Luke Caspar Pearson

Design by Decoding: Exposing Environments Mediated by "Cultural Software"

JAE: Scholarship of Design Submission

Abstract

This paper considers the manipulation of physical environments by what Manovich terms "cultural software," computer programs we use to generate culture. Utilizing case studies from teaching workshops, I examine methods for interrogating the logic underpinning the software of global imaging, virtual construction and videogames to understand their effect upon physical environments. These are proposed as ways for architects to engage with the layered condition of what "environment" constitutes today. I conclude that the logics we carry from design software into reality must be scrutinized so that we do not fail to see the ideology our digital tools transported into physical conditions.

Paper

We can say with some certainty that any architectural student undertaking their education today will at some point produce work using software. Whether they pursue computational design, hand-drawing or live-site work, the chances are some form of software will be involved in the production of the project. This might be an Autocad drawing for fabrication, a Photoshop visualization, or the production of a portfolio in InDesign or the dimensioning of a site from Google Earth. Each of these software forms imposes systems upon the architect-user, not only in terms of models of working, but also in the type of design environment they establish while using them.

These applications are what media theorist Lev Manovich has termed *cultural software*—"certain types of software that support actions we normally associate with 'culture.'"[1] This software does not necessarily require knowledge of code, and typically utilizes a Graphic User Interface (GUI). Yet as Manovich points out: "Given that today the multi-billion dollar culture industry is enabled by media applications, it is interesting that there is no single accepted way to classify them."[2] Manovich's attempts to categorize these programs and frame their impact might leave an impression on how we understand the relationship between architectural design and software. Given the effect that software programs have had on the way we are able to visualize and conceptualize the world around us today, it would follow that interrogating our day-to-day operation of software as designers is key to understanding our new relationships to environments—physical or virtual. As Manovich says, this concerns our use of software on many levels:

> "Therefore, if we want to understand how software has already re-shaped media both conceptually and practically, we have to take a close look at the everyday tools used by the great majority of both professional and non-professional users—i.e. application software, web based software, and, of course, mobile apps."[3]

As Manovich points out, cultural software is of interest because it is so ubiquitous, and also because it establishes multiple environments for us: not least new working environments

and stand-in versions of our real environments. This is a kind of technological soft power, where the way environments are represented and approached in software affects the manner in which they are represented and approached in reality. Of course, the role of technology in shifting cultural and physical environments has long been studied by architects. When the Smithsons said "but today we collect ads,"[4] they expressly positioned themselves in the lineage of Gropius' interest in grain silos and Le Corbusier's fascination with aviation. If the impact of grain silos could be seen written onto the landscape through mechanized agriculture and airplanes collapsed distances across the globe allowing for greater movements of people, then the impact of advertisements was seen in the cultural environment in which designers operated. Advertisements were different from industrial technology, their status was more nebulous as representations designed to create impulses in popular culture. The Smithsons declared that architects should learn from the "pace-setting" of the advertising industry and its engagement with ordinary people. In other words, if advertising was creating a cultural environment for the easy transmission and reception of desire, then architects and designers needed to operate within that situation as well. Today social media and video games provide object lessons in the pace-setting properties of communication and virtual environments generated by cultural software.

Reyner Banham was similarly detailing the effects of mass produced pop cultural *gizmos* and media on our environments. Banham's *great gizmo* was a mass produced object "whose function is to transform some undifferentiated set of circumstances to a condition nearer human desires."[5] While Banham's gizmo was described as a physical object that colonizes a landscape, today we understand such devices as combinations of hardware and software. With cloud-based technologies, hardware is taking an increasingly subservient role. A gizmo may still be "self-contained" physically, yet it is often ultra-connected. If we are still able to bring environments nearer to human desires via gizmos, then it is through a combination of hardware and cultural software. To navigate a city using *Google Earth* on our phones is to hold the physical gizmo in our hands, while operating the cultural software of the app (itself produced using cultural software), each of which are constantly feeding their position and situation within the environment to remote servers.

Contemporary writers and artists (themselves using cultural software as communication channels) such as Keller Easterling, Geoff Manaugh, Douglas Coupland and James Bridle give us some sense of the current situation. Bridle's blog-cum-art-movement *The New Aesthetic*[6] is a repository for the collected debris of our digital realms, the products of new environments that straddle the digital and physical, with often unintended and grotesque consequences. Likewise, Easterling has reflected on environments regulated by technological and legal forces other than the architect, from the jurisdictions of Free Economic Zones to the cultivated environments of Arnold Palmer golf resorts. From Banham to Bridle, from the gizmo to software, technology has fundamentally changed both our environment proper, and the intellectual and cultural environments within which we operate as designers. None of these projects are necessarily about defining the "epochal" architectural style of our time, but more concerned with elucidating links between architecture and the everyday technologies that affect most people.

The impact of everyday technologies is evidently huge. The virtual city of *Los Santos* in *Grand Theft Auto V*, a deviated version of Los Angeles, can have over 54 million visitors in two years,[7] calling into question what constitutes *place*. Cultural software also sidesteps the

traditional media hierarchies of top-down communications. For instance, *Reddit* is both a platform for President Obama to communicate on, while also a hive of questionable internet culture. The rise to power of Donald Trump, with his reliance on social media platforms, has further advanced non-traditional ways of disseminating news. Twitter is now a direct source of seemingly unfiltered information straight from the diminutive fingers of the president. This means that as architects and educators we need to engage with the intensity and diversity of environments held on blogs, feeds and forums, in free paint programs or in videogame worlds. The Smithsons or Banham would surely have embraced the structural possibilities they offer for engagement with the world around us, even if we might recoil from some of the content they proffer.

By fostering curiosity about how such cultural software has altered traditional methods and media of architectural design, we can encourage scholarship that embraces all the contradictions and conflicts of our technological world. Drawing from the work of media theorists such as Manovich and Alexander Galloway, and game theorist Ian Bogost, this essay outlines ways in which we might encourage the curiosity to question and subvert the tools that mediate our modern environments. To do this I will introduce a series of case studies drawn from my own research, alongside research projects conducted by students under my supervision, that I feel are beginning to "decode" the digital tools of the architect. None of these projects decode through coding itself because, in the world of Manovich's cultural software, the vast majority of users do not encounter or engage with the deep coded structure. Indeed, the success of cultural software typically depends on being user friendly. Although twisting the coded structure is clearly a fertile site for research, each of the projects examined in this paper engages digital environments as the typical person would encounter them, generally effecting change through the tools provided to the user rather than inventing or inserting new ones.

Each of these studies seeks to unpack the rules of a particular piece of cultural software in order to understand the environment it establishes for the user, and compare its relationship to our physical world, proposing new ways in which architectural design might respond in turn to the particular logics and hierarchies implicit in these situations. These projects were all undertaken as separate investigations under a series of studios run by myself, and so they represent divergent approaches toward understanding these digital tools rather than one conclusive methodology.

Worlds of Remediated Representation

In *Software Takes Command*, Lev Manovich outlines the history of cultural software. He uses a set of pioneering computer engineers as reference points, in particular the Sketchpad work of Ivan Sutherland, to create "a communication system between two entities: a human and an intelligent machine."[8] He demonstrates software that first "remediates" existing media (for instance, pen on paper) and then extends its capability (for instance repositioning vertex points along a line in CAD).

CAD remediates and extends the line drawing. Photoshop remediates and extends painterly and drawn effects, as well as photomontage and collage. 3D modelling software encompasses both planes and (in software such as ZBrush) a sculptural process of hewing

form from a material, extending into rendering and fabrication. BIM remediates the organization and interpretation of drawn materials and adds the functionality to quickly export further media from a master model. GIS systems collapse numerous datasets into maps that may previously have been stored and recorded on completely different media.

Manovich makes clear that we cannot entirely divorce our work in cultural software from history and the environments in which design has operated for many years, but that they represent something new:

> "Computational media uses these traditional human media simply as building blocks to create previously unimaginable representational and information structures, creative and thinking tools, and communication options."[9]

Every time a student or practitioner constructs a collaged view in Photoshop, or produces a site map using GIS data, they are utilizing programs that take many of their principles and symbolism from non-digital media that came before. But this does not mean they have not changed them irrevocably—becoming tools of "permanent extendibility."[10] For instance, Manovich questions whether Photoshop may have turned the photomontage from a political device into one that smooths over difference,[11] the tool used by architects and students the world over to place their buildings into a remote context as captured via Google Earth, or to obfuscate undeveloped parts of the scheme that did not meet the deadline.

To see the seductive images of an end-of-year show or in an international competition is often to see Photoshop wielded in such a way: a texture applied here, a contrast adjustment there, a generic tree or a person smoothed into the context and equalized. As James Bridle has discussed in his *render ghosts* project,[12] this has the effect of flattening the cultural and social differences of disparate parts of the globe. If we continue to proliferate generic and non-contextual people (usually of European or North American ethnicities) alongside universal trees, street furniture and materials, the danger is that the production of architectural imagery through cultural software has the effect of flattening the broad spectrum of environments in which architecture may be placed. Perhaps it is already here: Charles Jencks' definition of "generic individualism"[13] in places such as Singapore emphasizes the standardizing effect of global capitalism on the built environment. It is also true that buildings bear the mark of their maker, both in their design and in the cultural software used to create them. If capital seeks out the iconic building and shapes environments around it, then the sweeps of a Zaha building—drawn from their employees' skill in Maya, Grasshopper and TSplines—shows the direct impact cultural software has on the construction of our cities. When Lebbeus Woods noted Zaha's Guangzhou Opera house was indistinguishable from its renderings, we see this relationship come full circle.[14]

But can we use these remediating and extending tools in different ways, to reintroduce an idea of the rupture as an environmental tactic? Being critical about software is doubly important when we use the same techniques in study or research as in commercial work.

Allegorithmic Environments

Manovich has previously described virtual "navigable space" as a form of *new media*, although he holds off committing to whether he sees this as a remediation of physical

architecture.[15] With the rise of easily obtainable *game engine* software, the ability to produce virtual spaces has become simpler, with engines able to import 3D models from standard architectural modeling programs. Game engines *do* remediate and extend architectural space; they do this by imposing their own environments through logic, physics and rendering systems. So if one makes a building model and uses a game engine to compose a *walkthrough*, it will appear, and be experienced differently, on diverse game engines such as Unity 3D, Unreal or Cryengine. The prevalence and popularity of video games today means that there are many potential clients, designers and students who are used to the spatial agency offered by virtual environments and how their peculiar logics divert from reality. But despite these deviations, in simulating reality they still say something about the physical environments that they portray. When architecture student Vincent Oscala managed to create a self-sustaining metropolis in SimCity 2000, and thus "completed" the game, his city turned out to be a totalitarian hellscape run under a perpetual state of "martial law."[16]

Through the use of game engines we can also add rules and protocols to interactions in our virtual building that allow us to design the ways the user can engage with it. On the simplest level, we might have to press certain buttons or perform actions on the controller to activate architectural spaces, or even to simply make our virtual camera see anything. Then we might have to achieve certain objectives defined by the developer in order to open new spaces to us. These rules of engagement can make arguments that Ian Bogost terms "procedural rhetoric."[17] The virtual navigable environment allows us to extend architecture through the application of rules and representations *directly into it.* McKenzie Wark and Alexander Galloway also find game spaces to be allegorical. They see the tension between a represented allegory and the logical rules of a virtual space as something they term an "allegorithm":

> "What is distinctive about games is that they produce for the gamer an intuitive relation to the algorithm. The intuitive experience and the organizing algorithm together are an allegorithm for a future that in gamespace is forever promised but never comes to pass."[18] (N.B. In the context of this quote Wark is referring to the real world as "gamespace.")

Clearly simulated space is a representation. But the presence of rules in establishing that environment as an interactive one produces tension. These environments support inhabitation, but only of a sort. And the sort of inhabitation they allow is at the sole discretion of the author. If millions of players experience Los Angeles through the rubric of Rockstar Game's authorship, or learn about historical cities via *Assassin's Creed*, then the logics of inhabitation they allow—and the messages this imparts—are significant.

My own research has used architectural drawing as a tool to regard the artificiality of video game environments. This produced a series of drawings entitled *Noclip World*, after a famous cheat code that lets the players float through walls, escaping the confines of the virtual environment. I utilize cheat codes to rupture the fabric of the game and then transcribe the results through screenshots to produce drawings that reframe glitches as architectural conditions in their own right (Figure 1). Through this process of detaching myself from the game world, I am able to identify the ruptures in its carefully cultivated environment. As a result my drawings tackle the blurring of edges, the failure of lighting systems, and the revealing of the virtual space's proscenium (Figure 2). Critical drawing, long a tool of the architect, now becomes a tool to unpack the increasingly realistic worlds that video games offer us. This is comparable to what Daniel Reynolds has called "virtual world naturalism"—

which he terms as a wandering exploration through game environments in order to reframe them from their own constituent parts.[19]

In *Works of Game*, John Sharp also discusses artistic practices that remove virtual environments from their normative frames of display. He argues that this "distancing" allows us to regard the absurdities of the digital worlds which we regularly enter into.[20] For instance, Harun Farocki's *Parallel* series of films explores the nuances of video game environments through "documentary" footage and voiceover, calmly discussing the placement of clouds or the propagation of virtual wind that the player will never actually feel. Kim Laughton's *los_santos.obj* involves the removal of textures from *Grand Theft Auto V* to turn it into an environment of geometry and fog, alerting us to the fact that this realistic world can only exist through both 3D modeling and the applied materiality of mapped textures.[21]

The future of our design environments may not be in game engines for photorealistic walkthroughs, but in pushing the aesthetic possibilities of video game spaces for speculation. In combining the procedural and representational, we may be able to encode the political and cultural forces of real environments into their virtual equivalents, and use this to thoroughly test strategies before enacting them on our physical surroundings. But if this might be the preserve of new speculative projects to emerge from the relationship between our environment and the media by which we design it, what does cultural software *currently* do to our physical world?

Glitching Territories

Our conception of the site as a context in which architecture is placed is now under the pull of technologies for the "remote viewing" of space. Student projects and competition submissions sited across the world utilize *facsimile* versions of the built environment. Google Earth Pro provides powerful tools for capturing high-resolution imagery of the world, and terrain can then be plugged into programs such as Sketchup. More intrepid and legally flighty users might even utilize a second order of software such as *photogrammetry* programs that create 3D models from photographs, using screenshots to obtain particularly high-resolution meshes of Google's terrain data. Of course we can also make our own maps in the software, adding points of interest that facilitate our own particular reading of an environment.

These interfaces are undoubtedly liberating in the possibilities they offer to comprehend remote locations, but they are by no means devoid of politics or inconsistency. As Mark Dorrian argues, if the famous "Blue Marble" photograph of the earth taken by the Apollo 17 astronauts framed the planet as a "single organism," then the "suturing" of multiple images together into the Google Earth globe[22]—as he puts it—demonstrates the planet under the logic of surveillance imagery and algorithmic processing. The Blue Marble photograph framed the earth as a singular environment, but the Google globe is a *Frankensteinian* mix of imagery often occluded by legal and environmental restrictions. This can have the effect of reinforcing boundaries and inconsistencies when we encounter elements that appear to be "glitches" in continuity.

Errors in such software are nothing new. The 2012 release of Apple Maps was accompanied by copious screenshots detailing the glitches that saw freeways melt into hills, monuments become flattened and cities moved hundreds of miles. Sites such as

http://theamazingios6maps.tumblr.com/ popped up to document this weird new urban realm of glitch and half-truth. As with much cultural software activity, the identification and communication of these errors was initiated by internet communities and social media users. Their exposure of the everyday glitch opened up new lines of architectural inquiry that questions the status of the site itself.

But as these slippages and meshes that take place in software have a real world counterpart, are there new hybrid sites with which to engage? Cultural software such as Google Earth demands our curiosity because it reframes real places and layers our cities into new sites. As Bogost points out, the algorithm does not work in a vacuum:

> "It's not just mapping software running via computer—it also involves geographical information systems, geolocation satellites and transponders, human-driven automobiles, roof-mounted panoramic optical recording systems, international recording and privacy law, physical- and data-network routing systems, and web/mobile presentational apparatuses."[23]

There are many layers to our environments that are now pushed and pulled by their framing in software. Roofscapes become the primary elevations for architectures viewed by satellites: Dubai already expresses its economic and political hubris through buildings designed to be seen from space. Cultural software places new values upon our environments, and in turn effects not only the way we engage with them, but also the way we conceptualize them. Computation is never isolated from the world we inhabit.

My first case study is an undergraduate student, Chiara Barrett, whose work tackled precisely these issues—suggesting that our built environment is being mapped and recorded in new ways, and that we as architects should interrogate this.

Chiara's research was entitled *The Tenets of Google Picturesque*, and was produced as part of a studio I ran at The Bartlett School of Architecture in 2012, entitled *Facsimile*. We asked students to question the tools they would use to remotely understand a city from afar and propose modes of engagement that critiqued both the city as they understood it, and the digital facsimile of that city as presented to them. We then used a field trip as a method to test those assumptions, and compare the digital environments presented to us via the internet to actual situations on the ground.

Chiara's research sought to peel apart the ways in which Google Earth combined modeled geometry and texture mapping in its three-dimensional cartographic representations. As explained by Clement Valla, this combination is automated through a patented algorithm called *The Universal Texture*.[24] Chiara's work sought to explore how physical architecture placed into a city may provoke this algorithm into certain behaviors through its design. In this case then, an algorithm used by millions every day as part of a cultural software program is seen as something architecture could react to in physical space, a form of site condition.

Studying Los Angeles, Chiara developed a classification system for the different types of glitches that appeared while traversing the city using Google Earth. She identified how The Universal Texture behaves in certain ways in particular situations, thereby establishing a series of cause-and-effect scenarios within which she might engage.

In one example of behavior, *Façade Hierarchy* (Figure 3), the quality of mapped imagery varied between the size and perceived importance of roadways—main roads had higher quality mapping, producing buildings that had "slippages" between different resolutions across their facades. Taking a house in the Pico-Union district as a case study, she proposed disruptions to one face of the building, distorting and abstracting its joinery and ornamentation within a gap that Google Earth opened through the resolution slips of its texture mapping. The implication of this intervention is that, were such a building to be physically constructed, it would initially appear to be part of the errors that users of Google Earth would take for granted. Over time, with the inevitable increase in resolution and snagging of glitches, such a structure would now read as a physical anomaly rather than a digital one.

Another typology investigated through physical models, *Inside-Out* (Figure 4), dealt with how the algorithm appeared to collapse buildings and turn them inside out, producing strange interior conditions. This was revealed to users in early versions of the program's interior exploration ability, where the exterior texture of a building would appear mapped onto interior walls when the user entered it. A further study, *Vessel Distortion*, explored how buildings might react to the precise height and position of the Google Streetview car recording elevations and the depth of field this produced in photographic imagery. In these cases, the behavior of the Universal Texture algorithm gave Chiara new sites for engagement, straddling the virtual and the real.

As a result, study trips become particularly important, allowing one to judge the situation on the ground and make comparisons with the represented version of an environment viewed remotely. Being able to scale and record a place with our own eyes allows us to remain critical of the mediated versions of reality given by the proprietary systems of Google, Bing or Apple. As Bogost points out, when we use such software, we are experiencing the combined output of a milieu of governmental and corporate protagonists, each of which will have their own particular inflection on what we are allowed to see and interact with. In this case, we might see that there now exist three potential sites for architectural exploration: the real site, the mediated site and the *gap between the two*. Working in 2012, the gap was significantly larger for Chiara than that which exists today. But it still exists.

In Chiara's case, the exploitation of glitches became a methodology for the production of a design project for architectures that explored ruptures between the illusion of transparency and security in the built environment. As a student working in London, a city with one of the widest CCTV networks, the mediation of environments by gizmos and software is a particularly pressing concern. By blurring boundaries between a building and its facsimile, she developed an architecture critiquing Galloway's definition of our society of control: "Reflective surfaces have been overthrown by transparent thresholds. The metal detector arch, or the graphics frustrum."[25]

If our perception of physical environments has been changed by cultural software, my second case study explores how a virtual environment that was not created as a design tool has become a site for spatial experimentation that blurs the divide between "professional" and "amateur" architectural practitioners.

Crafting Habitat

If apps have changed how we conceive of the world, then video games have offered the possibility of alternative worlds, structured environments with their own logics and communities. My second case study is a Master's Thesis student I supervised, Marcus Stockton, whose project *Importance of the Block: Why Minecraft Matters* (2015) attempted to outline the impact the popular videogame has had on architectural design. *Minecraft* (Mojang, 2011) is possibly the most well-known videogame in the world, a landscape of colorful blocks where one literally mines cubes of terrain, and crafts using particular material combinations to produce a whole range of different built elements.

The saturated visual of its environment style melds *Generation Y* videogame nostalgia with the world of Lego, and has turned *voxelisation* into a trend reflected by cultural software such as Qubicle, which is a voxel modeling program developed by Minddesk that has been used in the production of further successful video games such as *Crossy Road* (Hipster Whale, 2014). Against the drive towards ever-more accurate simulations of environments, *Minecraft's* complex world is given over to a "toylike" aesthetic.

Minecraft has a "Survival" mode where players must make shelters in order to protect themselves from monsters emerging at night. But it is the free reign to build spaces and construct communities that has elevated it from a small-scale "indie" game into a global phenomenon. Throughout this process the feature-set of the game has grown, as has the user base, and the wealth of its creator, Markus "Notch" Persson. But many of the decisions on the features and logics of the game were developed through close conversation between Persson and the *Minecraft* community on internet forums. The *Minecraft* environment, and the creative possibilities it offers—taking it from a game to a piece of "cultural software"—were developed in collaboration between developer and community.

The game spent two years from its initial release in Alpha and Beta testing as a growing, unfinished product. During this time the community expanded and participated in this testing process, contributing their ideas to the software. As such, it became clear to Marcus that the only way he could adequately judge the game, and the environment it had established for architectural experimentation, was through using community websites as a key source.

Minecraft has a large community exploring the architectural possibilities the game offers. As Marcus discovered, it was the second most searched term in YouTube in 2014,[26] and there are a huge number of people sharing video tips and tricks for building structures within the game. Alongside this, there are a large number of internet forums, such as *Minecraft Builders Inc.* (Figure 5), with complex manuals for the constructions of certain building typologies and strategies for success in exploiting the randomly generated environments that *Minecraft* creates at the start of the game. The game not only establishes virtual environments that we can learn from, but also drives social environments online.

Marcus' research also utilized some of the *freeware* programs designed to transfer information to and from *Minecraft* itself. These modifications extend *Minecraft* from a game into cultural software proper under Manovich's terms—software for reading and writing information. One can use *Minecraft* to produce high-resolution renderings of architectural structures and landscapes using a render client like Chunky. Or one can export geometry from the game world in order to make full colour 3D prints at another scale using Mineways.

This extends into Printcraft, which is a *Minecraft* server that is directly connected to fabrication technologies—players building objects within the in-game environment that are then sent to 3D printers directly.[27] In Printcraft the environment of the game performs the function of the cultural software environment, while at the same time existing as a navigable virtual space. While Mineways operates with typical 3D modeling software, ripping geometry from the game and allowing one to open it in other programs, Printcraft allows players to build structures directly within an in-game zone that represents the building bay of a 3D printer. In this way it collapses the relationship between an experiential virtual environment and a scaled architectural model. Behaviors observed in *Minecraft* realms can be directly translated into a fabricated object.

Printcraft is but one collaborative system for the production of designs within *Minecraft*. Through participating in *Minecraft* communities Marcus was able to build a system for categorizing different types of behavior within the game environment, establishing a vernacular of sorts. He created typologies that showed how virtual buildings were produced in response to a number of stimuli. His examples varied from small shacks designed to ward off monsters to the "Precedent and Mimicry" (Figure 6) of real world structures,[28] and on to huge collaborative cityscapes, such as "Megacraft" (Figure 7), with their own laws and governments. Each of these typologies were collected by Marcus, rebuilt and recommunicated into 3D Studio Max, and rendered in a "generic" grey style in order to make comparisons between their morphology.

As Marcus discovered, to delve into the *Minecraft* community was to reveal the game as a popular tool for ersatz architectural design—which he termed "Rise of the Amateur."[29] While *Minecraft's* voxelised world, with its procedural generation, sets the basis for its material composition—what can be *mined*—it is the creativity of its user base that has turned its environment into cultural software proper. Seen literally, *Minecraft* is blocky and cartoonish in its transcriptions of natural landscapes and their material economy. However, it is an incredibly complex system—and seeing *Minecraft* as a new design environment, a surrogate for reality, shows it in a much more powerful light. *Minecraft* developer Mojang has already collaborated with the United Nations on a project called *Block by Block*, using the game to provide tools for residents in the developing world to have a say on the future of their towns and villages. Marcus also cited the *Blockholm* initiative by the Swedish Centre for Architecture and Design in 2013, which used the game to engage with citizens after having transcribed the city of Stockholm into the game. So there is increasing evidence for *Minecraft's* environment being a driver for ideas of how to shape our physical world, beyond the undoubted size and influence of its game communities.

Marcus' thesis was produced through virtual site work and reportage, exploring conditions on the ground within this virtual world. For many people, these communities and environments are real spaces where they congregate and collaborate—from *Minecraft*, to *World of Warcraft*, through *Grand Theft Auto Online*, to curiosities such as Tale of Tales' *The Endless Forest*. *Minecraft* and these other worlds demonstrate a need for us to recognize that future generations of architects and designers will be ever more familiar with cultural software and the communities that surround it. The possibilities for creativity, even within a video game environment, and how it challenges the divide between expert and amateur, pushes at the boundaries of what architecture might be and where we might decide to train our sights (or sites) as designers.

Conditions of Caricature

One of the main powers of videogames as (or as a product of) cultural software is that they twist versions of reality according to rules—much as *Minecraft's* rules instigated a landscape of free creativity. Games theorist Ian Bogost argues that all algorithms are *caricatures* in that they "take a complex system from the world and abstract it into processes that capture some of that system's logic and discard others."[30] For Bogost, videogames are the sole type of algorithms that celebrate the fact that they are caricatures, and this is their power as a critical tool. Indeed, Bogost himself has written at length in *Persuasive Games* about how videogame rules promote certain behaviors in players. Beyond *Minecraft* as an environment for architectural expression, we might ask what other video games have to say about the world and what caricatures of reality have they established.

My final case study is the work of Agostino Nickl, which was produced as part of a 6-week workshop at The Bartlett School of Architecture entitled *Pressure Drop*. For Agostino, an interest in the suburban drift of Chicago led to the search for cultural representations of suburbia. One of the most culturally pervasive but curiously abstracted representations of suburban life—with its economy of mass production and commodification—is *The Sims*, by Maxis, a 16 year old series of "life simulator" games. In an autobiographical move, Agostino chose to analyse the original *The Sims* (2000) because it was a game he played as a child, a virtual doll's house for a new generation of architects.

The Sims is effectively a consumerism simulator: *Sims* work to make money, and their happiness metrics are determined by objects placed within their suburban homes. The game is set in an area circumnavigated by a road named *Sims Lane*, later called *Pleasantview*. Although we see our Sims' day-to-day lives, we never see the world from their perspective, as in *Minecraft*, but rather from a disembodied "god" view. Watching a virtual life unfold from this close, yet detached, view is—as Michael Nitsche argues—akin to our obsession with reality TV.[31] Agostino was interested in how the suburban home and its objects contributed to these metrics of happiness, and how its cartoonish logics may be persuading us towards certain behaviors within its environment. In Bogost's terms, the game is *persuading* us to make our Sims into productive citizens, whose goals are to get a better job, to earn more money, to consume more, and repeat. Because this is a videogame it wears its caricature status on its sleeve. But by treating the logics of its world as if they were real, Agostino attempted to use it as a structure to regard the physical world.

As with Marcus' thesis, to draw out the suburban logics of Pleasantview Agostino had to play *The Sims*. And in doing so, he rediscovered its critical agency and relocated some of its *persuasive* aspects back into the real world. Through a series of calculations, Agostino revealed that the typical daily cycle of a Sim leaving for work to earn money and returning in the evening to spend it on commodities could be reversed. In fact, self-employment and working from home were more efficient ways to create wealth. He identified the "hobby" of making *garden gnomes* as Pleasantview's most efficient economic activity. Agostino found that a Sim at the maximum "crafting level" could produce 21 gnomes per day, netting themselves an income of $2100. Taking the economic environment established by the game to its furthest extent exposes the absurdity of its caricature. The production of throwaway objects and their impact on our real environment is held up to a mirror—the gnome, a

particularly useless object, is designed to sit outside yet never exposed to weather or degradation in the virtual realm.

But gnomes are a serious business in Pleasantview, as they are in the original town *The Sims* caricatures. On April 17th 2014, the Levittown police publically logged the theft of a *Philadelphia Eagles* themed gnome.[32] With an estimated value of $25 in reality, one quarter of the in-game value, it appears *The Sims* really has privileged the importance of gnomes to its economic system. As such, Agostino's response, a town entitled *Permaville*, sought to rebalance the spaces of suburbia through collective living predicated on the "gnomic" economy. The gnome became emblematic of *The Sims'* logics and, as the product of a Sim's hobby, ironically it is not too dissimilar to real working situations. As Galloway argues: "It is impossible to differentiate cleanly between non-productive leisure activity existing within the sphere of play and productive activity existing within the sphere of the workplace."[33] The constructive play of *Minecraft* is being monetized by forums such as *MCMarket*, and games such as *EVE Online* have a long history of real-world capital being driven through the economic systems of virtual environments.[34]

This reinforces the fact that technology cannot be separated from politics—we might encourage students to investigate futures in digital fabrication but, as Bogost argues, to realize how multitudes of people are still involved in the chain we only have to look at the world.[35] He references the *CHINA* photography of Edward Burtynsky, with its row upon row of workers in a world we are told is becoming ever more automated. So having revealed automated gnome-making as a lucrative profession within the game's environment, Agostino set out to place this digital caricature in the reality of labor by becoming a gnome maker himself.

Agostino constructed three gnomes to compare with the material economy of *The Sims*. One was carved by hand from timber, one 3D printed, and one CNC machined (Figure 8). By producing a time-lapse film, *Reality Check*, he was able to relate the rates of production between a handmade object, an object that requires formatting in cultural software in order to be machined, and how quickly a Sim would do an equivalent job. The gnome becomes an artifact that straddles the digital and the hand crafted. Each of these objects speaks to the unique design environment in which they were created, and reinforces the role that software has in their production. Working on the gnome by hand in the workshop, it could be crafted and reassessed at each step. On a CNC machine, complex tooling paths needed to be defined in software beforehand, and in the 3D printed version, a system of material supports were required for structural integrity. As per Bogost's definition of caricature, each one of these processes emphasizes one way of working and disavows others. Like some bug in the game causing the Sim to keep making broken gnomes, the CNC machine and 3D printer will keep marching on long after an error has rendered the structure unbuildable: their environment is entirely instructional.

After these material comparisons, Agostino continued to develop *Permaville* through drawings composed using screenshots from the game (Figure 9). Suburban plots were allowed to develop under the logic of the Sims' economy, a once idyllic landscape overflowing with gnomes. For Agostino, *The Sims* and its caricature of suburbia gave him a route into a critique of the real logics implicit in mass-produced and commodified architecture.

If this seemed an ironic application of videogame logic onto a suburban townscape, writers such as Galloway remind us that it is not so different from reality. Young people in China genuinely do farm gold in *World of Warcraft* as a new form of networked menial labor.[36] Gold can only be farmed because of the virtual environment established by the game makers. Virtual environments increasingly affect our real ones. Agostino's application of the gnome reminds us that cultural software bridges the gap between the virtual environment and the physical one, and that there might be embodied symbolism and politics in the objects that emerge through our digital tools—if we look deep enough.

Decoding

Design today is implicitly engaged in the use of software environments and environments created by software. Each of the case studies in this paper establish methodologies for exposing the complex relationship between what we would consider to be the real world, and those facsimiles we see on our screens and devices. We now have to be wary of sites and territories that are layered with new hierarchies of information by software, and the fact our remote access to these spaces is often given at the price of our personal information. So many people now obtain spatial experiences through totally synthetic virtual spaces that could not have existed before computation that we must understand the ways they pull at our expectation of environments we encounter in reality.

Perhaps the physical world is now not the only environment in which architects could be expected to operate, when the game engine offers the possibility to test spatial ideas and ecosystems before attempting to apply them to reality. And, by the same token, it might be that the proliferation of all those symbols, logics and hierarchies imposed by cultural software on our view of the world might confuse the issue, requiring us to break from the images we are given and peer inside the machine. If our environments continue to be so pumped full of information, saturated with data and encoded by proprietary software systems, then it falls to the architect to peer beyond and embrace *design by decoding*.

The projects that I have discussed share a desire to examine the way environments are presented to us through cultural software. Yet none of them represent digital architectural discourse in the normative sense, nor are any of the projects explicitly about breaking the programmed structure of such software through hacking or other code-based inquiries. They also do not tend toward examinations of the wider coded structure of the built environment defined through legal jurisdictions or other influences that underpins the work of Keller Easterling, for instance. Of course, such layered conditions are implicit in software like Google Earth, which is an emanation of a network spreading from satellites to algorithms, to camera vehicles operating under the local traffic laws. Were these projects longer, unpeeling further layers of these environments would surely be possible.

It might be argued that by not tackling code itself, our view into the machine is limited. And indeed it seems that codes, whether Easterling's political ones or the software engineer's algorithms, are still opaque to many. But by decoding the representational regimes of cultural software we can reveal and challenge the normative ways of working they might structure for us. As Manovich has pointed out, the history of cultural software is not the pure computer science of the coder, it has always involved interfaces for communicating with creative

users—users who, through that software, generate culture and reframe our world. These creative users help shape such tools even when they do not invent them, whether this is user feedback influencing the development of Adobe Photoshop—now in its 18th edition—or aiding the development of an open source application such as Blender. Each of the projects I have mentioned above has explored cultural software and game spaces through the eyes of an architect rather than the software designer, and this is where their criticality lies. Manovich's framing of cultural software means that we can consider and explore the spatial implications of the information we are given and, as designers, generate new spaces and environments even within the parameters of the "surface" tools provided by applications, should we care to look. And look we should, because digital environments represented to us by codes and worlds regulated by codes are now our other real worlds, and they are ever shifting.

Image References:

Figure 1: Drawing by Author, *Noclip World: Hall of Mirrors,* 2014.

Figure 2: Drawing by Author, *Noclip World: Dissolving Surface,* 2014.

Figure 3: Chiara Barrett, *The Tenets of Google Picturesque*. 2012. The Bartlett School of Architecture, UCL.

Figure 4: Chiara Barrett, *The Tenets of Google Picturesque*. 2012. The Bartlett School of Architecture, UCL.

Figure 5: Screenshot of "Minecraft Builders Inc.," https://minecraftbuildinginc.com/.

Figure 6: Marcus Stockton, *Importance of the Block: Why Minecraft Matters*. 2015. The Bartlett School of Architecture, UCL.

Figure 7: Marcus Stockton, *Importance of the Block: Why Minecraft Matters*. 2015. The Bartlett School of Architecture, UCL.

Figure 8: Agostino Nickl, *Permaville*. 2015. The Bartlett School of Architecture, UCL.

Figure 9: Agostino Nickl, *Permaville*. 2015. The Bartlett School of Architecture, UCL.

---

[1] Lev Manovich, *Software Takes Command* (New York, London: Bloomsbury, 2013), p.21.
[2] Ibid., p.27.
[3] Ibid., p.31.
[4] Alison and Peter Smithson, "But today we collect ads," Ark Magazine, 1956.
[5] Reyner Banham, *Design by Choice* (London: Academy Editions, 1981), p.110.
[6] James Bridle, "The New Aesthetic," *The New Aesthetic*, May 2011. http://new-aesthetic.tumblr.com (accessed 10 December 2016).

[7] Sam Loveridge, "GTA 5 sales top 54 million," Trustedreviews.com, 11 August 2015, http://www.trustedreviews.com/news/gta-5-sales-top-54-million (accessed 16 December 2016).
[8] Lev Manovich, *Software Takes Command* (New York, London: Bloomsbury, 2013), p.88.
[9] Ibid., p.96.
[10] Ibid., p.91.
[11] Ibid., p.142.
[12] "The Render Ghosts—James Bridle," *Electronic Voice Phenomena*, 14 November 2013, http://www.electronicvoicephenomena.net/index.php/the-render-ghosts-james-bridle/ (accessed 10th February 2017).
[13] "WAF podcast: Charles Jencks on generic individualism, 'the reigning style of our time'," *The Architects' Journal*, 17th May 2016, https://www.architectsjournal.co.uk/home/culture/waf-podcast-charles-jencks-on-generic-individualism-the-reigning-style-of-our-time/10006523.article.
[14] Lebbeus Woods, "Zaha's Way", http://lebbeuswoods.wordpress.com/2011/03/27/zahas-way/ (accessed 12th January 2017).
[15] Lev Manovich, *Software Takes Command* (New York, London: Bloomsbury, 2013), p.113.
[16] "The Totalitarian Buddhist Who Beat Sim City," Vice, accessed 18th September 2016, https://www.vice.com/en_us/article/the-totalitarian-buddhist-who-beat-sim-city.
[17] Ian Bogost, *Persuasive Games* (Cambridge MA, London: MIT Press, 2007), Kindle for iPad edition, Loc.84.
[18] McKenzie Wark, Gamer Theory (Cambridge MA, London: Harvard University Press, 2007), Note [030].
[18] Daniel Reynolds, "Virtual World Naturalism", *The Fiberculture Journal Issue 16: Counterplay* (2010), http://sixteen. fibreculturejournal.org/virtual-world-naturalism/ (accessed 20th December 2016).
[19] John Sharp, Works of Game (Cambridge MA, London: MIT Press, 2015), p.14.
[20] Patrick Klepek, "Grand Theft Auto V Looks Incredibly Creepy Without Textures," *Kotaku*, 29th January 2015, http://kotaku.com/grand-theft-auto-v-looks-incredibly-creepy-without-text-1682545969 (accessed 10th December 2016).
[21] Mark Dorrian, "On Google Earth," *Seeing From Above*, ed. Mark Dorrian and Frederic Pousin, (London: I.B. Taurus, 2013), p.299.
[22] Ian Bogost, "The Cathedral of Computation," *The Atlantic [online]*, 15th Jan 2015, http://www.theatlantic.com/technology/archive/2015/01/the-cathedral-of-computation/384300/ (accessed 5 December 2016).
[23] Clement Valla, "The Universal Texture," *Rhizome Blog*, 31st July 2012, http://rhizome.org/editorial/2012/jul/31/universal-texture/ (accessed 5th December 2015).
[24] Alexander R. Galloway, *The Interface Effect* (Cambridge UK, Malden MA: Polity Press, 2012), p.25.
[25] Marcus Stockton, "Importance of the Block: Why Minecraft Matters" (MArch diss., [name anonymised], 2015).
[26] "Make things with Minecraft," *Printcraft*, http://www.printcraft.org/ (accessed 15th January 2017).
[27] Marcus Stockton, "Importance of the Block: Why Minecraft Matters" (MArch diss., [name anonymised], 2015).
[28] Ibid., p.63.
[29] Ian Bogost, "The Cathedral of Computation," *The Atlantic [online]*, 15th Jan 2015, http://www.theatlantic.com/technology/archive/2015/01/the-cathedral-of-computation/384300/ (accessed 5 December 2016).

[30] Michael Nitsche, *Video Game Spaces: Image, Play, and Structure in 3d Worlds* (Cambridge MA, London: MIT Press, 2008), p.221.
[31] Tom Sofield, "Police Log: Gnome Stolen, DUI Arrests & More," *LevittownNow.com*, 27th April 2014, http://levittownnow.com/2014/04/27/police-log-gnome-stolen-dui-arrests-more/ (accessed 20 December 2016).
[32] Alexander R. Galloway, *The Interface Effect* (Cambridge UK, Malden MA: Polity Press, 2012), p.135.
[33] Erik Kain, "Massive 'EVE Online' Battle Could Cost $300,000 In Real Money," *Forbes*, January 29th 2014, https://www.forbes.com/sites/erikkain/2014/01/29/massive-eve-online-battle-could-cost-500000-in-real-money/#5bc6d0606877 (accessed 8th February 2017).
[34] Ian Bogost, "The Cathedral of Computation," *The Atlantic [online]*, 15th Jan 2015, http://www.theatlantic.com/technology/archive/2015/01/the-cathedral-of-computation/384300/ (accessed 5 December 2016).
[35] Alexander R. Galloway, *The Interface Effect* (Cambridge UK, Malden MA: Polity Press, 2012), p.121.