike *Minecraft* Classic, the survival mode *pushes* the player to explore the space, learn to build, and then actually construct within the first few minutes of the game.

Robertson's point is well-taken in that the common misconception that *Minecraft* is purely about construction invites inaccurate comparisons to LEGO™ and ignores survival mode's most useful role in helping to guide the player's experience in the earliest stages. In a game without overt tutorials or much in terms of in-game information at all (again, see Figure 1), it falls upon the design of the game's challenges to guide players into a deep immersion within the world. Robertson effectively described *Minecraft* as a game in which "everything in the world was already made of LEGO™ and bits of it wanted you dead" (Robertson, 2010, pg. 1), with this fear of death — or, to be more accurate, fear of losing one's objects and respawning — helping

to give added practical significance to one's construction activities.

The use of the survival mode to accomplish this should not be understated — while there are alternative ways to drive immersion in the game, many of the obvious choices employed by other (often commercial) games would simply break *Minecraft*. For instance, though Persson and Mojang are currently designing an “adventure mode” for the game, I argue that immersion cannot be easily accomplished by simply inserting some kind of overarching narrative into the space — the world is, after all, procedurally generated and thus unique for each player, and much of the joy of the game derives from explorations of a truly unknown, unmapped space. Plus, the inclusion of in-game tutorials or instruction would be jarringly intrusive in a game so carefully designed to create a naturalistic experience (c.f., C418's minimalist and evocative soundtrack). In *Minecraft*, the key relationship to note is that survival is necessary to propel initial construction, but that construction also “pays off” by supporting survival.

Robertson successfully identified that the co-existence of construction and survival is what makes the game “work,” but it's still a bit more than that, I suggest — one could easily argue that *neither* construction nor survival are particularly interesting game mechanics on their own. The construction-only of *Minecraft* Classic needed a survival element to drive it, but the survival activities of *Minecraft* (essentially, “hide whenever the sun is down”) would make a dull game in and of itself. Construction and survival are interrelated, but also competing; player immersion in the game seems to balance the two, and much of the fun seems to involve avoiding the problems of focusing on one over the other.

Here, we might consider Csíkszentmihályi's (1975) widely cited notion of “flow.” For Csíkszentmihályi, flow can be describe as the positive psychological notion of experiencing a heightened, optimal state during an activity, with flow states being balanced

between two competing states of “boredom” and “anxiety,” resting in neither. Similarly, perhaps *Minecraft*’s construction and survival modes map more-or-less onto these notions; the balance of survival mode keeps the game from drifting into boredom territory, while the creative construction of the game helps to keep anxiety from being the overriding experience of the game. The individual components of survival and construction need one another to drive players deeper into the game and to achieve a joyous, “flow”-like state of play.

Furthermore, we might be able to characterize the design path that *Minecraft* has taken as reflecting a bouncing between the boredom of construction and the anxiety of survival. Clearly the introduction of survival mode in the first place added consequence to the game that propelled play, but, if, according to Robertson, this is primarily useful in order to build immersion in the game, what is to keep players from being bored with construction later in the game? Perhaps this is what is driving Persson and Mojang Specifications’ interest in adding an “adventure” system to the game, and may also account for the allure of playing on open multi-player servers (often plagued by “griefers” who may seek to destroy one’s creative work).

Thus, in *Minecraft*, it seems that a form of *emergent gameplay* evolves out of the interaction of two, relatively simple and less compelling game mechanics and it is this that makes the game work so well, taking the edge off of the “grueling,” time-consuming creative work that is part of complex, advanced play. And this brings us around to the topic of those intricate constructions, as well as how we might better understand the ways that the emergent gameplay of *Minecraft* has led players to think of it less like a “game,” and more like a platform for their creative works. Focusing on two major themes — *Minecraft* as *instructional platform* and *Minecraft* as *experiential platform* — I want to show how players move “beyond *Minecraft*,” taking the significance of the game beyond what Persson and Mojang Specifications may have ever originally intended.

Beyond Minecraft

Many have identified that *Minecraft* provides opportunities for creative construction well beyond the need for the survival mode of the game. And, as Robertson implies, after a point there is really very little need for the survival mode: it serves to help embed the player in the world, but later might impede the construction goals of the player. Banks & Potts (2010) outlined many of the ways that communities have formed around the game to build instructional websites, share creative constructions within the game, and otherwise employ social learning to further players' understanding of the game. The variety of many of the well-publicized creative constructions implemented in *Minecraft* is certainly impressive, ranging from full-scale models of the USS Enterprise-D to a working arithmetic logic unit implemented architecturally. And, clearly, the forms of social learning fostered by the game are a testament to how effectively one can use the building blocks of *Minecraft* to construct truly elaborate spaces within *Minecraft*.

But, while Banks and Potts (2010) focused on the co-constructive elements of play, there was little focus on the co-constructed elements of the game's design, nor the role that the survival mode played. If the heightened state of complex construction in *Minecraft* emerges out of the interaction of construction and survival, I argue that a more complete understanding of the game necessitates thinking about both but also *beyond* them. That is, thinking about how players may capitalize on these activities to build experiences with *Minecraft* that are not necessarily a core part of the game. Mojang is not above continuing to iterate means toward creating player engagement within the game, and players have taken it upon themselves to use the game for similar aims.

With user-generated content such a key part of *Minecraft*'s success, it's perhaps unsurprising that some have begun using *Minecraft* as a platform for the development of other games, virtual spaces, and experiential experiments. Nothing from Persson or Mojang Specifications would seem to indicate that

this is outside of their view of appropriate uses of the game and, quite to the contrary, they seem to be receptive to new uses for their game. I'll present two types of uses here, one geared very specifically toward educational uses, and then a "super-set" case, in which larger artistic and experiential goals seem to be at play.

First, we should consider *Minecraft* as an *instructional platform*. In 2010, Joel Levin made a splash with *The Minecraft Teacher* (<http://minecraftteacher.net/>), a blog detailing his experiments using *Minecraft* as an educational environment for first- and second-graders (Levin, 2011). Levin's experiments have caught the attention of Mojang Specifications, and are one of the most prominent ways that *Minecraft* has moved from being simply a game for entertainment and has been adapted into other contexts. As the game becomes more and more entrenched in gaming culture and more educators have become exposed to it, others have found themselves drawn to using it within educational environments. For instance, *Massively Minecraft* has recently arisen as a community for teachers to explore ways that *Minecraft* can be used across the curriculum (Kay, Groom, and Stuckey, 2011).

Across a number of levels of instruction, there is the potential that *Minecraft* can be useful for as a platform for designing new learning environments that utilize and sit atop the commercial game. Using only the construction and survival elements built into the game, players can craft instructional environments using the elements presented within it and begin to develop virtual environments that rely on many but not all of Persson and Mojang's designed elements. Primarily focusing on construction, Levin has described modifying the game to make children players impervious to damage (Webster, 2011) — removing, in essence, exactly what Robertson saw as being one of the most compelling elements of the game, and what drove immersion. Reducing the game back to a construction set, Levin found that *Minecraft* might be utilized to foster creativity in young children.

Others have attempted to use the game to teach very specific content areas that evolve from the core elements of the game. For example, in 2010, three undergraduates at Miami University (Michael Kolich, Alexi Chow, and Tim Mason), created a game prototype entitled *Circuit Madness*, a game to teach logic, implemented within *Minecraft*. *Circuit Madness* was geared specifically toward teaching players how to differentiate logical operators, using the embodied experience of moving around in a *Minecraft* world to convey the learning content. The students implemented the game entirely within a single-player world of *Minecraft*, building all devices and environments in the game using standard items (from wooden signs to levers to redstone circuitry). Critical for this group of students was using *Minecraft* as a *design* environment, and layering a novel experience atop *Minecraft*'s familiar features, even if it ignored the game's survival mode.

Though just a simple prototype, the use of space in *Circuit Madness* was reminiscent of games such as Valve's *Portal* and *Portal 2*, in which adjoining rooms were presented as puzzles that players needed to solve before progressing to the next in the structure (see Figure 4, below). Each room in *Circuit Madness* used built-in *Minecraft* elements to craft a space in which players needed to first learn simple logical operators (AND, OR, XOR, and so on) and then enact the logical operators in the correct sequence to proceed on to the next room in the game. Similar in content to Warren Robinett's classic *Rocky's Boots*, this game prototype indicates that beyond simply teaching the content of logical operators, *Circuit Madness* utilized familiar elements of *Minecraft* to do unfamiliar things. That is, the experiment of this prototype helps us to think about *Minecraft* somewhat differently — as a space in which the player's goals are imposed by what's already built in the space before the player gets to it.



Figure 4. A screenshot of *Circuit Madness*, a game implemented within *Minecraft* to teach players basic logical operators.

But, for *Circuit Madness*, we again fall back on the space being used as, more or less, a construction set to develop a new experience. Nowhere do the survival mode elements of the game have consequence, and, as an instructional environment, there are impediments that can hamper such a game’s usability — for example, left-clicking on a switch will punch and destroy it, as is the norm for all *Minecraft* objects, meaning players are forced to right-click to progress through the game. Though the potential exists for both Levin’s experiments and games such as *Circuit Madness* to create transformative learning experiences, they utilize *Minecraft* as a jumping-off point. Are there other experiences that can more fundamentally capture the survival/construction dynamic of *Minecraft* and still aim to do something different than the original game?

There have been game designers who have attempted to craft other experiences atop this designed game, using it for what I’m labeling here as an *experiential platform*. This is, admittedly, an awkward term, especially one to contrast with “instructional” (as, certainly, instruction is an experience). But I consider “instructional” to be a specific subset of “experiential” in this paper, intending to characterize the ways that *Minecraft* works to

provide players with experiences that are somehow “about” something other than game’s presumed original intent. Instructional contexts crafted within *Minecraft* fit the bill, but, so far at least, seem to involve modifying the game itself (to better accommodate younger age players) or layering in a new game that has little to do with the original.

An excellent example of a success in going beyond basic *Minecraft* is Jason Rohrer’s well-publicized “Chain World” experiment, originally developed as part of Eric Zimmerman’s Game Developers’ Challenge at the 2011 Game Developers Conference (for an excellent write-up on the history of this experiment and its subsequent controversies, please see Fagone, 2011). Tasked with creating a game that itself was a religion, Rohrer mused that to simulate a religion, one could simulate the history of a group of people, experiencing in sequence a common set of cultural artifacts, interpreting them, and reinterpreting them as time went on, until the long-ago past began to take on the hue of myth. The past would become something wondered about and (potentially) revered, with the veil of history clouding the intent of long-ago inhabitants of the space.

Like a spatial, ludic game of “telephone,” *Chain World* is ultimately quite simple — it features one single-player *Minecraft* world, set on a non-“Peaceful” difficulty, initiated by Rohrer and stored on a flash drive. The player is tasked with doing whatever they’d like in the world: building monuments, exploring caverns, leaving their mark on the world in some fashion. Once the player dies (and for Rohrer, playing with his son, it was apparently an excruciatingly unexpected virtual death), the game must be stopped and then the flash drive passed on to the next player. Each successive player is left wondering who created what in the previous world, and what their intentions might have been.

Like with his earlier *memento mori* game, *Passage*, Rohrer seems to have been focused on in-game death with *Chain World*, elevating *Minecraft*’s death/respawning to a greater significance

than it has in the basic game, where it is a temporary impediment and loss of objects. Death and, by extension, survival can be more than just elements that can spur the player toward deeper immersion in the game — in *Chain World*, death is the end of a single player's experience, a state that is to be avoided at all costs, and one that you cannot speak with others about after it's happened. Not so different from the real world, it seems, and a key difference between *Chain World* and the other attempts to move “beyond” *Minecraft* described in this paper. For Rohrer has attempted to create an experience that speaks to a lofty goal (to make a “game that is itself a religion”), while also reinterpreting the essential elements of *Minecraft* in construction and survival.

In *Chain World*, construction and survival are just as important as they are for any single-player, non-“Peaceful” game of *Minecraft* — it's just that they *mean* something completely different. Survival is not just something one does to avoid losing objects and having to walk back from the respawn point, it's the literal end of one's gaming experience in this particular world. Construction is not just a tool to help one stay alive during the monster-filled nights, it's also a means of building on the work of previous “generations,” and the legacy that the player leaves for future players. Mechanically, *Chain World* is absolutely identical to any single-player game of *Minecraft*, but the social restructuring of the experience gives rise to new levels of meaning in the gameplay.

So, perhaps we can view use of *Minecraft* as an experiential platform that may be at its most powerful when the game's core relationship between survival and construction is kept central. The kinds of uses that, in particular, *Circuit Madness* and *Chain World* seem to exemplify are akin to what James Paul Gee and Elisabeth Hayes alternately call “soft modding” or “socio-technical modding” (Gee and Hayes, 2010): the social modification of a gaming experience without actually modifying any of the game's code. *Chain World* in particular is an experience that sits atop another game, requiring no additional coding or hard

modifications to implement (though Rohrer did implement code to update the *Minecraft* installation on the flash drive). As such, retains much of the spirit of the original game, while shifting the meaning of the experience to something new. Ultimately, beyond the “anxiety” of survival mode and the “boredom” of consequence-free construction, there may continue to be a realm of possibility for *Minecraft* “soft mods,” one that is afforded by few other games in quite the same way.

Final Thoughts

All in all, *Minecraft* continues to work due to the interaction of these many factors — construction and survival, certainly, but also social interaction and iterative design. While Markus Persson and Mojang Specifications rely upon an agile model of development to iterate and continually shape the game, we can already see that its evolution from a solely construction-based game to one in which a survival mode spurred on immersion in the game was critical for its success. And, as others have made clear — ranging from elementary school children to college students to game developers such as Jason Rohrer — there is an appealing flexibility in *Minecraft* to serve as more than just a play space, but also be a platform for new, meaningful experiences.

As *Minecraft* further develops and takes hold in other systems (such as iOS and Android devices), it will be interesting to see how Persson and Mojang Specifications continue to change the game, especially taking into account how players have co-constructed the game with them. I argue that the tension between survival and construction is key to the game's success, and even to the success of the experiences that use *Minecraft* for other aims (such as *Chain World*, that successfully reconceives the core activities of survival and construction). If the game's long-term value is ultimately tied to how these dynamics play out not just with Mojang's intentions but also through the goals of *Minecraft*'s players, perhaps keys to the game's significance will end up lying somewhere within the experiments that players build atop the game as much as with the features that give rise to these experiments.

References

Agile Alliance (2001). *Manifesto for agile software development*. Retrieved online via <http://agilealliance.org>.

Banks, J. & Potts, J. (2010). Towards a cultural science of videogames: evolutionary social learning. *Cultural Science* 3 (1), 1-17.

Chess, S. (in preparation). Going with the Flo: Diner Dash and feminism. To appear in *Feminist Media Studies*.

Crecente, B. (2010). Minecraft apparently under siege by angry addicts demanding more updates. *Kotaku*, retrieved online via <http://kotaku.com/5669766/minecraft-under-siege-by-angry-addicts-demanding-more-updates>

Csikszentmihályi, M. (1975). *Beyond boredom and anxiety: experiencing flow in work and play*. San Francisco: Jossey-Bass.

Fagone, J. (2011). *Chain World* videogame was supposed to be a religion — not a holy war. *Wired*, retrieved online via http://www.wired.com/magazine/2011/07/mf_chainworld/all/1

Levin, J. (2011). *The Minecraft Teacher*. Retrieved online via <http://minecraftteacher.net/>

Lynley, M. (2011). Indie sandbox game *Minecraft* racks up 1 million sales sans a publisher. *The New York Times*, retrieved online via <http://www.nytimes.com/external/venturebeat/2011/01/12/12venturebeat-indie-sandbox-game-minecraft-racks-up-1-million-27319.html>

Persson, M. (2011a). Postmortem: Mojang's *Minecraft*. *Game Developer*, 18(2), 24-30.

Persson, M. (2011b). About the game. Retrieved online via <http://www.minecraft.net/about.jsp>.

Robertston, M. (2010). Five minutes of... *Minecraft*. *Gamasutra*, retrieved online via http://www.gamasutra.com/view/feature/6179/five_minutes_of_minecraft.php

Royce, W. (1970), Managing the development of large software systems. *Proceedings of IEEE WESCON 26*, 1–9.

Webster, A. (2011). Educational building blocks: how Minecraft is used in classrooms. *Ars Technica*, retrieved online via <http://arstechnica.com/gaming/news/2011/04/educational-building-blocks-how-minecraft-is-being-used-in-the-classroom.ars>