# The Ethics of Code in a Simlation World

David Parry, University of Texas, Dallas

## The Rise of Simulations

In January of 2003, during the height of the build-up preceding the invasion of Iraq, 3,000 military officers and civilian employees assembled in Germany for a strategic planning exercise called "Victory Scrimmage." Much of this "scrimmage" was dedicated to the running of large computer simulations meant to predict and help shape U.S. invasion strategies. And while not everyone agreed with the simulations that promised a swift and overwhelming victory, few disagreed with the idea that a computer simulation could help shape U.S. policy. That is, few questioned the predictive possibilities of modeling the outcome via large scale computer simulations.

In June of that same year, Linden Labs launched what was to become the most popular online world, *Second Life*. A large "open" simulated 3-dimensional world, complete with its own currency, *Second Life* allows players to create an account, develop an avatar, and engage with other players in a simulated space. Within *Second Life* players can purchase land, erect buildings, socialize, and even engage in virtual sex. A significant departure from other Massively Multiplayer Online games, *Second Life's* popularity was not based on a fantasy world where players developed powerful characters to wage combat against demons and dragons (*Everquest* or *Ultimate Online*), but precisely the opposite: the simulation of a world in which players were free to construct their own meaning. Indeed, players have repurposed the environment for everything from educational meetings to political demonstrations and virtual museum openings. Seemingly any event or behavior from the "real" world is modeled and reproduced in *Second Life*.

While much has been made of the military industrial complex, or, perhaps more precisely, the military entertainment complex, the link I want to draw from

GENRE XLI - FALL/WINTER 2008 - 107-124 

the above two chronologically linked moments is not about how military technology is increasingly approaching entertainment technology. The connection I want to make is rather one about how the power of simulation, or rather the belief in the power of simulation, reshapes the way that we come to think of the world, into what Katherine Hayles calls a "computational worldview." According to Hayles, society now understands the world as a computational process which via the power of code can be modeled in simulation. Certainly the military believes in such a possibility, spending millions of dollars to create simulators for training foot soldiers and increasingly make decisions about policy based on simulations meant to predict outcomes of tactical and strategic decisions. And while there might be a perceivable difference, for the moment, between the battlefield trainers run on computers and real combat situations, or between real world events and those produced in *Second Life*, as the technologies of representation advance such differences recede at an ever faster rate.

As technologies evolve, increasing in abstraction, they change the possibilities both for communication and for the goal of producing a rigidly determined communication, a communication which would not be subject to the errors and misinterpretation one finds in language, a form of representation which would reproduce without difference that which it is representing. Where a linguistic description of an event might not actually depict the entirety of that event because it is subject to limitations of language and individual perspective, simulation purportedly allows the reader to relive the entirety of the event, not only as it happened but all other alternative possibilities as well. The ideal of simulation then becomes to bring not only the past events under control in the guise of representation, but, in another sense, to bring the future, or all possible futures, under control by predicting events and outcomes.

Clearly there is some slippage and overlap in terms here. For, from one angle narrative is just one aspect of simulation. A dramatic re-enactment or a staged play is meant to simulate an event that is not present. From this perspective, narrative is but a subset of simulation. But it is equally important to distinguish between these two representational modes, especially given the particular prevalence simulations are taking in our digital world. The view that the world is simulatable given enough computational power is a world metaphor which is rapidly replacing the notion that one learns of the world through narration and linear representation.

Western narrative has a rather strong connection to the view that art is mimetic (language and art as technological tools), re-presenting reality, that one can learn of the world through these representations, and finally that art therefore comes to have a uniquely instructive role. In one sense the field of literature and literary criticism is based on this connection, which assumes that there is specific knowledge of the world to be gained by studying objects that represent and recast the world in narrative and aesthetic terms. Narrative initiates a specific causal relation to time, recasting events that are singular as continuous and producing a whole from which a reader can make sense of the represented events. Most importantly, narrative ought to be understood as a particular way of seeing and making sense of the world, a way of representing in an ordered way what initially appears as chaotic.

However, narrative is losing its place as the primary metaphor by which chaos comes to be ordered. The material supports that structure and contextualize such representations necessarily frame and help to delineate, although not to determine finally, the range of meanings communicated by a work. Thus as technologies of representation change, it should come as no surprise that the content of those representations themselves will change. As Walter Benjamin argued in "The Work of Art in the Age of Mechanical Reproduction," "During long periods of history, the mode of human sense perception changes with humanity's entire mode of existence" (222). As our mode of existence has been transformed from analog to digital support structures, our perception of how that knowledge is ordered, represented, and archived for further exploration, has significantly shifted.

As Ian Bogost suggests in *Unit Operations,* the dominant mode of representation, narrative, has been supplanted in the age of the digital by a new world metaphor. Whereas narrative might have served as the primary representational model for seeing and making sense of the world, for ordering chaos in the age of the codex book, it no longer serves as the chief conceptual or cognitive metaphor for the ordering of knowledge or data. No longer are cognitive projections limited to what we can narrate, Bogost argues, but, given the computational ability of modern computers and the vast databases of knowledge they contain, we can run simulations, demonstrating not only what happened, and why it happened, but also showing what might have happened. Bogost's position is not that simulations are in fact "true," providing a more accurate ordering of chaos than narrative, but rather that simulations have now become the dominant model of

representation in public consciousness. Clearly in the case of the military this is true as they invest millions based on the promise of simulations, but this is also true of smaller scale public decisions such as city planning, where simulations are now used to model the effects of any policy or construction decisions. Having a simulation of any proposal has now become a powerful rhetorical tool ("We have run the simulations and they suggest we ought to . . ."). Bill Nichols echoes Bogost's position in his aptly titled refiguring of Benjamin's essay: "The Work of Culture in the Age of Cybernetic Systems." Nichols contends that prior artistic moments fetishized the consumption of the object, whereas now we fetishize the process of consumption, a move from the "to-be-looked-at-ness of a projected image" to the "control of simulated interactions" (632).

One does not have to turn to digital games, as Bogost does, or to military simulations such as the "Victory Scrimmage" to see how pervasive this metaphor of the world as "simulatable" has become. The 1999 film *The Matrix* perhaps serves as the most popular representation of this idea, where internal human consciousness can be simulated on a giant computer, simulated to such an extent that all but a few participants are unaware that they are "in reality" only experiencing an effect of binary code. Outside of fictional narratives as well, simulations are increasingly used as learning tools that work precisely on the logic that simulating a real experience is possible via elaborate computer aided structures: pilots receive hours of training on simulators prior to ever flying real aircraft, doctors perform digitally simulated surgery, and future officers are trained in *Second Life* on how to police border crossings. Scientists no longer speak of empirical tests as the only way to gain knowledge of the world, but rather turn equally to supercomputers to run elaborate models, to predict climate change, or understand the evolution of life on this planet.[17] This ability to simulate extends in both macro and micro directions, where simulations can be used to understand complex astronomical events, and to be used to explain how cellular automata and DNA interact. A metaphysics shaped by narrative becomes a metaphysics shaped by simulation, with the result that the means by which we negotiate the signifiers of language or code that compose the respective worldviews have also necessarily changed.

[17] See for example the July 2008 issue of *Wired* magazine, where Chris Anderson argues that science is undergoing a fundamental shift from being driven by experimental practice to being driven by collecting and processing large amounts of data, resulting in a science which Anderson labels "The Petabyte Age." In the terms at hand science is moving from interpreting single runs (narratives) towards using massive amounts of data to run simulations.

## The Simulation Worldview

As with other metaphysical metaphors, the world as simulation is a result of technological change and subsequent cultural assimilation. The simulation view of the world is predicated on the increasing cultural prevalence of computers and computational technology. Janet Murray's *Hamlet on the Holodeck* (which takes its name from the simulation technology of *Star Trek's* holodeck) encapsulates the utopic hope that computational technologies will lead to a significant alteration in our forms of artistic representation. Fundamental to Murray's view is that "computers offer a thrilling extension of human powers" and "greater control over different kinds of information" (6-7). In this vision, simulation signifies the eschatological end of narrative: while once all of life was a story to be recounted in narrative form, now all narratives become subsumed under the wider technological possibilities of simulation, and all life becomes a simulation subject to computation and replayability. To be fair, Murray does not state the case in these extreme terms, but as Bogost points out, while she focuses on how simulations are interpretations of the world, her ideas can be easily extended to seeing all narrative texts as merely one version of a simulation (71).

As I suggested, simulation rhetoric tends to position it as the eschatological end of narrative; that is, as not only what reproduces the possibility of narrative—any simulation can encapsulate a narrative—but what exceeds narrative by purporting to contain all of the possibilities of any given world. According to this same line of reasoning, simulation becomes the privileged term, the superior model of representation. Whereas narrative is linear in representation, a one-way transaction which, although it might allow for complex causal relations, is predicated on a unidirectional telos—narrative as progressing from beginning to end—simulation is determined by "feedback loops." Its casual relations are not necessarily unidirectional, with cause and effect mapped in a far more complicated manner, enabling analysis beyond a single teleological progression, not limited to the trajectory of beginnings and endings. This is commonly featured as replayability, whereby a simulation is executed or run again, to produce or predict other variants. Predictably in this line of thinking, narrative then becomes associated with the continuous and analog, whereas simulation is associated with the discrete and digital. This is not to suggest that one cannot have digital narratives, or analog simulations, but rather that in the case of narrative, narration occurs according to a continuous logic, a narrative even if presented on a computer screen, is produced by a causal relation one might frame as continu-

ous; and in the case of simulation, simulations are made possible only inasmuch as events are discrete and separate in order to be consistently refigurable. Narrative also proves to be more closely associated with the constative, describing what happened, whereas simulation is performative, enabling a happening. And finally, according to this division, narrative is a feature of verbal language, while simulation is ultimately a feature of numeric code—the crucial distinction in this line of reasoning.

More to the point though is the belief that these simulations, through the saturation of context (inclusion of all variables with in a system) gain a predictive power (again the reason military organizations are willing to spend vast sums of money to run simulations of possible and future conflicts). This predictive power is a result of the underlying structures of the simulation. While narrative representations are underwritten by language, an inherently inadequate and shaky ground, simulations are underwritten by the power of code, a supposedly fixed and determined form of representation not subject to the indeterminacy of language.

While narrative can only describe events, past or future, computational simulation is valued for its predictive power. That is, rather than simply narrating and positing a series of events, the representational power of simulation comes from its ability to continually generate multiple narratives of what might be, running thru endless simulations in order to view all possible outcomes. Simulation thus tackles what seems to be beyond the pervue of narrative, not describing a future which might occur (narrative) but rather modeling the path to a particular future, predicting the process by which said future is obtainable. *The diegesis of narrative is replaced by the power of multipling mimesis.*

This predictive ability of simulation extends to the past as well, where narrative might only describe events that did occur or might have occured, computational simulation reflects all of these possibilities, containing within the database multiple versions, and multiple causalities, describing the process which produced the events rather than the events themselves. Thus with simulation one gives up the hope that narrative can accurately depict an event, present a subject with a complete account of the situation, replacing it with a control over process. No longer is the description of the event what is archived, but rather what is archived are all of the forces and processes which led to the production of the events. The inadequacies of narrative are overcome by the power of digital

simulation unleashed by code. By repeating the process, replaying the situation one approaches a better understanding of reality.

Relying on the narrative/simulation and language/code distinctions critics often rhetorically equate simulation with technology and technological advance, where often simply the ability to perform large scale computation, enables simulation. Once simulation is cast as technological, it is suggested that narrative is pre-technological, the more "natural" form of communication. Thus we get conclusions that "narrative is much older than simulation modeling in artificial media—almost as old, many anthropologists believe, as the human species itself. Narrative, with its evocation of the human lifeworld, speaks to subjectivities that remain rooted in human perceptual systems, human languages, and human cultures" (Hayles 6). Simulation becomes aligned with the technical, or the advanced, and narrative with the pre-technical, originary, natural human.

By maintaining a distinction between narrative and simulation, language and code, one subjects each to different interpretive modes. Narrative is subject to hermeneutic criticism whereas simulation manifests a different case, for the object itself is not fixed; there are multiple configurations (indeed, simulation is predicated on the notion that multiple configurations are possible). One no longer negotiates the meaning given by the entire work, but rather how meaning is produced. Accordingly, in simulation the question becomes how multiple meanings are produced from such a structure, and what works are possible given its rule set (a question of algorithm and code not representation). A simulation is thus able to claim a power that exceeds narrative, for it preserves not just one discourse but all possible discourses, whether real or merely a digital hypothesis. What narrative excludes in order to constitute itself—the other possibilities—become part of simulation, included within its frame. Of course, there is nothing particularly new in the view that the world can be simulated. Museums and historical villages operate according to just such a logic of simulation. But the digital, and specifically the computational when paired with the digital, extend the degree to which these simulations can be run. For, if the power of a simulation is increased by the rigors of code and the ability to model complexity, as computational power increases so does the ability of simulation to mimic the "real."

These simulated worlds are always figured as places of infinite complexity, the ability to model quantum variability over Newtonian simplicity. Limits in the simulation world are merely arbitrary and easily overcome by a change in code. But if the labyrinths that compose these simulation worlds are seen as infinitely

complex, the complexity is always subject to absolute control on the level of the code, which is supposedly free from arbitrariness, and understandable in the simplest of terms as composed of zeroes and ones.

However, rather than see simulation as a radical break in the history of representation, one that constitutes a substantial change from narrative, I would argue that simulation itself is part of what should be understood as the history of archivization, another historical technique used to archive experiences. And despite the notion that simulation will yield to a perfection of this practice of archivization, exceeding narrative, by developing an archive that could not only preserve the past, but also preserve what might have been, and what will be, I want to argue that simulation is still reliant on narrative, *a narrative told about the power of code*.

## Code and the Computational Universe

In her most recent book, *My Mother was a Computer: Digital Subjects and Literary Texts*, Katherine Hayles argues that a fundamental ontological shift has occurred, which, to a large degree, is coextensive with the move from narrative to simulation, and which, although she does not directly argue as such, is crucial to the "simulation fever" described by Bogost. Underlying the ubiquitous presence of simulation technology is what Hayles calls the "Computational Universe." One of the central tenets of digital simulation is the belief that one can discern and represent the world by means of a series of algorithms. More fundamental than the notion that the universe can be represented as computations, i.e., simulated through equations and the rules governing them, is the belief that the world itself is a computational program, "that the universe is generated through computational processes running on a vast computational mechanism underlying all of physical reality" (3) and that the computational universe, "works simultaneously as means and metaphor in technical and artistic practices" (4). Crucial to this analysis is the recognition that simulation and computation are not simply metaphors for understanding the universe and even human consciousness, but rather constitute a shift in ontology: *the world is a computational process*. Thus a one to one equivalence can be established between tools—computers as world replicators—and the world as a computer; our instruments for understanding the world and the world are in fact one and the same. Computation now becomes an ontology and a metaphysics accounting for all phenomena and thus the ideal way not only to represent, but more fundamentally to understand, these process-

es, from the sub-atomic level of the nano to the macroscopic level of universal scale.

Hayles shows how this series of beliefs, regardless of their "truth" function, shapes the social and comes to be treated as originary. While there is much of value in her thinking through the complex issues of digital subjectivity, especially the careful analysis of how subjectivity is an effect of multiply mediated effects and recursive loops, rather than simple linear causal structures, what is important for our analysis here are the ways in which Hayles separates narrative from simulation: "whereas computation is essential for simulations that model complex phenomena, literature's stock-in-trade is narrative…. Narrative is much older than simulation modeling in artificial media" (6). So, for Hayles there is something natural if ambiguous about narrative, whereas simulations rely on computational power to enact a more artificial determinate representation.

As Hayles's division between narrative and simulation, language and code, shows, code fulfills the crucial role of establishing the explanatory power of the computational universe and the growing ubiquity of simulation as a representational model. Hayles summarizes the implications of this idea:

> The Regime of Computation, then, provides a narrative that accounts for the evolution of the universe, life, mind, and mind reflecting on mind by connecting these emergences with the computational processes…. This is the larger context in which code acquires special, indeed universal, significance. In the Regime of Computation, code is understood as the discourse system that mirrors what happens in nature and generates nature itself. (27)

The computer becomes the ultimate technical instrument—one might even say the eschatological one—for it is not only a tool itself, but a tool that possesses the ability to replicate all other tools, to simulate their function.[18]

In one of the chapters that develops the conceptual groundwork of *My Mother was a Computer*, Hayles considers three "worldviews" of communication and representation: speech, writing, and code. (It is worth noting here how this treatment of representation is instrumental in approach, seeing each moment as a separate tool set that can be analyzed from a humanist perspective based on its ability to communicate a message accurately.) Her position here is that while there is a great deal of work that has been done analyzing the differences

[18] This is, of course, related to the Universal Turing Machine, the theoretical construct on which much of modern computing is based. The Universal Turing Machine is a machine that can simulate itself and all other machines.

between speech and writing, more needs to be done to differentiate code from these two prior systems of representation, for as a system of representation code is *fundamentally different*. "Code has characteristics that occur neither in speech nor in writing processes that, by exceeding these legacy systems, mark a disjunction" (57). It is in these characteristics that we come to learn the stakes of the argument, for these characteristics empower computational simulation to exceed the reaches of narrative. The stakes here are large, for thinking the digital and the technological is inseparable on some level from thinking through code, and if code is different from, or exceeds, these other representational models, then indeed narrative will be exceeded by computational simulation.

## Code, Writing, Speech; Speech, Writing, Code

I take the extent to which code can be separated from writing and speech to be the central turn in Hayles's argument. Her fundamental claim is that "code *exceeds* both writing and speech, having characteristics that appear in neither of these legacy systems" (40, emphasis added). Although she does not claim outright that code is a superior representational mode, her terminology certainly suggests that she grants it a certain privilege: "the *progression* of speech to writing to code" marks the telos she sees as operating here (39, emphasis added). Indeed, Hayles argues that writing and speech are now legacy systems, that commonly accepted ideas about signification need to be re-evaluated in the context of coding technologies (39). The thrust of much of her analysis seems to be to carve out a space beyond the regime of postmodern views of language; to suggest that contrary to postmodern or poststructuralist thought (that which troubled narrative), there is something like, a structure of material logic to code below the level of linguistic signification (that which might produce simulation).

For Hayles, unlike other systems of representation, code is both determinate and decidable. An inversion takes place, whereby natural languages are made to be a result of code, despite code evolving from written language.[19] Hayles claims that code produces a seismic shift in how we understand representation, "that code and language operate in significantly different ways" (15). The primary difference between code and language, or so she argues, is that language,

[19] In one sense this is the conceit of Neal Stephenson's *Snow Crash*, one of the first science fiction novels to envision the possibility of a simulation world such as *Second Life*. The "virus" at the center of the novel is a code which underlies human language, a universal communication, which if accessed allows another to directly "hack" into another human's brain and control their actions. Communication sans language, a return to a moment prior to the Tower of Babel.

as Derrida articulated in *Of Grammatology*, relies on an intrinsic slippage, which does not occur on the level of code. Code, supposedly, does not "tolerate" the grammatology that Derrida described (*My Mother was a Computer* 46). Whereas in language, Derrida is able to talk about the infinite deferral of signifiers, with code "it makes no sense to think of signifiers without signifieds" (47). The result of this viewpoint is a suggestion that code not only cancels ambiguity, but it also achieves discreteness in a way that is unavailable to language. The meaning is precisely determined by context, eliminating what Derrida has argued is the structuring ambiguity present in all linguistic utterances, and thus despite its constructed status code, actually supplants "natural" language, leading to the power of simulation.

The implications of Hayles' argument are significant. If her analysis is correct, code "inherits little or no baggage from Western Metaphysics" (51), and she has effectively found a way to talk about representation sans recourse to postmodern undecidability. According to Hayles, code differentiates itself from language (written or spoken) based on two features. First code is not iterable, but rather absolutely fixed by context. In her critique of Derrida, Hayles writes, "although Derrida asserts that this iterability is not limited to written language but 'is to be found in all language' (*Limited Inc,* 10), this assertion does not hold true literally for code, where contexts are precisely determined by the level and nature of code" (48). The suggestion here is that code, unlike language, relies on context such that if it is transported to a different context it becomes unintelligible. Whereas in language one might suggest that by transporting bits of speech to a different context one changes the meaning, in the case of code context is so absolute that any alteration in context removes the ability of the code to mean. And thus, "regardless of what humans think of code, the machine is the final arbiter of whether the code is intelligible" (50). Further, because code is "performative in a much stronger sense than that attributed to language" and it "is the only language that is executable," it "stands in striking contrast to the communities that decide whether an act of speech or a piece of writing constitutes a legible and competent utterance" (50). For if code is not iterable, but rather absolutely fixed by context, only executable, and purely performative, then there is indeed a way to guarantee transmission of knowledge and information, to ensure that a message arrives, even if ambiguity is reintroduced when a given piece of information is translated into the imperfect system of language.

But as Hayles aptly points out and then seems to forget, Derrida's critiques consistently show that just when we think we have exorcised Western metaphysics, it sneaks in the backdoor (17). The first problem in Hayles's structure is the contention that code is unambiguous, determinate at every point and thus not subject to undecidability.[20] Yet code itself is an abstraction, the effect of a decision to conceptualize and reduce logical operations to ones and zeroes. So, before there is anything like the binary division of code on which a machine can run, a decision is made that structures the governing logic of the system. Zeroes and ones are matters of decision, not pregiven to the machine. The binary division that lies at the base of the computational model is not natural; it is a model crafted for the precise purpose of reducing all difference to the space between 0 and 1 in order to aid commutation. Yet this decision already reveals a host of prejudices about the way that the world is structured: in essence, the binary computation model is itself an effect of narrative discursive production. At the heart of the abstraction of code is the notion of reducible binary difference. That is, the "1" can be absolutely distinguished and opposed by the "0," and, more importantly, all matters of representation are reducible to a binary representation. As David Gunkel explains, this is not a line of thinking invented in this current digital moment, that indeed the binary has a long history in western metaphysics. But, as Gunkel concludes:

> [Even] the seemingly indifferent 0 and 1 that comprise binary information encodes, at least for our purposes, moral decisions: no as opposed to yes, off as opposed to on, false as opposed to true, and nothing as opposed to something. Consequently, binary oppositions are neither impartial nor indifferent. (153)

We do not have to go to far down this line of reasoning to realize that the determinacy achieved in code is based on a metaphysical bias which privileges binary thinking at the expense of more nuanced approaches.

Code exceeds writing to the extent that it enables the representation of all series of marks as a simple opposition between 0 and 1, where 0s and 1s can then come to mean, to transmit precisely all other forms of written communication, regardless of content. By distilling representation to this absolute binary difference, the "1" or the "0," code is able to remove any ambiguity. But on

[20] Hayles does not recognize how determinate and decidable are not coterminous. For something might be determinate, yet remain undecidable: "Undecidability is always a *determinate* oscillation between possibilities" (*Limited Inc*, 148).

the other hand, writing exceeds code (in ways Hayles does not recognize), for at every instance, in order to operate, code must subject itself to being marked, to a marking, to a separation of zero from one, not only in space, but in time as well, and thus to all the logics of writing and iteration. In order to work, code itself must be materialized, subjected to the workings and trappings of the written mark. It must possess a technological embodiment. The writtenness of code need not be limited to its textual presentation of a series of zeroes and ones on a computer screen, application or written document, since it also refers to its being written in the sense of marked, as is the case in a differences of voltage, even if these differences do not appear on paper or are not perceptible to humans.

But, more importantly, the means by which these coding structures are represented and materialized are themselves matters of decision, interventions intended to eliminate ambiguity. One can cite in this regard Hayles's example of the circuitry that constructs a computer. According to her account, digital computers eliminate ambiguity by rigid structures that parse voltage into significations of zeroes and ones. Thus real material differences in voltage are changed into signification, decided as either present or absent. This is why "in the worldview of code, it makes no sense to talk about signifiers without signifieds … because every change in voltage must be given an unambiguous interpretation, or the program is likely not to function as intended" (47). We can note a few crucial points here. First, there is the claim that signification cannot be infinitely deferred because voltage requires a precise meaning, a precise material condition. But, ambiguity is removed only because every "change in voltage *must be given* an unambiguous interpretation." Because a decision must be made, because the system cannot tolerate ambiguity, as Hayles points out, any variation in voltage must be eliminated. The circuitry that represents those 0s and 1s is actually measuring a change in voltage. For example it translates a change of .9 volts into a 1 and a change of .2 volts into a 0, but what about half a volt? Because the circuitry and computer cannot allow ambiguity, there must be a decision that specifies what to do with any ambiguity. But this is an arbitrary decision. In other words, a decision is made to stop the chain of signification and restore the presence of the signified. Who makes it, or how this decision is made is absent from Hayles's account. And since this is a decision which is made without an absolute reference to a signified, there must be ambiguity within the chain. Just as importantly, in the end, if this does not occur the "program is likely to *not function as intended*." Yet the program will function, communication

will happen. The message might not be the same, it might not arrive, it might not be as intended, it might appear to be complete noise, but nevertheless it will function, even if that function is the blue screen of death. Intent here is returned at the final moment to recycle and guarantee communication; the intended function happens because it was intended.

In fact, far from not being iterable, code is radically iterable, constituted of only zeroes and ones repeated in long strings. The marks of code are often repeated—albeit in different contexts, but nonetheless repeated. As Adrian MacKenzie has argued, despite its formality as rule-governed expression, code…is an unstable volatile medium" (*Cutting* 6). Far from containing a text that would provide a stability of meaning, context subjects the coding structure to even greater ambiguity, for at every moment more context is required. Anyone who has experienced computer crashes, viruses, or bugs, realizes this, and the only thing that can indeed restore meaning is a reference to the "function as intended" (47). But this is precisely the point: any system or structure of signification might not function as intended—indeed, must contain the possibility of not functioning as intended for it to function at all. Intention cannot "govern the entire scene and system of utterance" (Derrida, *Limited Inc* 18).

Code does not eliminate ambiguity, as Hayles suggests, becoming absolutely determined by context, but rather the reverse is true: code, perhaps to a greater extent than writing, relies on context and is thus more ambiguous, increasingly subject to slight contextual variants. Code only operates inasmuch as more code is present to contextualize it; the context of code is always more code. The more we move up the "Tower of Babel" of code, to higher-end object-oriented code, the more one needs other code to provide context. The more one codes, the more there is to code. Code means what it means when it means and it works as intended, but this means neither that the meaning is guaranteed nor that it will function as intended.

As Adrian MacKenzie argues, "the act of trying to demarcate what can be encoded from what cannot be encoded sets in motion a process whose end can never be guaranteed" (*Undecidability* 367). Thus in spite of, or rather because of, code's computational and predictive powers, there is always a remainder, something beyond prediction, which the system cannot address. This undecidability affects our very ability to articulate and demarcate the computer and hence the digital, code, and computational possibilities. And, most importantly, it affects the regime of simulation, for despite the predictive claims of infinite

computational power and elaborate systems, the coding is based on a certain inclusion and exclusion.

Negotiating the indeterminacy of code might seem like a minor matter when we are talking about models such as *Sim City*, or playful virtual worlds such as *Little Big Planet*, or even when speaking of social interactions in *Second Life.* But the inherent ambiguity and indeterminacy of code, and subsequently any computer simulation, becomes a more urgent matter when we realize that political actors now increasingly rely on the power of simulation to execute choices on small matters, as well as large ones such as how and when to resort to state violence. It might be comforting to separate *Spore* from something like the "Victory Scrimmage," but ultimately the metaphysical conceits which underly these systems are closer than we would perhaps like to admit.

Despite the concrete and increasingly discrete and measurable forms digital representations take, we should not forget how any representation contains the possibility for failure: "We have to say that the electrical impulse falls within it, structurally speaking, within the possibility of miscommunication…that at the moment that the electrical impulse is emitted its destination is not assured" (Wills 121). Indeed this is precisely the point that analysis that privileges code tries to ignore. In the case of the electrical impulse, the message is always said to arrive. That is the very power upon which the computational model of the universe relies, and, by extension, the simulation archive. The power of simulation over narrative tries to exploit this divide, to claim that while language might be a-destinational, an electrical pulse is not. However, the pulse itself is structured by the same idea as the mark: a change in time, a movement in space, into which ambiguity and undecidability will necessarily enter.

I am not suggesting that code be collapsed into writing; certainly not. Clearly code and writing have different structures, are contextualized in different ways, play different roles in knowledge creation and dissemination, and operate in and through different material support structures. What I am suggesting is that we avoid seeing either as anterior or posterior to the other, seeing one as exceeding the other. To claim that code is somehow absolutely determinate and not subject to intervention, that it can only be intervened with prior to and after its executing, is to give ourselves over to simulation, to be determined by a binary code of presence and absence without paying attention to how these elements are themselves metaphysical constructs. We ought not to elide such an analysis if we want there to be anything like a politics of negotiating code.

Beyond all of this examination of the iterability of code and its reliance on a structure of signification, it is important to recognize one final important point: The regime of computation is itself an effect of narrative, a narrative told about the power of code to describe the present, preserve and discover the past, and predict the future. That is, the regime of computation itself relies on the power of narrative. Even Hayles recognizes this, as demonstrated by a passage I cited earlier: "The Regime of Computation, then, provides *a narrative* that accounts for the evolution of the universe, life, mind, and mind reflecting on mind" (27, emphasis added). The power of code does not come from an intrinsic power to describe the nature of the universe, but rather itself is an effect of tracing causal relations, a discursive system made possible only inasmuch as there is a narrative that gives it power. Code is narratively described as the final moment in technical advance, a progression of systems of knowledge preservation and world modeling that excludes all ambiguity. Despite claims that its simulative power exceeds prior models of representation, the force on which it relies is narrative, as it fulfills an eschatological vision founded in the rhetoric of beginnings, endings and progress. The power of code is a discursive effect of narration, not an intrinsic feature of 0s and 1s.

Just as the need to negotiate the complex web of signification seems to be exorcised, and the bias of Western metaphysics conjured away by a reliance on the deceptively simple structure of code, the need—indeed, the requirement—not to leave these questions aside, to continue to shuttle and transfer meaning, to negotiate, returns. And in ever more complicated terms, for no longer are we merely negotiating the recounted story, but now a multiplicity of recounted stories and rules that govern these recountings: simulations meant to capture the past, and simulations meant to predict control and determine the future. While it might be comforting to see massive computational simulations as finally descriptive of our world, finally able to predict the future and preserve the past, in fact what we now have is an increasingly complicated structure, one that produces more meaning by producing ever more context to negotiate. Indeed, there are more places to intervene, to stop, to pause, and to replay, even if these opportunities are presented in accelerating fashion, so fast as to hide their opportunity. What simulation opens up is the possibility of recognizing all of the places where we are required to intervene, to interact, to negotiate with the text.

Computational simulations might promise a predictive power that exceeds the limitations of print narrative, the ability to represent multiple narrative

trajectories, to reduce, to quantify, to make calculable all of their stories. And simulations might also promise a power to predict and control the future; certainly, a representative power that appears to exceed narrative. Nevertheless, such modes of representation, models for communicating, predicting, preserving knowledge and experience are still very much structured by what we can recognize as markings, tracings, which is to say moments that require another for existence, require the negotiation of an other. Such moments of negotiation are more or less present in any given medium, print or digital, and figure across all textual representation.

In *Echographies*, responding to a question by Stiegler, Derrida reminds his readers that one cannot extend the means of representation without also simultaneously extending the support structures that make meaning possible: "meaning and intelligibility can be extended—on the scale of what you have called the 'discrete,' the spacing of the discrete—only by multiplying the conditions of these very discreteness, in other words, spacing, non-sense, the blank, the interval, everything that bounds [*borde*] sense and non-sense as it were, exceeds [*déborde*] or splits it" (108). And so it goes with the digital. Although such technologies extend the possibilities of representation, they still are part of the history of representation, which is to say they are governed by the logic of iteration. And while the age of the digital might multiply moments and possibilities of representation and thus offer and multiply possibilities for interventions, it also passes over them with greater speed and with smaller distance in the space between the 0 and the 1.

## WORKS CITED

Anderson, Chris. "The Petabyte Age." *Wired.* July 2008. 106-121.

Benjamin, Walter. *Illuminations*. New York: Schocken Books, 1986.

Bogost, Ian. *Unit Operations: An Approach to Videogame Criticism*. Cambridge: MIT Press, 2006.

Derrida, Jacques. *Limited Inc*. Evanston, IL: Northwestern UP, 1988.

—, and Bernard Stiegler. *Echographies of Television.* Cambridge, MA: Polity Press, 2002.

Gunkel, David. *Thinking Otherwise: Philosophy, Communication, Technology.* West Lafayette, IN: Purdue UP, 2007.

Hayles, N. Katherine. *My Mother Was a Computer: Digital Subjects and Literary Texts*. Chicago: University of Chicago Press, 2005.

Mackenzie, Adrian. "Undecidability: The History and Time of the Universal

Turing Machine." *Configurations*. 4.3 (1996): 359-79.

—. *Cutting Code: Software and Sociality*. New York: Peter Lang, 2006.

Murray, Janet Horowitz. *Hamlet on the Holodeck: The Future of Narrative in Cyberspace*. Cambridge: MIT Press, 1998.

Nichols, Bill. "The Work of Culture in the Age of Cybernetic Systems." *The New Media Reader*. Ed. Noah Wardip-Fruin and Nick Montfort. Cambridge: MIT Press, 2003. 627-41.

Stephenson, Neal. *Snow Crash*. New York: Bantam Books, 1992.

Wills, David. "An Interview with David Wills." *Homo orthopedicus. Le corps et ses prothèses à l'époque (post)moderniste*. Ed. Nathalie Roelens and Wanda Strauven. Paris: Editions L'Harmattan, 2001. 113-140. Available online at <http://mdr.aletheia.be/crossings/interview.htm>.