FCJ-076 Continuous Materiality: Through a Hierarchy of Computational Codes

Kenneth J. Knoespel and Jichen Zhu, Georgia Institute of Technology.

Introduction

2 The legacy of Cartesian dualism inherent in linguistic theory deeply influences current views on the relation between natural language, computer code, and the physical world. However, the oversimplified distinction between mind and body falls short of capturing the complex interaction between the material and the immaterial. In this paper, we posit a hierarchy of codes to delineate a wide spectrum of continuous materiality. Our research suggests that diagrams in architecture provide a valuable analog for approaching computer code in emergent digital systems. After commenting on the ways that Cartesian dualism continues to haunt discussions of code, we turn our attention to diagrams and design morphology. Finally we notice the implications that a material understanding of code carries for further research on the relation between human cognition and digital code. Our discussion concludes by discussing several areas that we have projected for ongoing research.

3 In popular culture, computer code is usually given a role that transcends the physical world. In William Gibson's 1984 cyberpunk fiction Neuromancer (Gibson, 1984), the ultimate expression of layered codes is the omnipotent cyberspace, which represents an ethereal escape from the filthy, hopeless 'meat' world. It seems just a touch revealing that in Gibson's world where the body can be artificially modified at will, cyberspace can only be accessed through that which remains untainted – the mind.

4 Gibson's choice for the brain/mind as the ultimate interface to cyberspace is not accidental. It conforms to a long tradition of Cartesian dualism, whose origin may be traced back to Plato. In this tradition, metaphysical assumptions often separate the human capacity for language from the material world. Whether one considers Saussurean linguistics, or even Shannon's writing on information theory, much of the work that attempts to relate language to the world seems to accommodate the ghost of Cartesian dualism.

5 With the advent of the digital computer, Cartesian dualism has become challenged by computer code. On the one hand, computer code, relying on a complex network of the imperceptible electro-magnetic shifts, is generally regarded as immaterial. On the other hand, researchers in tangible computing demonstrate that digital media exist at the very boundary of the physical and the digital (Ullmer and Ishii, 2000). The unsettling relation between computer codes and materiality, therefore illustrates that code is by no means a self-contained language, but rather should be understood as a dynamic system intertwined with the material world.

6 Contrary to the legacy of Cartesian dualism inherent in popular accounts of language, the relation between natural language, computer code, and the physical world provides a basis for delineating what we call continuous materiality, that is, a wide spectrum of materiality activated by a hierarchy of codes that moves from 'lower' machine code to 'higher' readable computer languages and to codes in general (structural, legislative, social, cultural, etc.). Each level of code engages natural language and the physical world in different ways, varying from the shifting voltage of computer circuits to our everyday activity. Altogether, the hierarchy of codes constructs a field of diverse materiality that is continuous and interconnected.

The Legacy of Cartesian Dualism

7 For many, computer code belongs to the immaterial side of the Cartesian dichotomy. Unlike its hardware counterpart, or for that matter other physical objects, computer code is not bounded to perceivable time and space. For many, computer code only reveals itself in the momentary flickering of the screen. As implied in Cartesian dualism, it appears as if the immaterial dictates the material. Kittler observes a similar phenomenon in the very root of digital computers – the universal Turing machine, with its ability to imitate any other machine (Kittler, 2006). The abstract mathematical machine declares that 'the eventual differences between hardware implementations do not count anymore [and that] the so-called Church-Turing hypothesis in its strongest or physical form is tantamount to declaring nature itself a universal Turing machine'. However, this spirit versus matter dichotomy ignores the complex exchange between software and the material

world including hardware. Kittler's article draws attention to the hardware because the boundary of software is always affected by the limitation of hardware. By using examples of different types of code, we argue that code is inseparable from the material world and manifests its participation in what we refer to as continuous materiality.

8 **Computer Code and the Specter of Cartesian Dualism**

9 Traces of the Cartesian ghost can be easily identified in places where computer codes are used. Artificial intelligence (AI) research, especially in its heyday of the 1960s and 1970s, set its foundation on the Cartesian model. Often referred to by contemporary AI researchers as Good Old Fashion AI (GOFAI), this inferential paradigm of AI was built on the premise that thinking is essentially a process of immaterial symbol manipulation. For this reason, it was believed that intelligence could be simulated in computers. In effect, by drawing distinctions between an 'internal' mental process and 'external' activities in the world, the whole field of GOFAI rests on a combination of enlightenment rationalism and Cartesian dualism (Mateas, 2002; Agre, 1997; Brooks, 1991; Dreyfus, 1992). The enthusiastic attempts to teach the computer to play chess perfectly exemplify the emphasis on abstract symbol manipulation at the price of neglecting the surrounding physical world. Such an imbalance contributed to the failure of many AI projects and eventually the 'AI winter' in the late 1980s.

10 Computer code also brings romantic notions of immateriality to the aesthetics of digital art. Software art, for instance, uses codes as an expressive medium. Arguably, the code serves a more important role than the final images that it generates. Hence the aesthetics of the genre emphasizes the conceptual level of the piece more than the perceivable properties of the artifact. It is not accidental that many software art works are based on mathematically inspired algorithms. Their often geometric output lends itself to a math-like immaterial aesthetics of abstraction, precision and elegance. Software artists sometimes trace their heritage to conceptual art, another art movement that refuses a material-based aesthetics. The idea or concept and its (sometimes mental) execution are the central element of conceptual art pieces. For example, an influential conceptual art piece, La Monte Young's Compositions 1960, only consists of one line of instruction: 'draw a straight line and follow it'.

11 Fortunately, significant effort has been directed towards linking the physical and digital world. In the 1990s, a new paradigm of AI called the Interactionist AI was established by leading researchers such as Rodney Brooks. Compared to GOFAI, the new paradigm paid much closer attention to interacting with the physical world than to modeling and formalizing it symbolically. Meanwhile, Tangible Computer researchers demonstrated that the tangible user interface, and digital media in general, exist at the very boundary of the physical and the digital (Ullmer and Ishii, 2000). Essentially, the claim they make is very similar to Kittler's – namely that code can never be separated completely from the hardware/interface that makes it possible for humans to author, perceive and interact with code.

12 **Computer Code, Natural Language and Materiality**

13 Natural language has struggled with its relation to the physical world for a very long time. In her latest book My Mother Was a Computer: digital subjects and literary texts, N. Katherine Hayles criticizes the dominant western position found in the study of literary texts (Hayles, 2005). Again and again Platonic, antirealist positions assert themselves for they 'focus almost exclusively on linguistic codes, a focus that allows them to leave the document as a physical artifact out of consideration' (Hayles, 2005: 96). Text, as Hayles points out, is more than merely linguistic code. For example, in a website devoted to William Blake, the editors tried their best to preserve the physical characteristics of a text, including page size, typeface, margin, etc. because of their strong belief that materiality potentially affects meaning: 'The editors make canny use of the computer's simulation powers to render the screen display as much like the printed book as possible. They provide a calibration applet that lets users set screen resolution so the original page dimensions can be reproduced. They include a graphical help section that uses illustrations of pages to indicate the site's functionalities and capabilities' (Hayles, 2005: 90).

14 Besides being an interesting alternative to the immaterial reading of the text, the Blake Archive Project more importantly invites us to ponder whether our understanding of materiality is confined within a physical-based understanding of materiality. The lessons from literary study are relevant here. Hayles argues that our understanding of materiality, in the transition from print to

electronic text, needs to be expanded beyond mere physicality to accommodate the new praxis brought about by digital technologies. She proposes that '[t]he materiality of an embodied text is the interaction of its physical characteristics with its signifying strategies. Materiality thus marks a junction between physical reality and human intention' (2005: 103). The 'material' argument made by Hayles has implications that extend far beyond the recovery of materiality understood within textual criticism. It asks that just as literary scholars have given attention to the physical transmission of texts, they now are learning to give attention to their digital transmission as well. The interaction of codes and symbolic systems recognizes the role of semiotic theory but in a way that emphasizes the critical importance of movement across different codes. Together our work anticipates a new comparative study of codes and their integration.

15 Consideration of the materiality of computer code must be considered together with their connection to natural language. We would like to use AI as a departure point for this discussion. AI researcher and critic Agre pointed out that much of the public and philosophical debate regarding AI was surrounded by the seemingly fundamental question of 'can machines think?' (Agre, 1997). Yet, that said, the actual practice in the field was hardly dependent on the answer to such questions. In day to day research, what really matters is the effort to build computer systems whose operations can be narrated using intentional vocabularies, such as reasoning, planning, learning and strategizing. These key AI terms, however, are simultaneously vague and formal. The meaning of planning, for instance, when used to describe the behavior of a system, depends partially upon the practitioner's sense of the vernacular meaning of the word. On the other hand, it is only possible to ascribe the term to system behavior if the term is formally defined in regard to mathematical entities or computational structures and processes. It is precisely what we may think of as the 'hermeneutical circle of AI' that allows the AI narratives to be built, simultaneously revealing that computer code cannot exist without connection to natural language and to the materiality of the world.

16 A similar conclusion is reached by another AI researcher Michael Mateas (2002), who identifies two inseparable elements of any AI system – a code machine and a rhetorical machine. In his framework, the code machine of an AI system lends itself to the actual 'uninterpreted' computation and the complex causal flows, whereas the rhetorical machine provides both programmers and audiences discursive strategies to interpret the complex computation and definitions of progress within the system: 'The rhetorical strategies used to narrate the operation of an AI system vary depending on the technical approach, precisely because these interpretative strategies are inextricably part of the approach. Every system is doubled, consisting of both a computational and rhetorical machine. Doubled machines can be understood as the interaction of (at least) two sign systems, the sign system of the code, and a sign system used to interpret and talk about the code' (Mateas, 2002). With the discursive elements provided by the rhetorical machine, it becomes possible to attribute intelligence to an AI system. Both Agre's analysis of the ambiguous AI key terms and Mateas's double machine may be applied to computer codes in general. Even the simplest command to write a line on the screen contains such natural language vocabularies as 'print' which was and remains heavily associated with the experience of the physical world. In other words, computer code and its rhetorical context are heavily related to natural language and our embodied experience of the physical world.

17 So far, we have noted cases in which Cartesian dualism fails to capture the intricate interaction between the material and the immaterial. The complex relationship among natural language, artificial computer code and the physical world opens the door to the delineation of a continuous materiality, activated by a hierarchy of codes from 'lower' machine code to 'higher' level computer languages and to codes in general (legislative, structural, social, etc.) Each level of code engages natural language and the physical world in different ways, varying from the shifting voltage of computer circuits to our daily activity. When considered together, a hierarchy of codes constructs a field of diverse materiality that is continuous and interconnected. In the next section, we look at the use of diagrams in architecture to reveals the ways in which a hierarchy of codes is embedded in a spectrum of materiality.

Diagrams and the Hierarchy of Codes

18 Considering how much work has gone into the study of diagrams in architecture, the place of diagrams within architectural theory and practice still remains somewhat allusive. After all, what

relation do they have to sketches, plans, construction or building codes? Or is it not so much the single diagram but the linkages that they engender that mark the genealogical nature of diagrams? As visual containers of a hierarchy of codes, diagrams negotiate the space between the semiotic system and the physical world in a similar way to computer codes. Diagrams engage not simply a horizon of understanding but a terrain in which structures literally appear in the world. If we are to think about diagrams closely, we must do more than simply mark their presence. Instead we should register their cognitive significance as they direct work and establish networks of relationships between multiple symbolic fields. Diagrams are important, and indeed so much so, that rather than drifting within a hermeneutical setting they should be approached as vehicles for accessing a hierarchy of codes within a material setting. In effect, schools of architecture as well as computer science might become more recognized as laboratories for exploring material cognition and its bearing on the ways we approach technology. The question of diagrams in technology is important for as genealogical structures they can reveal the theoretical grammars and social codes used to enforce them. From a sociological perspective, diagrams might be thought of as comprising the circuit system of networks. The layered hierarchy of codes may also be demonstrated in the ways that diagrams are used within design morphology in architectural practice. Here diagrams are not fixed but transient moments in an emergent material practice.

19 **Diagrams and Cognition**

20 The ways in which diagrams manifest meaning is in ample evidence in the daily practice of architecture even though the graphic diagrammatic operations through which architecture is taught and thought are often pushed to the margins of its history. Our scientific conception of space just as our architectural formulation of space is thoroughly mediated by diagrams. A broad distinction can be made between ephemeral and professional applications. Doodling on a napkin is in a category separate from the diagrams of textbook traditions. Even here, however, it is not possible to make a rigorous distinction. Casual or formal, any diagram instantiates a set of codes – whether the Euclidean laws of geometry or fire codes regulating building structures. Diagrams hardly stand as isolated figures but are placed within a narrative setting. Along with the layers of code they incorporate, they become – or are intended to become – part of a structured argument. We may think of architecture as a process of building logical modalities that entail the representation of diagrammatic space. Diagrams thus offer the thinking space that connects to the physical world in various ways – either the earliest stages of design or, in the retrospective clarification of design aims that become crystallized at the later stages of design, or even after the completion of the building.

21 From the vantage point of phenomenology or cognitive science, diagrams have an optical foundation because they suggest the ways in which connections are made within a visual field. It is their optical foundation that also affirms their haptic role as recorders of operations such as drawing, tracing or plotting. But we may also identify a linguistic orientation in which the visual field is shaped from the vantage point of grammatical or lexical structures. Diagrams may mark a way to follow the body into language and even more a way to follow language into the spatial experience of the body. Historically, it is possible to relate the dissemination of the idea of diagram to optical geometry. It is quite appropriate to think of diagrams as being constituent features in the process of perception analyzed by Locke or as the instrumental figures that Hume describes in the evolution of the thought process (Locke, 1959).

22 Diagrams are phenomenological agents within the cognitive process and work as elemental mental constructions that enable us to hypothesize about the world. Multiple valences surround the word diagram or diagramma in Greek. The root verb of diagramma means not simply something which is marked out by lines, a figure, form, or plan, but also carries a connotation of marking or crossing out. In contemporary Greek the verb diagrapho, [noun diagraphe] means to write someone off. The verb may also be used to describe the movement of planets in the sense in which their movement may be thought of as ‘inscribed’ in the heavens or ‘reinscribed’ in subsequent orbits. In such a setting, the word diagramma suggests planetary trajectories that ‘write over’ themselves in each orbit. The word is also used to mark the transitory figures written on a wax tablet with a stylus. Here the word diagramma literally suggests that diagrams emerge from diagrams. The definition of diagram as well as its etymology is useful because it reminds us that diagrams are part of an evolving cognitive continuum.

23

Diagrams and Emergence

24 Diagrams point toward the technologies of emergence which can be enacted in the world. We need to think of technology as a continuous set of interactions with signs that become increasingly reified. The diagram is an important mark within the genealogies of sign systems. The point we would make is that technology should be regarded not as a jump from an idea to an artifact but as a complex process of increasingly complex sign systems. One way to enter this zone is to approach diagrams as vehicles that register a process of becoming.

25 Emergence is a concept that describes unexpected discoveries, emergent phenomena, or global behavior rising from the conjunction of local behavior or local conditions (Wilson and Frank, 1999: 964). Drawn from AI and biological theory, emergence describes non-deterministic, self-organizing phenomena that arise from local interaction between low-level units within a system. The design practice is less an isolated set of steps than a phenomenological assembly of code. When regarded together, such an assemblage compiles an emergent process (Poon and Maher, 1997). What is of utmost importance is that the ways emergence has been incorporated by creative design calls attention to the limitations of linear, formalist models of design. More precisely, the artificial application of the concept of emergence to architecture, reminds us that a connection between design cognition and the organization of living systems is absolutely crucial. Rather than abandoning an idea of emergence, we would like to argue that instead of modeling emergence through linear genetic algorithms, we seek its force through a theory of self-organization.

26 Diagrams are central to a theory of emergence. Our point is not the naïve argument that a single rational continuum of diagrams moves from idea to structure but that the relationship of imagination, shape-logic and building is one that is repeatedly negotiated through diagrams accompanied by speech. From a philosophical vantage point, there is not a single rational continuum (Descartes) but an infinite number of possible connections (Leibniz). Diagrams not only participate in building design but mark the regulation of building construction. But no matter how formulated they are, they can always be written over marking a moment of change. What is important is that diagrams don't work by themselves. They constitute what we may think of as diagrammatic genealogies that participate in the material construction of a building. They also constitute a time line which orders the ganglia of construction. What is also interesting is that the design process continues into the construction phase. The collection of diagrams then becomes not simply an archive but a genealogy of building construction. What once described where to place electrical outlets becomes a diagram that helps in locating an electrical outlet. Diagrams participate in the shift from construction to maintenance. This is not insignificant but represents diagram as vital component that stages moments in the construction process. Where the design stage authorizes multiple narrative tangents, the construction stage works to integrate diagrams into a common narrative. There are multiple practical levels in which diagrams participate in an emergent material practice.

27 **Diagrams and Computer Codes**

28 The iterative use of diagrams within architectural design and construction coincides with the iterative use of code in digital practice. Here diagrams reveal an instrumentality that permits them to move from being agents that negotiate space to be instruments that manipulate space. The capacity of diagrams both to work as heuristic vehicles in the process of design and to dictate how something is to be constructed reinforces the diagrammatic continuum within architecture theory and practice.

29 From the vantage point of semiotics, diagrams mark a locus where there is a continuous set of exchanges between signifier and signified – a bundling of systems of signification – and where the structure becomes a sign or referent in its own right and where its existence does not depend on the word as signifier. Diagrams establish networks of relationships between multiple symbolic fields. Computer code, particularly when it is approached within an evolving spectrum of code, also works to connect multiple symbolic fields. The comparison may be expanded. Both the diagram and code may be viewed as governed by rules. However both may be approached as vehicles for discovery and invention. Obviously, both diagrams and code are rule-bound. However, both may be used to break rules when they are placed in another context. But far more than simply the logical setting in which they are placed, we would like to emphasize that they both function as mediators to the material world. It is their role in material mediation that is worth

looking at more closely. Rather than rendering such a role invisible, we would ask what it means to call attention to materiality. As we noticed in our earlier reference to Hayles, we must ask questions that do more than call attention to a digital textuality.

30 Computer code as well as diagrams may be viewed as either rule-bound or rule-breaking. The disruption associated with each is hardly a mystical process but an act of projection or even translation. Both diagram and code mark moments of stability and disruption. Such disruption, however, does not mean the absence of rules but rather the movement to another stage of development. Within code practice, the discovery of a new syntactical order may mark such a development. In both the case of diagram and code, one is hardly engaged simply in the creation of a new order. Instead, the new order challenges a new set of translations.

Integration and the Hierarchy of Codes

31 Architecture provides a laboratory for integrating a hierarchy of codes into a continuous materiality. From the moment that a design begins, it undergoes a process through which codes, aesthetic, structural, civil and social are incorporated and materialized. Even a roughest sketch embodies the aesthetic codes that influence the spatial volumetric and visual design decisions made by the architect. As building continues to be further materialized as more detailed drafts, physical models, CAD drawings, construction drawings and eventually the physical structure, codes at all levels are mingled in a process that is material and continuous. Such continuous materiality covers a wide spectrum includes the modestly materialized concept, the 3D rendering of the structure in the AutoCAD software, and the fully constructed building. A hierarchy of codes manifests itself even through the most mundane object in the building. A power outlet on the wall, for example, speaks to many layers of code. The electrical voltage running through it and its connection to other electrical systems in the building are regulated by electrical, safety and regional codes. Additionally, its spatial location obeys social codes, and its material and color are chosen based on aesthetic and economic codes.

32 **Strategies for Approaching the Material Integration of Code**

33 Evolutionary design has constituted an important component in design research in the last decade. Its application comprises the use of various techniques of evolutionary computation or artificial intelligence to generate design solutions (Mitchell, 1996: 205). Overall, methodology consists in the use of algorithms to increase and optimize the design-solution space. The approach – based on what is known as the neo Darwinist model – combines ideas from genetic theory from Mendel and evolution theory from Darwin to explain processes of natural evolution (Frazer, 1995: 128). By using genetic algorithms and neural networks, evolutionary design integrates the idea of genetic coding with the definition of an artifact's structure (Holland, 1998: 58; Stiny, 1980). Shape grammar has been used to analyze and to describe designs, and to produce variations based on the same grammar (Stiny, 1994). Underlying the rules are transformations that permit one shape to be part of another.

34 But evolutionary design models seldom give attention to an evolving idea of code itself. Instead, these models identify an evolutionary process that relies on a metaphor of development. The work of Edelman postulates a far more literal application of neo-Darwinism to the evolution of the brain and finally to consciousness itself (Edelman, 2006). Edelman's work is important because it posits an important expectation. Even though the brain is not a computer, its electro-chemical complexity inevitably becomes analyzed through the application of heuristic codes. The biological and so-called artificial codes do not have a one to one identity for the important reason that one is infinitely more complex than the other. At the same time, however, the biological and artificial codes are conceived as belonging to a material spectrum that we have referred to as continuous materiality. Edelman's work may be mapped in relation to Varela and Maturana. In contrast to Edelman's biological empiricism that focuses attention on neurophysiology (with the assumption that such a neural-physiology has universal applicability) Varela and Maturana stress the importance of approaching cognition as distributed.

35 The theory of autopoiesis, proposed by Humberto Maturana and Francisco Varela in 1970, argues that a living system embodies a continuous process of self-organization and emergence (Maturana, 1980). According to Maturana and Varela, living systems are self-producing systems. In contrast to assumptions that viewed living systems as generators of something different from

themselves, autopoiesis approached systems as simultaneously producers and products. Since an autopoietic system is organized as a network of processes of production that ultimately produce the system itself, they could claim that cognition was intimately linked to biological phenomena. Acting as a network of processes, the autopoietic system bears two distinct consequences. In the first, organization is understood as a network of production that makes the system possible; in the second, a particular structure constitutes a distinguishable component in the topology of the network (Thacker, 2004). Overall, organization determines the identity of a system, whereas structure determines how its parts are physically articulated. Organization identifies a system and corresponds to its general configuration. Structure shows the way parts interconnect.

36 Varela and Maturana have provided ground for approaching the constitution of design through cognition that is distributed or socially situated rather than dissecting the condition of one artifact in order to seek its replication through genetic code or grammar manipulation. It is through such a process that we may locate the hierarchy of code and continuous materiality. The hierarchy of codes rendered visible through digital media has an ontological status that may accurately be described as continuous materiality. A theory of continuous materiality bears consequences for Cartesian dualism but also for our understanding of the ways in which 'writing' has become transformed and expanded within digital media to include new ontologies of building and making.

37 The biological model proposed by Edelman and Varela and Maturana carry important consequences for computer code. By stressing the impossibility of making a direct correspondence between the brain and the computer, they recognize the ways in which code is part of a material process. Code cannot be isolated but participates in a complex interaction with other codes that are both biological and technological. Finally, they propose a means for asking if the generation of code that is tested by the human community is not a fundamental element of human identity. Contrary to Cartesian traditions that would situate the impulse to code in metaphysics, they locate code in the evolution of material condition.

38 **Morphology of Computer Code**

39 Our previous discussion has shown some of the ways diagrams provide access to design morphology and also establishes a basis for asking whether we may also speak of a morphology of code. There is ample reason to consider a morphology of code. Such a morphology, however, must not assume or posit the presence of a single-master code. Instead there are multiple codes that continue to interact and alter each other. It would appear that narrative forms themselves are examples of the morphological change of codes. The argument made by Mark Turner and others regarding conceptual blending describes the experience of multiple interacting codes (Fauconnier and Turner, 2003). Such integration then becomes manifest in narrative forms (stories) and to their evolving forms. But here we must also recognize that we use hierarchy to describe simple relationships. Beyond the computer, the linearity indicated by hierarchy may be replaced by an idea of non-linear integration. Hierarchy pertains not to a rigid command-control model but to a layered understanding of codes that describes continuous interaction.

The challenge of continuous materiality

40 Maturana and Varela provided an important point of departure for approaching self-organizing systems. But Edelman and others have extended this work through neural-physiology that seeks to integrate codes with biological structures. Just as there is a material continuum between biological processes with the individual, there is a continuum that involves the human community. We are at a point where it is possible to remove mysticism and 'spookiness' from these relations as a default condition or challenged to comprehend that the impulse to the metaphysical is inherent in the spectrum of materiality and the interaction of codes.

41 As a biologically grounded viewpoint, continuous materiality attests to a material connection between our mental activity and the world in which we live. The material connection may be intuited but, at a fundamental level, it becomes comprehended through the codes that humans have created. While codes change – and may be said to be evolutionary themselves – they are the material manifestation of human self-reflexivity that must be understood within a community. Individual codes, of course, are continually projected and subjectively tested. Finally, however, it is the function of code to be shared and tested. It is not far fetched to think that the making and testing of code marks the way that we attest to our common humanity. The Kantian a priori of

space and time are not only posited as a ground for shared experience but manifest our interaction with code. Codes are patterns or sequences discovered or invented by humans to provide access to natural phenomena. They include logical codes but also include chemical chains or evolving interactions. The notion of continuous materiality posits a coherent, albeit complex, relation-connection between the human body, brain, and nature. Continuous materiality becomes understood as a continuum that includes mental experiences associated with consciousness.

42 The emphasis that we have given to the materiality of code and to the place of code in a material continuum allows us to make several comments about the status of digital artifacts. The integration of material code with design morphology offers multiple examples of the ways 'making and building' occur within settings of digital technology. By referring to 'building and making', we have in mind the ways in which working with digital technology can be accompanied by the sense that one has made or constructed something that has a presence that extends beyond its virtual presence on the computer screen. While such experiences may be associated with the manipulation of visual images on the computer screen, such experience by no means should be limited to vision. The broadly shared experience of 'building or making' something that is virtual rather than real – say of making a digital model of building rather than a physical model – has broad implications. Although the computer continues to be celebrated as a writing medium, there is a broad recognition that computer have inaugurated an era that is increasingly reliant on shaping visual information. The new iconicity enabled by computers may also be associated with a corresponding emphasize in orality. It would seem that the experience of 'making and building' within digital environments must be closely related to the emergence of new ontological experience. Digital media has surrounded us with new ontologies of building and making that require new conceptual phenomenologies. If we approach digital technology only through romantic notions of immateriality or of some ethereal half-life or isolated in between state removed or ever at a distance from the 'real world', we will continue to ascribe to a simplistic realism that bogs down in neo-Cartesian distinctions that create an illusion of separateness. We require a critical vocabulary that allows us to comprehend the ways in which we inhabit a continuum of codes. Inherent in the multiple codes through which we mediate our experience is an expectation that it will be possible to move quickly from one to the other. Indeed, it is increasingly the very objective of education to build settings where students may develop agility for code-switching.

43 The new ontologies of building and making bear an active rather than passive force. From the vantage point of hermeneutics, the very idea of interpretation becomes transformed from a passive response to an active engagement in making something in the space of a potentially unlimited number of screens. Rather than translating a code into another code for human reception, interpretation itself becomes an act of intervention. We may even wonder at the ways in which the passive hermeneutic analysis of late twentieth-century literary theory has shifted toward an active hermeneutic of digital technology. If we wish may even think of reader-response theory as a practice that emphasized the active importance of the reader in the constitution of the text. In the schema we are describing, the reader does not reassemble a text but instead works toward the constitution of objects. The ways digital media intensifies the experience of invention has been commented on by many theorists. For example, Barbara Stafford has argued with precision that the active force inherent in the ways we work with digital media requires a significant adjustment in the ways we integrate word and image. More specifically she has argued most recently for a more articulate cognitive practice to account for the ways in which we negotiate meaning through visualization (Stafford, 2007). Stafford's argument resonates with the shift toward materiality observed in N. Katherine Hayles's argument noted above. Both observe that the texture of discourse has changed in a material sense that is profound. For both Stafford and Hayles text no longer carries the assumption of something written in natural language or presented through iconic tradition.

Conclusion

44 We have used the term continuous materiality as a means exploring the materiality of computer code. We have done so because it reminds us that computer code is never isolated but always interacts with other codes. Considered together such codes – a hierarchy of codes – constitute an evolutionary spectrum. We have also noticed that a hierarchy of codes may be usefully compared

to diagrams. In addition, we have suggested that just as diagrams provide a valuable means for approaching design morphology in architecture, they offer a means of posing questions about a morphology of code. Finally, we have observed that a morphology of code may be situated in the biological and neural-physiological work of Edelman, Varela and Maturana. In a paper where much has been suggested about evolving processes within a material continuum, it is appropriate to conclude by recognize the degree to which this paper itself manifests itself a continuum of ongoing development. An important way to attest to the evolving nature of our thinking may be to recognize several questions that continue to occupy our own evolving work.

45 Of course, the entire question of the ontology of digital objects can be considered more extensively. What is striking is the way in which such research could draw on an evolving understanding of the technological process. Rather than making a radical distinction between making virtual digital artifacts and so-called real objects in the world, it is reasonable to approach each within a process of design. Once again architecture provides an important laboratory for such research. The development of multiple generations of CAD assisted design has registered stages in the design process that otherwise might be regarded as part of inspiration. Let us be very clear. While inspiration may well exist a material understanding of design morphology may help in defining more precise how the designer works within such an evolving process of design. Inspiration may be an abbreviated way of describing the complex interaction of material codes.

46 The omnipresence of digital technology challenges us not to mystify technology or ideas of code. On the contrary, the emergence of these new technologies over the past decades must not be met with ignorance of code and its essential presence. Rather than isolating code and rendering it invisible, computer code should be approached as part of an evolving network of interrelated codes. Such engagement, involving the active exploration of the relation between neurophysiology and code is facilitated by an idea of continuous materiality.

Acknowledgments

The term ‘continuous materiality’ was suggested by discussion with our colleague Sha Xin Wei (Concordia University). We also recognize the contributions of our colleague Eduardo Lyon from the College of Architecture at Georgia Institute of Technology to the discussion of architecture and design. An earlier paper by K. Knoespel and E. Lyon provided valuable orientation for this paper.

Authors’ Biographies

Kenneth J. Knoespel is McEver Professor of Engineering and the Liberal Arts at Georgia Tech and Chair of the School of Literature, Communication. In addition to recent work on cognition and visual practice in mathematics and architecture, he has worked on changing visual practices of interpretation within the natural and human sciences. He recently edited a collection of essays on Diagrams and the Anthropology of Space. In addition to his work for the Graduate Program in Digital Media at Georgia Tech, he has regularly taught a graduate seminar in the College of Architecture with his colleague John Peponis devoted to “The Spatial Construction of Meaning.” Work from their seminar has been presented in Greece, Italy, France, England, and Denmark. In Sweden and Russia, he has worked closely with the University of Uppsala, The Royal Institute of Technology, Chalmers Institute of Technology, Blekinge University, the European University of St. Petersburg, and the Russian Academy of Sciences. He is currently working on a project concerned with cities and landscape of the Baltic Sea.

Jichen Zhu is a PhD student in the Digital Media program in School of Literature, Communication, and Culture at Georgia Tech. She is a member of the Imagination, Computation, and Expression (ICE) Lab, with a focus on Expressive Artificial Intelligence. Jichen is interested in understanding computational technologies, AI in particular, from a social and cultural perspective, and exploring the expressive and subjective use of digital media. She holds a Master’s degree in Entertainment Technology from Carnegie Mellon University, and a B.S. in Architecture from McGill University, Canada.

References

Agre, Philip E. ‘Toward a Critical Technical Practice: Lessons learned in trying to reform AI’, in Geof Bowker, Les Gasser, Leigh Star, and Bill Turner (eds) Bridging the Great Divide: Social Science, Technical Systems, and Cooperative Work (Erlbaum, 1997).

Brooks, Rodney. ‘Intelligence without Representation’, Artificial Intelligence Volume 47 (1991): 139–159.

Dreyfus, Hubert L. What Computers Still Can’t Do: A Critique of Artificial Reason (Cambridge: MIT Press, 1992).

Edelman, G. M. Second Nature (New Haven: Yale University Press, 2006).

Gibson, William. Neuromancer (New York: ACE, 1984).

Fauconnier, G. and M. Turner. The Way We Think: Conceptual Blending and the Mind’s Hidden Complexities (New York: Basic Books, 2003).

Frazer, John. An Evolutionary Architecture. Themes vii (London: AA Publications, 1995).

Hayles, N. Katherine. My Mother Was a Computer: digital subjects and literary texts (Chicago: University of Chicago Press, 2005).

Holland, John. Emergence : from chaos to order (Reading: Helix books, Addison-Wesley, 1998).

Kittler, Friedrich. ‘There is No Software’, CTheory.net (1995), https://www.ctheory.net/articles.aspx?id=74.

Locke, John. Essay concerning Human Understanding Vol. I (New York: Dover Publications, 1959).

Mateas, Michael. ‘Interactive Drama, Art, and Artificial Intelligence’, Ph.D. dissertation (Carnegie Mellon University, 2002).

Maturana , H. Autopoiesis and Cognition (Dordrecht: Reidel, 1980).

Mitchell, Melanie. An introduction to genetic algorithms: complex adaptive systems (Cambridge: MIT Press, 1996).

Poon, J. and M. L. Maher. ‘Co-evolution and Emergence in Design’, Artificial Intelligence in Engineering, 11:3 (1997): 319-327.

Stafford, Barbara M. Echo Objects: the Cognitive Work of Images (Chicago: University of Chicago Press, 2007).

Stiny, G. ‘Introduction to shape and shape grammar’, Enviroment and Planning B, 7 (1980): 343-351.

Stiny, G. ‘Shape rules: closure. continuity and emergence’, Enviroment and Planning B, 21 (1994): 1-29.

Thacker, Eugene. Biomedia (Minnesota: University of Minnesota Press, 2004).

Ullmer, Brygg and Hiroshi Ishii. ‘Emerging Frameworks for Tangible User Interfaces’, IBM Systems Journal, Volume 39, Issue 3 (2000).

Wilson, R.K. and C. Frank. The MIT encyclopedia of the cognitive sciences (Cambridge, MA: MIT Press, 1999).

The Fibreculture Journal (ISSN 1449-1443) — Issue 11, 2008-11-12. This article is published under CC BY-NC-ND. The author retains copyright.

This PDF is an archival offprint generated from the journal’s text archive. It is not the publisher’s version of record; the published article is at https://eleven.fibreculturejournal.org/fcj-076-continuous-materiality-through-a-hierarchy-of-computational-codes/. Page numbers here are not the published pagination. Where paragraph numbers appear in the margin they are the journal’s own and are the citable unit.