Software aesthetics: from text and diagrams to interactive spaces

Paul A. Fishwick

Computer and Information Sciences Department,
University of Florida,
PO Box 116120,
Gainesville, FL 32611, USA
E-mail: fishwick@cise.ufl.edu

Abstract: Ubiquitous computing suggests that the computers and their information are located *everywhere* – inside walls, rooms, people and trees. A complementary scenario occurs along the virtuality continuum, where scripts and data are located separately in virtual 3D objects. A natural consequence of ubiquitous computing in physical and virtual environments is that information and software begin to transition from ethereal artefact to *design object*. Currently, this trend is being seen in ambient devices that display information. We describe how this trend will also lead to other forms of software being similarly expressed in a variety of human-interactive forms. The end result is a new vision for software representation that brings it into the public sphere.

Keywords: aesthetic computing; information aesthetics; information visualisation; software aesthetics.

Reference to this paper should be made as follows: Fishwick, P. A. (xxxx) 'Software aesthetics: from text and diagrams to interactive spaces', *Int. J. Arts and Technology*, Vol. x, No. x, pp.xx–xx.

Biographical note: Paul Fishwick is Professor of Computer and Information Science and Engineering and obtained his PhD from the University of Pennsylvania in 1986. He is active in modelling methods for computer simulation and software engineering and in the intersections among the arts, public relations and computer science. He is a Fellow of the Society for Modelling and Simulation (SCS), and recently edited two books: *Aesthetic Computing* (2006) and the *CRC Handbook on Dynamic Systems Modelling* (2007).

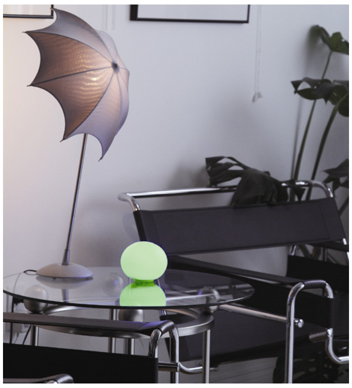
1 Introduction

Consider the two images in Figure 1. The image on the left is a diagram that represents the Dow Jones Industrial (i.e. Symbol: DJI) average over time from left to right, as seen in Google Finance (2008). The one on the right is a glass sphere (i.e. orb) that implements an FM radio transceiver sold by Ambient Devices, Inc (2008). One might say that the orb is an example of calm technology, and while this is accurate, the differences in representation between the objects shown in each image indicate more significant changes for the evolution of how software is perceived, and portend new ways in which we can interact with software.

Figure 1 Two methods of software interaction for representing a time series (see online version for colours)

(a) Dow Jones Industrial Average

Source: Google Finance (2008).



(b) Orb

Source: Ambient Devices, Inc (2008).

Table 1 A list of differences between Figure 1(a) and (b)

Quality	*Diagram*	*Orb*
Abstraction: the level of detail in terms of how much raw information is available	*Detailed*: there are interactive means to see events corresponding to points on the trajectory as well as to zoom along the time-axis	*Abstract*: the only information is the relative change in the DJI as mapped from red (-2.5%) to green (+2.5%)
Interaction: the manifold ways in which to sense the object	Flat (2D), virtual, interaction via a mouse, no immersion	Physical, tactile interaction with immersion
Culture: how we personalise or categorise the object within an individual or social framework	Culture of print media as applied to digital displays. Social interaction possible with the right display (i.e. large scale projection)	Relatively new type of physical/tangible computing culture. Social interaction facilitated due to its physicality

There are some obvious differences in the sorts of information possible (quantitative vs. qualitative) and levels of abstraction (detailed vs. simplified), and so let us create a table to get a handle on the qualities. Table 1 includes a set of qualities and explains how each representation addresses these qualities.

Each object is designed for different purposes as indicated in the table: the graph is for detailed analysis of the stock index and the orb serves as a gentle reminder of how the index is performing. The orb does not provide the same level of detail; however, it is possible to have more than one way to interact with information. The orb represents more than calm computing, since while one can see the orb in the periphery, it is also possible to sit and watch it to contemplate on its form as one would an artwork or an exhibit within a science museum. These observations lead to the following trends:

- *Towards a new form of aesthetics*. In the era of advanced methods in Human–Computer Interaction (HCI) technology, a new form of aesthetics is defined as a multi-dimensional HCI space. Thus, objects such as the orb are interesting primarily because they explore new partitions of that space, permitting us to interact with software in novel ways, adding to the more traditional and useful diagrammatic methods. An appeal to aesthetics is positioned as a commitment to creatively explore alternate methods and ways to represent software by navigating the space of interaction possibility.
- *From information to software*. We use the term *information* using common parlance of associating information with data even though the original theory-based meaning (Cover and Thomas, 2006) is associated with entropy and bit-oriented streams that can be interpreted as *data* or *computer program*. We define the term software as *not hardware* – a conceptual structure encoded in mathematically-based bits rather than physically-based atoms. Using these definitions, data, information and process are all types of software. Imagine culturally meaningful processes, such as how tsunamis propagate over the ocean floor or how obesity manifests itself in children, where each process is displayed or heard in new ways, encouraging new forms of interaction and contemplation. These processes can benefit from alternative forms of interaction and representation in the same way being achieved currently for information.
- *Incorporating multiple representations*. It is possible to build interaction devices that show us multiple facets of software so that one form need not replace the other (i.e. the orb complements the time series diagram). Even though the existing orb does not provide a seamless way to transition to the diagram using a unified, integral interaction space, other technologies could provide such transitions. Consider a single interaction space, possibly with a mixed reality approach, that allowed a human to see an orb or see the diagram overlaid and contextualised. We can see and interact with both representations, and potentially many more.

We will explore each of these three trends in turn, and then discuss related work that grounds the discussion. A multidimensional space contains dimensions labelled by variables that may stretch from style, personalisation, culture, ergonomics, and abstraction-level to technology forms such as animation, mixed reality, ubiquitous computing, physical computing, and shared networking environments. Aesthetics can be defined in terms of genre, style or a perspective, often within a cultural or historical context. It is a difficult word to define, and because of this the term can be thought of as denoting *variety* – the multitude of ways in which interaction is made possible, whether through technological advancement or through varying cultural settings. The orb captures this alternative form of interaction by exploring different degrees of *attention* with regard to information. For software representation, we favour an approach that steers away from defining a singular aesthetic. The variety of aesthetics is seen in Kelly (1998). One could create sharp divisions between areas and purposes by considering that 'this is art but that is science' or 'this is design but that is art'; however, this territorial map-making procedure is counterproductive at a time when computer science and the arts have so much to contribute to each other. To be sure, there are still soft boundaries due to separate cultures, but art and design tend to speak to the masses – the general public – whereas software has traditionally been difficult to decipher and meant for experts. To the extent that software can be argued to be publicly relevant, art and design can play key roles while simultaneously defining a new aesthetic.

The advance of database technologies and the marriage of the internet and hypermedia in the form of the web have promoted the relevance of information to the public sphere. Everyone understands that information is important and relevant: we want to know the value of a stock, see the dynamics of interpersonal relations extracted from blogs, or follow the latest statistics on the presidential candidates positions. With the web greatly expanding the amount and quality of information available, it is only natural that we seek new ways to interact with it. So, for example, information visualisation (Schneiderman, 1996; Card, Mackinlay and Shneiderman, 1999; Ware, 2004) was born as a field where one studies how to employ various visual tools and techniques to make sense of, especially, very large data sets. What the web has done for information, it can also do for software: expressions, processes, procedures, scripts and flows can be presented in entirely new ways for public consumption. The process-side of software is as relevant as information to the public, since software undergirds dynamic simulation models of natural phenomena, social processes and engineered artefacts. Knowing how ethanol production is made possible becomes just as relevant as raw statistical data on its utility, and knowing how levees break from water pressure is as relevant as viewing geographic maps of flooded zones. Software is about representing *how* and *why* (i.e. process), along with *what* and *where* (i.e. information). The general question of *when* tends to fall in-between process and information. Treating software as an aesthetic target provides new potential for creating public installations and exhibitions, paving the way for new forms of interaction. To use the orb as a starting point for representing more complex forms of software, consider that three orbs in a particular configuration could represent three states in a state machine.

Each representation of information in Figure 1 has a different goal or purpose. While, it is natural to envision design as a field where the concept of *goal* drives the design process, we are moving toward new forms of representation that may contain multiple forms of interaction and representation, and so there is not one purpose or goal – there are many goals for the same target. The glass orb is in a different location than the stock index trajectory, but one can imagine a situation where one sees the orb using a mixed reality display and then through some form of proximity-sensing or simple human interaction, the trajectory appears over the orb. We need not design interfaces that service singular purposes when software is the target representation. It is both possible and desirable to allow multiple views. In this way, software may look like an artwork, when being contemplated, but then gradually yields to provide more detailed mappings and knowledge upon request, or through change in human attention. Bolter and Gromala (2003) speak to this duality between reflection – with attention to the media – vs. transparency, where one seeks direct information from a product.

2 State of the art

Baumgarten first used the term aesthetics (Eagleton, 1990) after the Greek *aisthesis*. The goal of his thesis was to build a science of how things are to be known by means of the senses (Guyer, 2004). Oddly enough, this basic definition is only slightly redefined in terms of computing technology and its ability to reshape how humans can relate to objects and materials that are imbued with computational affordances. One might think of the new aesthetics as a science of how things are to be known through HCI.

A dynamic application of software aesthetics, in the form of information display, is present in the mechanical *clock*, which began its history using water and gravity and progressed several centuries ago into the more common mechanical form with a gear train and escapement. The clock displays a concept, a piece of data: time. It could just as well represent any form of cyclical data, but we needed to wait for the age of computing to enable routed information to be sent to arbitary devices. The interesting aspect about clocks is their significant diversity in representation, suggesting a corresponding degree of diversity for information and software. This requirement for diversity stems from research in creativity where the 'generation of possible solutions' are explored (Edmonds and Candy, 2002).

The first instance of representing information is tied to the use of language where *symbols* are used. The symbol is a physical artefact that, in most instances, bears no representational relationship to its target (Nöth, 1990). However, symbols have representational meaning as well as fostering the aesthetic experience. For example, Knuth's development of T_EX (Knuth, 1979, 2001) points beyond the idea of aesthetics in computer science being strictly about cognitive beauty, the ideal of the minimalist algorithm or an elegant data structure (Oram and Wilson, 2007): software is also about *look* and *feel*. It is remarkable that just one of the narrow dimensions of the large-scale multidimensional interaction space (e.g. typography, by varying the style of 2D linear symbol strings) can be explored with such diversity. For example, information presented in the Fraktur font may be viewed, and contemplated, differently than information typeset in Helvetica.

Weiser's (1993) ubiquitous computing approach, and broader collaboration (Weiser and Brown (1996)), elaborate on a future where not only are computational elements in every surface and object, but also that there are different ways of *interacting with information*. They cite artist Natalie Jeremijenko's 'Dangling String' which served as a representation of Ethernet network traffic. At the same research location at Xerox Palo Alto Research Center (PARC), a fountain was created in a hallway to indicate the rising and falling of the Xerox stock price.

Ishii and Ullmer (1997) at the MIT Media Lab continue to maintain a significant project in tangible interfaces – the idea that a blending of bits and atoms is being explored. The research is based on interacting with surroundings (i.e. objects and building architecture) as if the surroundings – infused by computation – were natural parts of the human experience (Wisneski et al., 1998). Schmitt (1999) and Engeli (2001) and colleagues at ETH Zurich take the topic of architecture and blend this physical space with 'bit space' thus making a fifth dimension of computer aided architectural design.

Something interesting is happening here with these projects – it is not only that the presentation is calm or peripheral; information is breaking out of the metal box and aesthetic variety evolves. A natural and causality related, effect of this technology is that information bubbles up from within the silicon substrate toward the human with new interaction possibilities. In turn, since the information is made publicly tangible, the information becomes 'product like' and so is treated as any other designed product such as an automobile or house – with a required balance of form and function. The interaction of art, design and computing lies at the heart of these innovations, and so Malina's strong claim for aesthetic computing (Malina, 2006) seems to have firm ground and is well underway.

Manovich (1999) draws off the symbolism naturally inherent within Avant Garde media early in the 20th century to claim that many of today's information interaction can trace lineage to that period. The application of aesthetics to HCI, which has tended to focus on visual interfaces and information is still in its infancy; however, there are a number of HCI researchers who are exploring the philosophical ramifications as well as empirical studies of information, aesthetics and HCI (Redström, Skog and Hallnäs, 2000; Hallnäs and Redström, 2001). Hallnäs and Redström refer to the new confluence for interacting with information as *amplified environments*, which further emphasises the potential connection between art and information. Udsen and Jørgensen (2005) provide an overview of the recent trend to provide more effective integration of the broader view of aesthetics within HCI, connecting research efforts between diverse fields. Vogel and Balakrishnan (2004) create a *multi representational* ambient display that captures four interaction phases from implicit to explicit. In their work, the display is in the periphery when the participant is distant and then as the participant nears the display, the display changes to afford further interaction. This type of multiple representation within the context of simulation models was posed as a simulation *grand challenge* (Fishwick, 2004) using the term *integrative multimodeling* (Park and Fishwick, 2005) and recently, incorporated into a mixed reality display using a magic lens made from a tablet PC (Quarles et al., 2008).

Löwgren and Stolterman (2004) describe how aesthetics can be positioned with HCI through a revised set of interaction qualities. Bertelsen and Pold (2004) specify how criticism aids in assessing experience, while Tractinsky (1997) and Tractinsky, Katz and Ikar (2000) has created human experiments to test the effectiveness of *aesthetics as usability*. Mankoff et al. (2003) evaluate the effectiveness of ambient displays. Results point to 'aesthetic and pleasing design' as an important heuristic.

If information is a fundamental component of software – data being stored in locations, routed and filtered in others – one imagines that the focus in the aesthetics of information will naturally progress toward the aesthetics of expressions, functional forms, processes and programs. Fishwick (2002) presents a prototype for treating programs as aesthetic targets. This approach, as well as prior activities between artists and scientists, prompted the formation of the Dagstuhl Conference in Aesthetic Computing (Bertelsen and Fishwick, 2002), an empirical study (Fishwick, Davis and Douglas, 2005), and the subsequent edited volume (Fishwick, 2006). Aesthetic computing has a broader agenda than for software aesthetics, since the goal is to consider all aspects of aesthetics –

including the traditional cognitive variety found in mathematics and computer science – to the field of computing.

Lau and Vande Moere (2007) provide a model for information aesthetics based on *Mapping Technique* and *Data Focus*. They also point out a critical cross-cultural issue where art may see this area as too functional, while science tends to over-stress function to the exclusion of a broader aesthetic. This is going to be a significant gap to cross especially when considering the extension of information more generally to software; however, as indicated, there has been significant historical evidence for the emergence of a new aesthetic form. Vande Moere's Blog *Infosthetics* (Vande Moere, 2008) contains a wealth of examples stretching from functional diagrams to innovative, aesthetic interactive installations.

3 Software and the machine

The relationship between software and the hardware (i.e. machine) is an interesting one from a HCI perspective. For software in the form of information, humans throughout the day are presented with many forms: alarm clock, watch, automobile displays, kitchen appliances and media centers. Therefore, objects like the glass orb are not too surprising, since we are already used to see textual information from these objects. One wonders, though, about human interaction with something more complex than information-process. First, we must look at a brief history of process as a type of software.

Software is defined as the area that deals with encoding and manipulation of bits rather than atoms, and so, data are the primary elements of software. We then progress to data structures and a variety of program structures, including functional expressions, programs and dynamic (i.e. process) models (Fishwick, 1995). Leaping from information to software suggests that we also consider representing *how* information changes over time. The root of structures necessary to achieve these representations can be found in automata and systems theory. The Turing machine, for example, represents information (i.e. state, input and external storage) and ways to modify that information (i.e. transition function). Numerous programming languages and software engineering paradigms have been created to create formal representations as text or in diagrams (Diehl, 2001; Zhang, 2003).

Software must make itself known to the human in terms of special purpose editors and interfaces. Also, since software is based on the concept of state, it is something sensed by the human, since the system output is a function of state. To select a biological analogy, the gene is the information and the body – which represents gene expression – directly reflects that information to considerable visible and interactive effect. Even though one may not see visible *process*, information is readily acquired through human interaction with the hardware. So, this leaves the question of how do we see process, scripts and programs? Even though programming would appear to be out of reach of the average person, most people are already programmers. People program their digital video recorders, washing machines, dishwashers and digital watches. They do not think of this as programming, and yet a human must have a mental model (Gentner and Stevens, 1983) of essentially a finite state machine (i.e. a simplified version of a turing machine). They know that when a certain button is pressed, the machine will return to a prior state. Therefore, our research in software aesthetics extends information aesthetics by surfacing this mental model through human interaction.

4 Aesthetic software

Figure 2 shows a flow graph model, drawn as a diagram using the system dynamics (Forrester, 1994) modelling methodology. This model is one example of *software as process*. System dynamics is an engineering approach to modelling involving various phases:

1 construct a causal graph showing nodal concepts, relations and feedback loops

2 construct a flow graph showing rates, levels and auxiliary variables

3 translate the flow graph into ordinary differential equations

4 run software that parses the equations into executable code.

The model in Figure 2 defines a situation where three state variables: fitness level, exercise and nutrition affect metabolic rate, which in turn flows into fitness level, which affects the rate of weight loss. Independently of this reaction, the amount of food eaten effects weight, which is reduced as weight loss measures are taken. This computer program is publicly relevant since obesity is one of the key issues found in the news. And so, reifying the process of *how* obesity occurs is something that will interest the public. As for information aesthetics, one goal is to create alternative interfaces to such a model.

To achieve this reification, we designed a 3D immersive scenario as depicted in Figure 3. The software that defines the model in Figure 2 has been recast as a virtual model with a virtual human figure that can interact with the participant. The nature of this design is based on the idea that by creating a scene that is immersive, with a particular visual aesthetic based on analog machinery, the design will attract the attention of the participant to explore further. In the figure, one can see water coming out of the floor and into the containers. Each water source has valves (i.e. rates) and each state is a glass container filled with water where the water height represents the current state value. Simulated mechanical connections encode the causality links in Figure 2.

Figure 2 A system dynamics flow graph for obesity, capturing the cause/effect dynamics

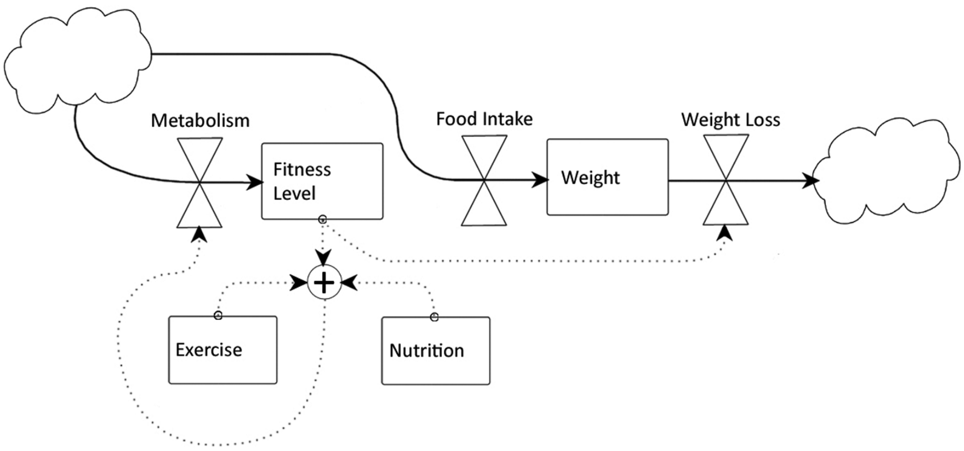
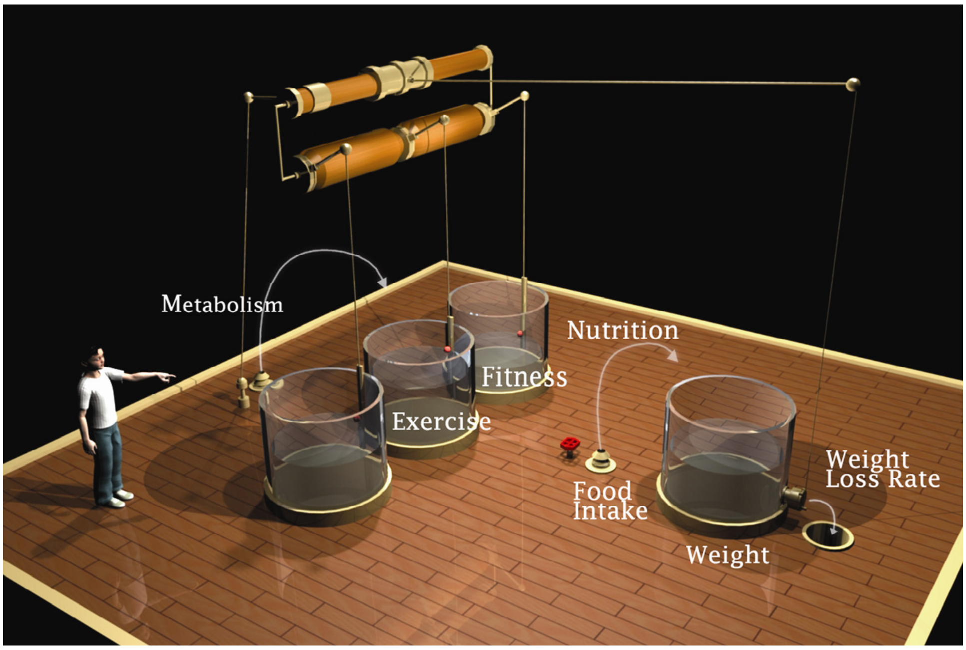


Figure 3 An immersive scenario containing a virtual child and the obesity 'software machine' (see online version for colours)



There is interaction between the human child and the system dynamics machine: the child tells of a story of how he obtained the machine from a doctor who explains that the machine will aid in helping him control his weight by learning more about *how* weight change occurs. As certain valves are set there are changes in the machine's state that may cause the child to say 'I am not feeling well' or 'I do not have enough energy'. These interactive empathic elements of the interaction increase the sense of presence and make a tighter connection between the artefact, child and participant. The software is no longer an abstract element of computing hardware. It has been elevated to a design object with which we form new forms of experience. This project is still in the design phase; however, we are exploring two implementation options with a science museum and a children's hospital.

5 Summary: three steps toward software aesthetics

We have identified aesthetics as being an integral part of society, our interaction with hardware and of software representation. There are a number of key challenges that need to be considered if one is to believe why software aesthetics should exist or be fostered. We will summarise as follows:

- *From ubiquitous computing to aesthetics*. The early work at Xerox PARC, MIT Media Lab and ETH Zurich in new forms of information interaction result in the philosophy centred on merging bits with atoms. This once futuristic trend is beginning to mature, since wiring and integrated circuits are becoming smaller, more powerful and physically flexible (i.e. polymer-based organic circuits). When computers spread out, there is a natural causality from this distribution to the blending of bits and atoms. This pushes the human interaction in a direction away from the traditional flat display. Moving away from flat displays *pushes information interaction design toward product design*. All product designs have strong aesthetic foundations, and a requirement to balance form and function. Conclusion: increasing ubiquitous computing results in a corresponding concern for product design aesthetics, with product = information.

- *Spanning the virtuality continuum*. The virtuality continuum (Milgram et al., 1994) stretches from the purely digital (i.e. virtual reality) to the purely physical (i.e. reality) with areas such as augmented and mixed reality situated on the continuum. This means that the concepts behind ubiquitous computing can be found along this continuum as well. Multi-user environments have scripts inside of virtual objects – just as hardware inside of physical objects in ubiquitous computing. Events cause the objects to respond. Conclusion: ubiquitous computing concepts are present along the entire virtuality continuum – applicable equally as well to artificial and physical spaces.
- *From information to software*. Like information, software in general is relevant to the public sphere. People wish to know *how* things occur in addition to know *what* is occurring. Therefore, approaches used in information aesthetics can lead to corresponding changes in how we view software. Humans build mental models of data (i.e. menu hierarchies present in television set-top media boxes) and programs (i.e. state transitions inherent within home appliances) and, so, are already familiar with software in its pure form even if they are not familiar with the theoretical foundation of trees, graphs and automata. Conclusion: Software defined as data, information and process can be represented within the virtuality continuum, thus encouraging public-software interaction.

The goal of information and software aesthetics is not simply to create casual, ambient, peripheral or otherwise fun objects that serve as carriers of information. Instead, the goal is to *explore the full range* of interaction potential between software in the periphery and in-depth software analysis. An interactive installation for the obesity example described in the previous section would be designed to allow the participant to interact with several diagrammatic forms, the underlying differential equations in text, as well as the 3D environment. The goal is not to supplant one design over another, but to incorporate multiple complementary representations.

There is a significant gap in culture between artists and scientists; however, the approaches to creating new interactive installations of information show a new confluence of ideas connecting computer scientists, designers and artists. It is possible to create multiple representations for the same artefact: art for the casual or contemplative participant and science for the analytic. It is necessary and advantageous to take software and bring both its syntactic structure, as well as its behavior, into the public sphere.

Acknowledgements

The author would like to thank his Zach Ezzell for taking rough and sketchy designs and manifesting them as beautiful artefacts for the obesity model. He would also like to thank all past and present students taking the Aesthetic Computing class since its inception in 2001, as well as to Benjamin Lok for discussions on the role of virtual humans within the context of aesthetic computing.

References

Ambient Devices, Inc. (2008) Available at: http://www.ambientdevices.com. Image source Available at: http://www.ambientdevices.com/cat/gallery.html.

Bertelsen, O.W. and Fishwick, P. (2002) 'Aesthetic computing, Dagstuhl seminar 02291', *Technical Report*, Dagstuhl.

Bertelsen, O.W. and Pold, S. (2004) 'Criticism as an approach to interface aesthetics', NordiCHI '04: Paper presented in the Proceedings of the *Third Nordic Conference on Human–Computer Interaction* (pp.23–32). New York, NY: ACM.

Bolter, J. and Gromala, D. (2003) *Windows and Mirrors: Interaction Design, Digital Art, and the Myth of Transparency*. Cambridge, MA: MIT Press.

Card, S., Mackinlay, J. and Shneiderman, B. (1999) *Readings in Information Visualization: Using Vision to Think*. San Francisco, CA: Morgan Kaufmann Publishers Inc.

Cover, T. and Thomas, J. (2006) *Elements of Information Theory*. New York, NY: Wiley-Interscience.

Diehl, S. (2001) *Software Visualization. LNCS 2269*.

Eagleton, T. (1990) *The Ideology of the Aesthetic*. Oxford, UK: Blackwell Publishers.

Edmonds, E. and Candy, L. (2002) 'Creativity, art practice, and knowledge', *Communications of the ACM*, Vol. 45, pp.91–95.

Engeli, M. (2001) *Bits and Spaces.* Basel, Switzerland: Birkhauser.

Fishwick, P. (1995) *Simulation Model Design and Execution: Building Digital Worlds*. Englewood Cliffs, NJ: Prentice Hall.

Fishwick, P. (2002) 'Aesthetic programming: crafting personalized software', *Leonardo*, Vol. 35, pp.383–390.

Fishwick, P. (2004) 'Toward an integrative multimodeling interface: a human–computer interface approach to interrelating model structures', *Simulation*, Vol. 80, p.421.

Fishwick, P. (2006) *Aesthetic Computing.* Cambridge, MA: MIT Press.

Fishwick, P., Davis, T. and Douglas, J. (2005) 'Model representation with aesthetic computing: method and empirical study', *ACM Transactions on Modeling and Computer Simulation (TOMACS)*, Vol. 15, pp.254–279.

Forrester, J. (1994) 'System dynamics, systems thinking, and soft OR', *System Dynamics Review*, Vol. 10, pp.245–256.

Gentner, D. and Stevens, A. (1983) *Mental Models*, Lawrence. Hillsdale, NJ: Erlbaum Associates.

Google Finance (2008) Available at: http://finance.google.com. Image source available at: http://finance.google.com/finance?q=DJI.

Guyer, P. (2004) 'The origins of modern aesthetics: 1711–35', in I. Kivy (Ed.), *The Blackwell Guide to Aesthetics*. Oxford, UK: Blackwell Publishing.

Hallnäs, L. and Redström, J. (2001) 'Slow technology-designing for reflection', *Personal and Ubiquitous Computing*, Vol. 5, pp.201–212.

Ishii, H. and Ullmer, B. (1997) 'Tangible bits: towards seamless interfaces between people, bits and atoms', Paper presented in the Proceedings of the *SIGCHI Conference on Human Factors in Computing Systems*, pp.234–241.

Kelly, M. (1998) *Encyclopedia of Aesthetics*. Oxford, UK: Oxford University Press.

Knuth, D. (1979) *TEX and METAFONT: New Directions in Typesetting*. Boston, MA: American Mathematical Society.

Knuth, D. (2001) *Things A Computer Scientist Rarely Talks About*. Stanford, CA: CSLI Publications.

Löwgren, J. and Stolterman, E. (2004) *Thoughtful Interaction Design: A Design Perspective on Information Technology*. Cambridge, MA: MIT Press.

Lau, A. and Vande Moere, A. (2007) 'Towards a model of information aesthetic visualization', *IEEE International Conference on Information Visualisation* (IV'07).

Malina, R.F. (2006) 'A forty-year perspective on aesthetic computing in the Leonardo journal', in P. Fishwick (Ed.), *Aesthetic Computing* (pp.43–52). Cambridge, MA: MIT Press.

Mankoff, J., Dey, A., Hsieh, G., Kientz, J. and Lederer, S. (2003) 'Heuristic evaluation of ambient displays', Paper presented in the Proceedings of the *SIGCHI Conference on Human Factors in Computing Systems*, pp.169–176.

Manovich, L. (1999) 'Avant-garde as software', *Versi´on Castellana: Mania*.

Milgram, P., Takemura, H., Utsumi, A. and Kishino, F. (1994) 'Augmented reality: a class of displays on the reality-virtuality continuum', Paper presented in the Proceedings of the *Telemanipulator and Telepresence Technologies*, pp.282–292.

Nöth, W. (1990) *Handbook of Semiotics*. Bloomington, Indiana: Indiana University Press.

Oram, A. and Wilson, G. (2007) *Beautiful Code: Leading Programmers Explain how they Think*. Sebastopol, CA: O'Reilly Media, Inc.

Park, M. and Fishwick, P. (2005) 'Integrating dynamic and geometry model components through ontology-based inference', *Simulation*, Vol. 81, p.795.

Quarles, J., Lampotang, S., Fischler, I., Fishwick, P. and Lok, B. (2008) 'Mixed reality merges abstract and concrete knowledge', *IEEE Virtual Reality 2008 Conference,* March 8–12, Reno, Nevada, pp.27–34

Redström, J., Skog, T. and Hallnäs, L. (2000) 'Informative art: using amplified artworks as information displays', Paper presented in the Proceedings of the *DARE 2000 on Designing Augmented Reality Environments*, pp.103–114.

Schmitt, G. (1999) *Information Architecture: Basis and Future of CAAD*. Basel, Switzerland: Birkhäuser.

Schneiderman, B. (1996) 'The eyes have it: a task by data type taxonomy for information visualization', *IEEE Visual Languages*, pp.336–343.

Tractinsky, N. (1997) 'Aesthetics and apparent usability: empirically assessing cultural and methodological issues', Paper presented in the Proceedings of the *SIGCHI Conference on Human Factors in Computing Systems*, pp.115–122.

Tractinsky, N., Katz, A. and Ikar, D. (2000) 'What is beautiful is usable', *Interacting with Computers*, Vol. 13, pp.127–145.

Udsen, L. and Jørgensen, A. (2005) 'The aesthetic turn: unravelling recent aesthetic approaches to human–computer interaction', *Digital Creativity*, Vol. 16, pp.205–216.

Vande Moere, A. (2008) Available at: http://infosthetics.com.

Vogel, D. and Balakrishnan, R. (2004) 'Interactive public ambient displays: tran-sitioning from implicit to explicit, public to personal, interaction with multiple users', Paper presented in the Proceedings of the *17th Annual ACM Symposium on User Interface Software and Technology*, pp.137–146.

Ware, C. (2004) *Information Visualization: Perception for Design*. San Francisco, CA: Morgan Kaufmann.

Weiser, M. (1993) 'Ubiquitous computing', *IEEE Computer*, Vol. 26, pp.71–72.

Weiser, M. and Brown, J. (1996) 'Designing calm technology', *PowerGrid Journal*, Vol. 1, pp.75–85.

Wisneski, C., Ishii, H., Dahley, A., Gorbet, M., Brave, S., Ullmer, B. and Yarin, P. (1998) 'Ambient displays: turning architectural space into an interface between people and digital information', Paper presented in the Proceedings of the *First International Workshop on Cooperative Buildings, Integrating Information, Organization, and Architecture*, pp.22–32.

Zhang, K. (2003) *Software Visualization: From Theory to Practice*. Norwell, MA: Kluwer Academic Publishers.