34

"So the Colors Cover the Wires": Interface, Aesthetics, and Usability

Matthew G. Kirschenbaum

Introduction: A Glass Darkly

The idea of interface, and related concepts such as design and usability, are some of the most vexed in contemporary computing. Definitions of interface typically invoke the image of a "surface" or a "boundary" where two or more "systems," "devices," or "entities" come into "contact" or "interact." Though these terms encourage spatial interpretation, most interfaces also embody temporal, haptic, and cognitive elements. The steering wheel of a car, the control panel of a VCR, and the handle of a door are all examples of everyday interfaces that exhibit these dimensions. In the context of computers and computing, the word "interface" is often used interchangeably with "graphical user interface," or GUI, most frequently encountered as a desktop windows environment. The command line prompt is perhaps the best-known alternative to the GUI, but there are a plethora of others, including screen readers, motion trackers, tangible user interfaces (TUIs, breathtakingly rendered in the 2002 film *Minority Report),* and immersive or augmented computing environments. In the humanities, meanwhile, it is increasingly common to encounter the idea that a book or a page is a kind of interface, a response to the understanding that the conventions of manuscript and print culture are no less technologically determined than those of the digital world. At least one observer, Steven Johnson, has defined our present historical moment as an "interface culture," a term he wields to embrace not only the ubiquity of computers and electronic devices but also the way in which interface has come to function as a kind of trope or cultural organizing principle — what the industrial novel was to the nineteenth century or television to the suburban American 1950s are his examples.

As much as it is talked about, however, interface can at times seem little loved. Usability guru Donald A. Norman writes: "The real problem with interface is that it is

an interface. Interfaces get in the way. I don't want to focus my energies on interface. I want to focus on the job" (2002: 210). Nicholas Negroponte holds that the "secret" of interface design is to "make it go away" (1995: 93). To further complicate matters, interface is often, in practice, a highly recursive phenomenon: Take the experience of a user sitting at her computer and browsing the Web, perhaps accessing content at a digital humanities site. The site's internal design imposes one layer of interface between the user and the content, and the web browser — its buttons and menus and frames — immediately imposes another. The user's desktop environment and operating system then impose a third layer of interface. The ergonomics of the situation (we'll assume our user is working with a keyboard and mouse, looking at a screen positioned the recommended 18 inches away) create still another layer of interface, a layer which becomes apparent when one considers alternatives such as accessing the same content with a PDA or a wearable device, or in a room-based virtual reality setting like a CAVE. Importantly, each of these "layers," as I have been calling them, exhibits the potential for interaction with one another as well as with the user. The desktop environment governs the behavior of the browser software, whose features and functions in turn directly affect many aspects of the user's interaction with the site's internal design and content.

While everything I have just been rehearsing is familiar enough in computer science circles, particularly the domain known as human—computer interaction (HCI, also sometimes identified as human—computer interface), aspects of this narrative may seem problematic to readers trained in humanities disciplines. It would not be hard to find someone willing to argue that my entire scenario is the product of yet another unacknowledged interface, a kind of common cultural gateway whose socially constructed ideologies govern our expectations with regard to technology, representation, and access to information. Moreover, in the scenario sketched above, my distinction between different layers of interface and something I casually called "content" is one that runs counter to decades of work in literary and cultural criticism, where form and content are almost instinctively understood as inextricable from one another. Thus, the weight of established wisdom in a field like interface design rests on a fundamental disconnect with the prevailing intellectual assumptions of most humanists — that an "interface," whether the windows and icons of a website or the placement of a poem on a page, can somehow be ontologically decoupled from whatever "content" it happens to embody. This is precisely the point at which Brenda Laurel begins her dissent in *Computers as Theatre*, her influential critique of mainstream interface theory:

> Usually we think about interactive computing in terms of two things: an application and an interface. In the reigning view, these two things are conceptually distinct: An application provides specific functionality for specific goals, and an interface represents that functionality to people. The interface is the thing that we communicate with — the thing we "talk" to — the thing that mediates between us and the inner workings of the machine. The interface is typically designed last, after the application is thoroughly conceived and perhaps even implemented; it is attached to a preexisting bundle of "functionality" to serve as its contact surface. (Laurel 1991: xv)

Laurel is writing to challenge this prevailing viewpoint. Yet the truth is that, from a developer's perspective, the interface is often not only conceptually distinct, but also

computationally distinct. As John M. Carroll points out, it wasn't until the comparatively recent advent of languages like Visual ,Basic that it even became practical to program both a user interface and an application's underlying functionality with the same code (Carroll 2002: xxxii). Like the history of hand-press printing, which teaches us that paper-making, typesetting, etching or engraving, and bookbinding came to encompass very different domains of labor and technical expertise (rarely housed under the same roof), resource development in the digital world is typically highly segmented and compartmentalized.

Interfaces are mostly discussed in mundane and utilitarian terms, but computing lore contains its share of instances where poor interface design has had lethal and catastrophic results. One notorious episode involved the Therac-25, a machine employed for cancer radiation therapy in the mid-1980s, whose cumbersome software interface contributed to several patient deaths from overexposure. Likewise, the small-plane accident that killed singer-songwriter John Denver has been attributed to a poorly designed cockpit interface, specifically the'placement of a fuel switch. While the stakes in digital humanities research are (happily) other than life and death, interface is nonetheless an indispensable component of any project.

Indeed, interface presents a number of interesting and unique problems for the digital humanist. Understandably driven by pragmatic and utilitarian needs, the interface is also where representation and its attendant ideologies are most conspicuous to our critical eyes. Ostensibly wedded to the ideal of user-friendliness, the interface is also where we deploy our most creative features and imaginative flourishes. Too often put together as the final phase of a project under a tight deadline and an even tighter budget, the interface becomes the first and in most respects the exclusive experience of the project for its end users. Seemingly the most creative or intuitive stage of the development process, the interface is also potentially the most empirical, subject to the rigorous quantitative usability testing pioneered by HCI. This chapter makes no attempt to offer a comprehensive survey of the vast professional literature on interface and usability, nor does it seek to serve as a design primer or guide to best practices. (Readers interested in those topics should consult the chapter's suggestions for further reading.) Instead, in the pages that follow I seek to develop some broader frameworks for thinking about the major challenges that interface poses for both theorists and developers in the digital humanities.

Ways of Seeing

Computers compute, of course, but computers today, from most users' points of view, are not so much engines of computation as *venues* for *representation.* The first use of CRT displays as output devices in the 1950s irretrievably situated computers within a complex cultural genealogy of screens, a genealogy which also includes video, television, cinema, photography, and indeed, as Lev Manovich and others have argued, the full lineage of visual aesthetics in the West since the advent of perspective. This context is important because it allows interface design to take its place alongside the other representational forms that have inhabited our many varieties of screens, frames, and windows. Moreover, although a graphical user interface built around the familiar desktop and windows metaphor currently constitutes the normative interface experience for the vast majority

of users, it is worth remembering that this particular graphical environment, and the representational practices it encourages, is historically quite specific.

From one perspective, our interface and display technologies have remained remarkably stable and consistent over the past thirty years. Though the Alto, built at Xerox PARC in 1973, is widely regarded as the first computer to implement a functional graphical user interface, the more important date is probably 1968, when Stanford's Douglas Engelbart demonstrated an operational GUI to a standing-room-only audience in San Francisco. Steven Johnson articulates the importance of Engelbart's presentation, which included a feature recognizable as "windows," this way:

> [H]istorians a hundred years from now will probably accord it the same weight and significance we now bestow on Ben Franklin's kite-flying experiments or Alexander Graham Bell's accidental phone conversation with Watson. Engelbart's thirty-minute demo was our first public glimpse of information-space, and we are still living in its shadow. (Johnson 1997: 11)

By "information-space," Johnson means the abrupt transformation of the screen from a simple and subordinate output device to a bounded representational system possessed of its own ontological integrity and legitimacy, a transformation that depends partly on the heightened visual acuity a graphical interface demands, but ultimately on the combined concepts of interactivity and direct manipulation. Engelbart's demo included the first public use of a mouse, and the sight of its pointer sweeping across the screen instantly collapsed the stark input/output rhythms of batch-process and command-line computing into a single, continuous sweep of user activity. Just as important, however, was the spectacle of Engelbart dividing his screen into distinct regions, heterogeneous in content but spatially adjoining, a feat demonstrated most dramatically by a window with a live audio and video feed to a colleague in Menlo Park, California. The computer — or more specifically, the screen — had clearly become a much more complex representational space, an *information* space whose surface owed as much to modernist collage as it did to brute force calculation. A crucial refinement came several years later when a team at Xerox PARC, led by Alan Kay (and including many of Engelbart's former Stanford colleagues), arrived at the realization that windows could actually *overlap*, thereby immediately imbuing the screen with the third dimension we take for granted today. As Johnson suggests, "The whole idea of imagining a computer as an environment, a virtual world, comes out of this seemingly modest innovation" (1997: 47).

While Engelbart's 1968 demo is the most venerable touchstone for contemporary human—computer interaction, there are other origin stories that bear repeating. Ivan Sutherland, for example, working on a PhD thesis at MIT in 1963, introduced Sketchpad, a system that allowed users to draw lines on a screen in real time with what we would today recognize as a light pen. Sketchpad is significant because it reminds us that the current hegemony of mouse and keyboard was not always in place, and indeed, there are indications that alternative input devices like light pens — which fundamentally alter the nature of one's bodily interaction with a computer — may once again displace the mouse (see this chapter's "Coda" section, below). Sketchpad is also significant in another context, the history of computer graphics (without which there would be no graphical user interfaces). Nicholas Negroponte comments:

> The achievement was of such magnitude and breadth that it took some of us a decade to understand and appreciate all of its contributions. Sketchpad introduced many new concepts: dynamic graphics, visual simulation, constraint reduction, pen tracking, and a virtually infinite coordinate system, just to name a few. Sketchpad was the big bang of computer graphics. (1995: 103)

Sketchpad was a vector system, meaning that the lines and shapes drawn by the user were mathematical formulas (vectors) that could be reproduced at will on-screen. Vector images provide another important context for understanding the significance of Englebart's work because the latter led directly to Xerox PARC's refinement of bitmapping as an alternative to Sketchpad and the era's prevailing vector displays. A "bitmap," as many readers will know, is a grid or matrix of pixels ("picture elements"), which, not unlike a Seurat painting or a photographic halftone, yield a coherent visual image through the optical interpretation of the aggregate composition. Bitmapped displays are what permit the gorgeous, high-quality facsimile images that we see in many of today's digital humanities projects, but their significance is much greater. (Note that bitmap displays, which are also known as raster displays, can refer to individual image files in formats such as JPEG, TIFF, or GIF, as well as to an entire screen display; there is also a proprietary image format known as "Bitmap," or BMP, which is not to be confused with bitmapping as a general concept.) If vector images were the graphical inheritance of the computer's mathematical roots, bitmapped images were the visual realization of Turing's ideal of the universal machine: bitmaps enabled the computer screen to function as a representational surface capable of emulating *other* representational surfaces. Through bitmapping, the computer screen was transformed into a second-order or "meta" representational venue. This transformation quickly gave rise to intensive research on photorealistic rendering techniques in computer graphics as well as (eventually) the advent of hardware devices like scanners and digital cameras — which enable the computer screen to operate in the photographic tradition. (JPEG compression algorithms, it is worth noting, were introduced precisely to provide an image format that lent itself to reproducing photographic images.)

William M. Ivins, in *Prints and Visual Communication* (1953), his landmark survey of print-making technologies in the West, argues eloquently for the importance of photography and what he terms "exactly repeatable visual statements" in enabling the dissemination of scientific knowledge. Bitmapping, I would argue, endows the computer screen with much those same qualities and capabilities, and although Manovich is right to point to the origins of computer graphics in the vector images of Cold War radar displays, the visual authority of computers as we know it today clearly owes more to the advent of bitmapping. Taken together, however, Sutherland and Englelbart laid the foundations for contemporary computer graphics and today's graphical user interface through their competing paradigms of vector and bitmap displays; competing not in a strict commercial sense, but in offering alternative visions of the computer as a representational medium and as an information space.

Unlike bitmap or raster graphics, vector graphics are not well suited to representing continuous tone (especially photographic) images. Consequently, they may seem of little use in the digital humanities, where much of our work consists in providing high-quality facsimile renderings of documents, artwork, and other artifacts of cultural heritage.

However, because vector graphics exist as a kind of mathematical abstraction, with no one-to-one mapping to an external referent, they are scalable and modular in ways that raster graphics are not. Today the most popular vehicle for vector graphics is the animation tool Flash, which, characterized by its colorful, dynamic displays, is rapidly colonizing large segments of the Web; indeed, there are those who believe that at the level of interface design the Web itself will eventually be made over as an animated Flash-based environment, with static HTML (more likely, XML) documents existing as subsidiary, special-purpose content. Interestingly for our purposes, Flash is also capable of supporting embedded bitmapped images, suggesting that the representational field I described earlier has receded by one full order of magnitude and that vector graphics are now the true heir to Turing's universalism. (On the other hand, it remains true that all general-purpose screen displays, whether LCD or CRT, are rendered as bitmaps.)

To build a Flash animation, the designer creates a so-called "movie" consisting of a series of timed, sequenced, or triggered events. For some this may suggest that the Web is evolving into a medium that owes more to television than to the now-familiar (and comfortably humanistic) conceit of the "page." I would cast the argument differently. If, as Jerome McGann has repeatedly said, computers lend themselves to representing books because they exist on a different material plane, then vector images, whose mathematical construction allows them to operate in registers unavailable to raster images, may offer the opportunity for a similar re-conception of our current information spaces. Whereas Alan Kay's overlapping windows added a third dimension to our interfaces, event-based vector graphics may, I believe, give rise to more playful and pliable information *places:* interfaces that occupy a material middle ground between the bitmapped data objects they enfold and the bitmapped surface of the screen.

If all of that sounds hopelessly speculative and abstract, there are nonetheless some tantalizing glimpses to be had from the digital art world. Tomoko Takahashi's *Word Perhect [sic),* for instance, a Flash-based parody of a typical word processing interface: users encounter what appears to be a hand-drawn cartoon GUI, roughly limned in black ink on a white background. Typing generates crude, seemingly handwritten, less-than-perfect characters on the screen. Takahashi's word processor is fully functional, but the interface yields an inversion of the typical user-friendly experience, one that serves to underscore the distinctive materialities of both print and electronic textuality. Clicking the Mail icon produces the following set of instructions, which appear as a scrap of notepaper "taped" to the screen: "print the document. put into an envelope or ssomething similair *[sic)* which can contain the document. Go to post office and weigh it and buy stamps ... " and so on, for another hundred words, including further typos and blemishes. Jason Nelson, another Flash artist, in a piece entitled *the last machine with moving parts,* deploys the color picker interface familiar to users of image processing and paint programs to control the sequence of events in an animated word poem. As the reader moves the picker over the color palette, arrangements of words are pulled in and out of the visual field. "So the colors / cover the wires," reads this text at one point — and so the brilliant disguises of Nelson and Takahashi's localized parodies of interface culture cover (and discover) the hardwired histories of our information spaces.

The Artist of the Beautiful

Mary Shelley's 1818 novel *Frankenstein* is often invoked in discussions of computing for its meditation on the dualistic nature of science and technology, and has long been read as a parable of the promise and the peril of the Promethean flame. Yet *Frankenstein* is also, importantly, a novel of aesthetics. The anonymous creature at the center of the text ("Frankenstein's monster") is described repeatedly as a "wretch," "a thing such as even Dante could not have conceived" (Shelley 1992: 57). In fact, the creature's visage is so hideous that apart from Victor (his creator), the only person who can stand, to conduct an extended conversation with him is a blind man.

It might be tempting to adopt *Frankenstein* for my own ends and make Shelley's creature into a figure for the graphical user interfaces of the current day and age; wretched and hideous are qualifiers we can debate, but there is no doubt that most of our desktop views are oddly unlovely, dull and listless information spaces that, as has been pointed out many times, hew to the conspicuously corporate metaphor of the office. But I would like to turn instead to another, less well-known text in the *Frankenstein* tradition: Nathaniel Hawthorne's (1844) short story, "The Artist of the Beautiful." It is the tale of one Owen Warland, a watchmaker with an exquisitely refined aesthetic sensibility who, bored with crafting mechanisms in the service of "old blind Father Time," eventually succeeds in his experiments to "spiritualize machinery," imbuing a tiny, mechanical butterfly with the living breath of his artist's imagination:

> He produced, as he spoke, what seemed a jewel box.... This case of ebony the artist opened, and bade Annie place her fingers on its edge. She did so, but almost screamed as a butterfly fluttered forth, and, alighting on her finger's tip, sat waving the ample magnificence of its purple and gold-speckled wings, as if in prelude to a flight. It is impossible to express by words the glory, the splendor, the delicate gorgeousness which were softened into the beauty of this object. Nature's ideal butterfly was here realized in all its perfection; not in the pattern of such faded insects as flit among earthly flowers, but of those which hover across the meads of paradise for child-angels and the spirits of departed infants to disport themselves with. The rich down was visible upon its wings; the lustre of its eyes seemed instinct with spirit. The firelight glimmered around this wonder – the candles gleamed upon it; but it glistened apparently by its own radiance, and illuminated the finger and outstretched hand on which it rested with a white gleam like that of precious stones. (Hawthorne 1948: 235-6)

Unlike Shelley's *Frankenstein,* however, there is some ambiguity as to the true nature of Warland's creation — whether it is indeed a living creature or a delicate clockwork automaton. "But is it alive?" the onlookers in the story ask repeatedly. Warland refuses to answer definitively: " `Wherefore ask who created it, so it be beautiful?' ... `it may well be said to possess life, for it has absorbed my own being into itself; and in the secret of that butterfly, and in its beauty, — which is not merely outward, but deep as its whole system, — is represented the intellect, the imagination, the sensibility, the soul of an Artist of the Beautiful!" (1948: 237).

As in *Frankenstein,* "The Artist of the Beautiful" is structured by the classic binaries of nature and culture, animate spirit and technological artifice; but it is also a story about

form versus function, art for art's sake colliding with honest industry and labor. Warland's chief rival and antagonist is Robert Danforth, the village blacksmith, whose response to the butterfly is to exclaim, "There is more real use in one downright blow of my sledge hammer than in the whole five years' labor that our friend Owen has wasted on this butterfly" (1948: 238). These are the same binaries that have structured debate in interface and design since Apple introduced the Macintosh; it is not hard to hear the echo of Danforth's sledgehammer in the firm finger striking the Return key at the end of the command line, or to see Warland's labored handicraft reflected in the incessant mouse manipulations of the GUI. What role, then, should aesthetics have in interface design? How do we balance the competing demands of truth and beauty? For while most software and websites have pragmatic or functional ends, an interface such as Apple's OS X — with its zooming windows, gossamer transparencies, and luscious drop shadows — encourages us to cultivate an aesthetic sensibility even in the most mundane corners of the desktop.

Yale computer scientist David Gelernter has written at length on the subject of aesthetics in software design:

> Most computer technologists don't like to discuss it, but the importance of beauty is a consistent (if sometimes inconspicuous) thread in the software literature. Beauty is more important in computing than anywhere else in technology.... Beauty is important in engineering terms because software is so complicated.... Beauty is our most reliable guide to achieving software's ultimate goal: to break free of the computer, to break free conceptually.... But as we throw off the limits, what guides us? How do we know where to head? Beauty is the best guide we have. (Gelernter 1998: 22-3)

Gelernter, who often comes across as an unreconstructed Platonist in his writing, goes on to speak of "deep beauty," his term for an idealized integration of form and function that bears a striking resemblance to Owen Warland's statement (above) that the beauty he apprehends is "not merely outward, but deep as its whole system." Gelernter also quotes approvingly Ted Nelson's dictum that "the integration of software cannot be achieved by committee.... It must be controlled by dictatorial artists" (1998: 22). Historically, however, human—computer interaction has its origins not in the poet's eye in a fine frenzy rolling, but rather in the quantitative usability testing of the sort pioneered by Ben Shneiderman and his colleagues in the 1970s. (Shneiderman himself, interestingly, has just recently put forward the Renaissance artist/technologist Leonardo da Vinci as the inspirational muse for his vision of a user-oriented "new computing.") I propose to begin addressing the aesthetics of interface by narrowing the field to look closely at two projects, both from the realm of text analysis, and each of which, it seems to me, exhibits a "beautiful" user interface.

The first is a prototype tool called Eye-ConTact. Conceived and implemented by Geoffrey Rockwell and John Bradley in explicit relation to its predecessor TACT, Eye-ConTact is perhaps not beautiful in the superficial sense of being (especially) pleasing to look at; nonetheless, I believe the program well instantiates Gelernter's "deep beauty" in its stated approach to textual analysis. Rockwell describes its operations this way:

> Eye-Contact deals with the problem of recording the logic of an exploration by encouraging the user to lay out the fundamental steps in a visual environment. The user creates the logic

> by dragging out icons and "connecting the dots." This has the advantage of acting as both a record of the flow of choices made and a synoptic description of that flow, which should make the research easier to grasp. (Rockwell, website)

Note then that Eye-ConTact requires the user to make explicit procedural choices which are then recorded and represented in an evolving graphical construction of each specific analytical operation. The software, in other words, serves to model a series of ongoing hermeneutical events (by contrast, TACT lacks the means to log a record of a user's operations with a text). Eye-ConTact is also a realization of what Bruce Tognazzini, who has had a distinguished career at Apple, celebrates as a *visible* (as opposed to a merely graphical) interface: "A visible interface is a complete environment in which users can work comfortably, always aware of where they are, where they are going, and what objects are available to them along the way" (Tognazzini 1992: xiii). Still another way of thinking about Eye-ConTact is as an example of what happens when an application adopts as central that which is peripheral in most other tools, in this case the History feature — an approach that offers the basis for some interesting experiments in software design (see also in this regard Matthew Fuller's essay on his Web Stalker browser). Eye-ConTact clearly illustrates the way in which a particular tool, rather than seeking to hide its user interface, can instead use that interface as an active — indeed essential — component of the intellectual activity it aims to support: "not merely outward, but deep as its whole system."

My second example is even more contemporary. In April 2002, information designer and digital artist W. Bradford Paley debuted a web-based tool called TextArc. Drawing upon publicly available electronic texts in the online collections of Project Gutenberg, TextArc produces intricate visualizations of novels and other literary works. Every word of the original text is rendered on-screen, both in a one-pixel font that reprints the entire work line by line clockwise around the perimeter of the display, and then again in a larger font, a cloud of words with those appearing most frequently clustered brightly in the center. Paley describes the result as "[slome funny combination of an index, concordance, and summary." Clicking on a word highlights all of its appearances within the visualized text as well as generating rays or spokes connecting them so that a user can study whatever patterns may emerge. The visualization is also dynamically linked to a clear reading text of the work and a keyword-in-context concordance, and together these tools offer a powerful package for textual investigation. What is most striking about TextArc, however, is not its analytical engine, but rather its gorgeous, luminescent fields of display that seem to subordinate traditional hermeneutics to more stochastic modes of knowledge representation. The visualizations have a marked aesthetic dimension, asserting their integrity on a purely visual register independent of any functional use to which they might be put. Paley understood this from the start, and indeed supports TextArc (which is free) in part by selling hard copies of its output, each offset-printed and ready for framing on high-quality paper.

TextArc has received a great deal of positive press, including mainstream coverage in the *New York Times;* yet the basic functionality it provides — word frequency counts, distribution patterns, and keyword-in-context displays — has long been available with other tools. The platform-independent Oxford Concordance Program, which was capable of generating word lists, indices, and concordances for text analysis, first appeared in 1981, followed by desktop software such as WordCruncher (1985) and TACT (1988). Yet

all of these programs, as well as more recent packages like WordSmith Tools, deploy stark, ascetic interfaces very different from the illuminated constellations of a TextArc visualization.

Undoubtedly that contrast has much to do with the tools and development environments available to the authors of those earlier packages, as well as limited time and limited resources. These factors cannot be overstated: in one software project with which I was involved we spent eleven months of a one-year development period on the underlying architecture — which performed flawlessly — but mere days on the final user interface (all but unusable). Then the money ran out. Such problems are endemic to any institutional setting. But the digital humanities have also not yet begun (or else only just begun — see chapter 29, this volume) to initiate a serious conversation about its relationship to visual design, aesthetics, and, yes, even beauty. And just as Owen Warland's gilded creation fed the skepticism of Robert Danforth and the other artisans of his village, so too do conspicuous graphical displays often engender mistrust in contemporary academic settings — as though, like the traditional library carrel, our electronic surroundings have to be sensually impoverished in order to be intellectually viable. Visually suggestive interfaces are often derided as "slick," or "eye candy," or gratuitously "cool," or else — in an interesting bit of synaesthesia — too cluttered with "bells and whistles."

To understand what is at issue here we might return to Donald A. Norman's contention, quoted in the opening section: "The real problem with interface is that it is an interface. Interfaces get in the way. I don't want to focus my energies on interface. I want to focus on the job" (2002: 210). Interestingly, despite this fierce disavowal of interface, in their original typographic presentation Norman's words are printed in boldface, for emphasis. Pausing for a moment, we can enumerate many typographical conventions for cueing a reader: not just the weight, but also the size and face of the type, its justification, margins, and so forth. Not to mention more diffuse bibliographical features including paper, binding, illustrations, and even the size, shape, and heft of the codex. As scholars such as Johanna Drucker and Jerome McGann (among others) have long argued, these extra-linguistic elements cannot be written off as merely expendable or extraneous to "the text itself." They are, indeed, the working vocabulary of a particular graphical user interface that has become transparent to us only through long familiarity. All of us know how to *read* a modern newspaper or magazine in terms of its visual and typographic layout as well as its journalistic content. The debate over transparency in interface design that Donald Norman and Brenda Laurel (and more recently Matthew Fuller, Jay David Bolter and Diane Gromala) have all participated in thus mirrors the debate in the literary and textual studies community over the nature of a book's "contact surface," those physical and material features widely seen as incidental to the production of textual meaning. In both communities, the ideal of transparency is now being called into question and replaced with a broader awareness of how the visual (and aural, or tactile and olfactory) elements on page or screen function as integral aspects of the information experience, rather than as afterthoughts to some "pre-existing bundle of functionality."

Some may object that the language I have been using ("integral aspects of the information experience," "deep beauty," "deep as the whole system"), in addition to fostering a curious kind of New Critical organicism — for Cleanth Brooks, a poem was, famously, a well-wrought urn — is also militated against by actual information design practices,

which frequently make a virtue of the deliberate and explicit segregation of form and function. We see this in markup languages such as SGML and XML, when data tagged with a descriptive schema are then rendered with external stylesheets. Or if not stylesheets then so-called "skins," which allow different visual themes to be swapped in and out of a web page or desktop application. With the appropriate skins a generic MP3 application can be dressed to look like a jukebox or a crystal radio or in a thousand other guises. Likewise, a site called the CSS Zen Garden offers a striking demonstration of the power of stylesheets to represent the "same" underlying content in a dazzling array of different configurations. Moreover, as noted earlier, interface is recursive: thus we can use stylesheets to skin our data, skins to thematize our browsers and applications, and desktop themes to stylize our information spaces. Do we not then see interface and application cleaving apart in precisely the ways Laurel and Gelernter speak against?

The truth is that the variability introduced by such effects is literally and deliberately only skin-deep. 'Standards compliance and validated code are essential. At the Zen Garden site, valid XHTML and CSS "cleave" in the other sense of that Janus-word: they work together as an integrated whole fused by the centripetal force of an open, community-based standards environment. Littering this "road to enlightenment," notes the site's designer, are the "dark and dreary" relics of the past: "browser-specific tags, incompatible [Document Object ModelsI, and broken CSS support." The visual effects that play across the shimmering surface of a site such as the Zen Garden are now practical precisely because the Web design community has come to an understanding (as expressed by its standards and tools) that clearly regards a well-wrought information space as a deeply integrated system, not just a series of on-screen effects. Compare this to the state of affairs captured in David Siegel's old essay "Severe Tire Damage on the Information Superhighway," advertised as "an open letter to Netscape Communications and the WWW community," which details the baroque array of spoofs, tricks, workarounds, fixes, and kludges that characterized professional Web design circa 1996.

If there is a lesson here for the digital humanities it is simply this: just as interface cannot — finally — be decoupled from functionality, neither can aesthetics be decoupled from interface. Nor does talk of beauty always have to resort to mystification. I ultimately prefer the poet and critic Lisa Samuels to either Gelernter's neo-Platonism or Hawthorne's transcendentalism: "[W}e think that those parts of beauty which resist the translation back to knowledge are uselessly private and uncommunicative. In fact, they are what beauty `knows': that knowledge is also — perhaps most importantly — what we do not yet know.... Beauty is therefore endlessly talk-inspiring, predictive rather than descriptive, dynamic rather than settled, infinitely serious and useful." The important new wave of software studies (led by Manovich, Fuller, Bolter and Gromala, and others), which cultivates granular, material readings of the inevitable cultural and ideological biases encoded by particular applications and interfaces, offers one way to offset the potential risks of unvarnished aestheticism, and this literature should be watched — many of its contributors are also practicing digital artists. In the meantime, in the digital humanities, where we deal with the rich and varied legacy of cultural heritage, we ought to think about what it means to be artists of the beautiful ourselves. As we will see in the next section, however, being an artist isn't always easy.

The Blake Archive for Humans

"The Blake Archive for Humans" is the title of a page that collects William Blake resources on the Web. It is authored and maintained by an amateur enthusiast who links together a wide range of material. As visitors quickly learn, however, the page defines itself in part through explicit contrast to the Blake Archive for "non-humans": that is, the *William Blake Archive* (WBA), an extensive scholarly text- and image-encoding project that has been freely available online since 1995, with funding and support from the Getty Grant Program, the National Endowment for the Humanities, the Library of Congress, and the Institute for Advanced Technology in the Humanities at the University of Virginia. As of this writing the *Archive* contains SGML-encoded electronic facsimile editions of some 49 separate copies of all 19 of Blake's illuminated books, as well as a growing selection of paintings, drawings, separate plates, engravings, and manuscript materials. The *Blake Archive* has also won the Modern Language Association's 2002-3 prize for a Distinguished Scholarly Edition, the first time this award has been given to an electronic project.

What could possibly be wrong with any of that? The author of the "Blake Archive for Humans" site expresses his objections this way:

> Their site may be complete and accurate, but it is not particularly easy to use, and it's chock-full of all the ridiculous trappings of the scholarly profession. While such things certainly have their place, they can also interfere with the appreciation of the actual work itself. On the other hand, this site is the only place you're going to find this many of the illuminated books in forms you can actually read.... And it, too is a work in progress, so it will only be getting better; especially when we all have broadband connections.

I will say at the outset that I am not a disinterested party, having been affiliated with the *William Blake Archive* since 1997, first as its project manager and now as a consultant. If, however, as Blake claimed, "opposition is true friendship," then the opinions expressed above offer a friendlier-than-usual occasion for examining aspects of the WBA's interface – and the development and design process that produced it – in some detail as this chapter's central case study in usability. For the phrase "the Blake Archive for humans" can't help but raise the damning specter of divided loyalties, the suggestion that the *Blake Archive*, with its admittedly formidable layouts of menus and button panels and search tables, may ultimately be more suited to a machine's vision than the human eye. And there is a certain truth to that, since the evolution of the *Archive* has been marked by constant trade-offs between the (perceived) needs of our (human) users and certain non-negotiable demands of our adopted technologies. This divide between humans and machines is the crux of applied human–computer interaction, and is nowhere more visible than at the level of the user interface.

The notion that the WBA is "not particularly easy to use" and "chock-full" of "scholarly trappings" in fact typifies a certain range of responses we have received over the years. On the one hand, while we are happy to have users from many different constituencies, the site's primary mission has always been expressly conceived as scholarly research. To a certain point we make no apologies for that, since we emphatically believe

there is place for serious humanities scholarship — presented in all its customary "completeness and accuracy" — on the Web. On the other hand, however, the comments I quoted above clearly speak to a real, felt frustration that the most comprehensive online resource on its subject should sometimes prove awkward for the non-specialist to navigate and use. Don't we understand (I hear these users saying) that what really matters here is *Blake,* and access to "the actual work itself?" Sometimes a particular feature of the site, which may appear intrusive to lay users (one of those aforementioned bells and whistles), is indeed there for more specialized audiences; for example, the ImageSizer applet, which allows scholars to display Blake's work on-screen at its true physical dimensions. This slippage between scholarly complexity and bad design is suggestive, however, and deserving of further study — many years ago, in the pages of the *New York Times Review of Books,* Lewis Mumford and Edmund Wilson famously railed against the user interfaces they found in the edited texts of the Center for Editions of American Authors, with their "barbed wire" thickets of scholarly apparatus. I suspect something of that same dynamic is at work here, and the challenge would seem to lie in distinguishing legitimate intellectual complexity from what is merely a poorly executed design.

But that is perhaps easier said than done. The interface users encounter when they come to the *Archive* on the Web today is known to us behind the scenes as "WBA 2.0." In 2000-1 we undertook a major overhaul that introduced several new features into the *Archive* and also remedied certain design flaws that had been noted by users and reviewers.

As Stuart Curran wrote in his review of the 1.0 version of the site (figure 34.1), the then-current design was "luckily malleable and [could) be altered to accommodate altering circumstances." The revised design we see in WBA 2.0 (figure 34.2) clearly has been altered in a number of ways, and there is not space here to discuss all of them. A modest interface element, such as the arrow icons that are visible just beneath the pull-down menus toward the bottom of the screen, serves well to illustrate the nature of the revision process. The problem was that users who wished to view the images of a work in sequence were forced to scroll down the page to examine the image and perhaps work with the transcription, enlargement, or other associated features, and then scroll back up again to where the "Previous" and "Next" links were located. Moreover, the long vertical profile of the page meant that its top and bottom halves could not both be kept on-screen at the same time. This turned out to be particularly awkward in classroom settings, where the instructor was forced to do much scrolling upwards and downwards in order present students with a sequential series of images. The arrows situated in the lower portion of the screen now alleviate that extraneous scrolling, a simple design solution to a more or less straightforward usability problem.

The generous screen-space around the facsimile image of Blake's work (which contributes significantly to the page's vertical dimensions) is partly the result of the display area for our ImageSizer applet, but it is also a product of the site's aesthetic values. From the outset, the WBA was conceived as an image-based resource, with the "actual work itself" (to borrow our friendly critic's phrase) visibly central to the user's active eye, and buttons, links, and apparatus positioned either above or below. While the earlier version of the site relied on a tables layout for its presentation of this apparatus, WBA 2.0 employs pull-down menus. The pull-downs offer an efficient way to condense a number of different user options, but there was considerable internal debate among the project team as to whether or not they were the right solution. I reproduce below an abridged version of a long and

Previous Index Next

The Book of The& copy O, plate 1 (Bentley 2, Erdman ii, Keynes ii)

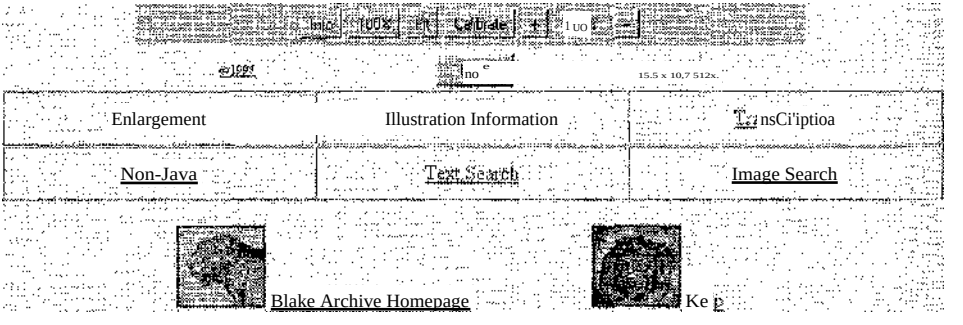

Figure 34.1 WBA 1.0: The *William Blake Archive (1997-2001)*

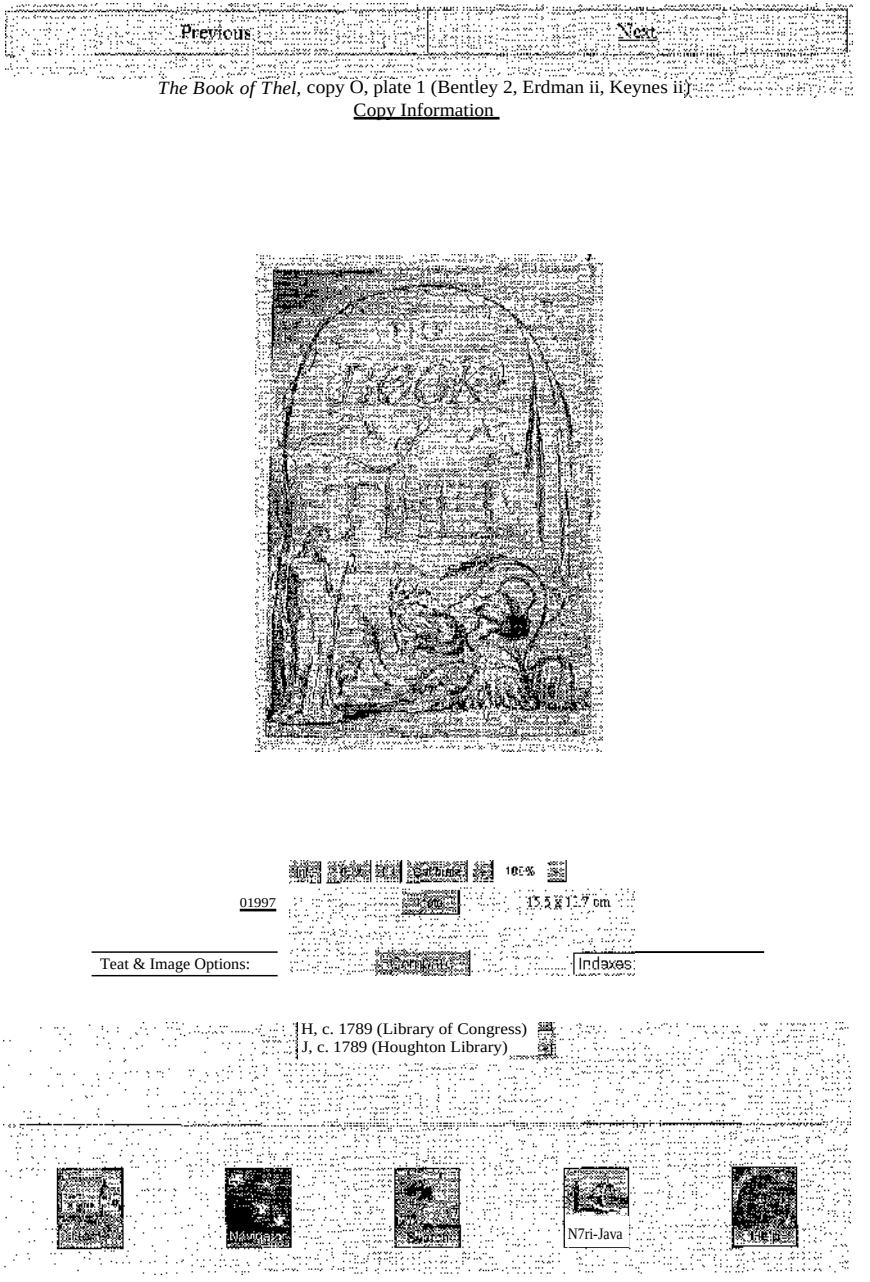

Figure 34.2 WBA 2.0: *William Blake Archive* (2001 to the present)

thoughtful piece of e-mail from Morris Eaves, one of the WBA's three general editors (Eaves, e-mail to blake-proj Discussion List, "WBA 2.0." July 11, 2000), not only because it captures well the flavor of those internal discussions, but also because it illustrates the way in which humanities scholars are nowadays thrust into the role of interface designer —

an impromptu role that some are quite comfortable in assuming, while others find it altogether unnatural. Eaves is clearly in the former category:

> I'm pretty sure I've read the whole discussion of the WBA 2.0 interface from start to finish. I know that time is short – but one of the nicest things about these digital confections is that they can be changed.... Here are a few personal reactions to the options that have been presented and discussed:...
>
> For me at least, there's a small chasm between form and function here. We all want the page to look good, and we all want it to respond readily and efficiently to the user's needs.... The way to make it look the best, probably, is to put everything on the page except the image behind a digital curtain and hide it from users until they want to do something – then they can open the curtain, rummage through those ugly old technical contents (previous/next, comparisons, enlargements, indexes, etc.) to find what they want, and close the curtain again to recreate the pretty image on a nice, paperlike background... .
>
> On the other hand, the way to make the page function the best is probably to pull down the curtain, arrange all the tools neatly, and label them clearly, so that the user always has everything needful right at hand. Commercial kitchens usually have open shelves and overhead racks; home kitchens usually have doors on the cabinets and tools in drawers that close.
>
> One of our problems is that our user's toolbox is so large – much larger than the toolbox of a book-reader. So we have a lot more to label, organize, and (maybe) hide until needed... . It's worth remembering how hard it is for us to look at this stuff without insider's prejudice – after all, we've been using Imagesizer and Inote and several of the other functions available in those lower areas of the page for a very long time now, so we constitute the least confusable set of users on earth. But I think it's very hard to imagine other users – and I don't mean dumb users, I mean very smart, alert, expert Blakean users – who wouldn't welcome some visual help in organizing that formidable display of tools and information below the image.
>
> PS: Final confession about *my* kitchen: Georgia and I have two big commercial overhead racks full of hanging stuff, but then, at her insistence, we also have doors on our cabinets and drawers that close. I may be showing my hand when I say that I've never been completely happy about those doors on the cabinets – we built open shelves in our Albuquerque kitchen and I always loved them. The Rochester kitchen compromise must have something to do with living together for 100 years.

As it happens, we arrived at a similar compromise (in much less time) for WBA 2.0, the doors variously put on and left off different interface elements. And as both Curran and Eaves note, "digital confections" like the WBA can always be changed — at least in theory. But in practice, as I suggested in my introduction, the interface tends to come very late in a project's development cycle (Eaves's recognition above that "time is short" is a reference to the fact that our lead programmer was about to be reassigned to another project). Most digital humanities scholarship is produced incrementally, in layers; at least for large-scale projects like the WBA, housed within particular institutional ecosystems, those layers tend to be more sedentary than we like to admit. The truth is that WBA 2.0 is unlikely to change much in its interface until the site's next major incarnation as WBA 3.0.

There is also another consideration, beyond the exigencies of project management. The conventional wisdom, reinforced by commercial marketing campaigns like Microsoft's

"Where Do You Want to Go Today?" and AT&T's "You Will," is that computers can do just about anything we want them to do. *As* programmers and developers know, however, there are always limitations that come with particular hardware and software environments, and those limitations render the computer an eminently material medium, this materiality not so different in its way from the characteristic marks and traces left by Blake's brushes and burins. Both versions of the WBA, for example, rely upon Inso's DynaWeb software to provide the stylesheets needed to display our SGML-encoded electronic editions in a conventional web browser. We have, in the process, customized the appearance and behavior of the DynaWeb environment to such an extent that it may be unrecognizable from its out-of-the-box implementation. But while DynaWeb has been an enormous boon to the Archive, making possible (among much else) powerful search functions, its idiosyncratic architecture clearly imposes constraints that manifest themselves at the level of the site's interface. In the case of the navigation arrows that I discussed above, for example, some readers may have wondered why we opted for icons on the bottom half of the page but retained the textual "Previous/Next" links up top. The answer is that (for reasons too esoteric to detail) the DynaWeb environment could not be made to accommodate the graphical image files of the icons in the top portion of the screen. Curran, likewise, notes the strong hierarchical nature of the WBA: "the user must descend four levels to get to the texts of the individual illuminated works.... This notion of penetrating to an inner sanctum is, of course, antithetical to Blake." We have attempted to rectify this in WBA 2.0 through the addition of Comparison and Navigator features, but the broader point is that this particular order of things is largely an artifact of the DynaWeb architecture, which was originally intended to support not Blake's illuminated visions but, rather, large volumes of text organized in top—down hierarchical structures. The technologies we work with at the WBA thus constantly make their presence felt, visibly and palpably pushing back against the interface we attempt to enfold around them. This situation is particularly acute in the digital humanities, where necessity often dictates that we adopt and adapt tools and technologies that were originally developed for other needs and audiences. If, as I suggested earlier, interface design is a dialectic between the competing demands of human and machine, then the art and science of usability lies in striking a balance between the two.

The *William Blake Archive is* — emphatically — for humans. But humans are not homogeneous, and different users will have different needs. While it will be impossible to please everybody all of the time, a design team must at least ensure that it. is meeting the needs of its most important user communities most of the time. We believe we can now make this claim. Following the example of the large portals, more and more sites on the Web are also trending towards user-customizable interfaces, and this is a potential long-term solution for a project like the WBA. But it is a solution that, like all others, will have to be tempered by the non-virtual realities of staffing and development time, software and data standards, and project funding.

Coda: Magic Carpet Ride

The basic conventions of the desktop windows GUI have not evolved much since their popular inception with the Apple Macintosh in 1984. Until recently, however, our display

hardware had not changed very greatly either (despite higher resolutions and the universal shift from monochrome to color). But if, as I insisted earlier, the addition of a glass pane in the 1950s irrevocably situated the computer within the sphere of visual representation, then everywhere today there are signs that computers may be on the verge of another broad-based shift in the tangible construction of their information spaces. The rise of featherweight laptops, tablet computers, PDAs, and wearable devices, on the one hand, and wall-sized or room-based projection and display systems, on the other, is even. now wrenching apart the Procrustean setup of the desktop workstation, which has forced users to accept what hindsight will reveal to be an almost unbearably constricted and contorted relationship with our machines (while the ongoing pandemic of carpal tunnel syndrome and other repetitive strain injuries offer more immediate and irrefutable bodily evidence). In this same climate, HCI research has also convincingly demonstrated that the conventions of the desktop GUI do not scale well to either larger displays, such as one might find with wall-sized projections, or to smaller displays, such as one now finds on a PDA.

What, then, does the future hold? Of one thing I am sure: the typewriter and the television set will not enjoy their conceptual monopoly over our computing machinery for much longer. My own ideal system might look and feel something like this. I think of it as a magic carpet: a rectangle of thin, flexible, waterproof plastic, perhaps 3 x 4 feet, which I carry about rolled up under my arm (or folded in a bag). I can lay it out on any tabletop or flat surface, or else unfold only a corner of it, like newspaper readers on a train. The plastic sheet is actually an LCD screen, with an embedded wireless uplink to the Web. Applications, both local and remote, appear on its surface like the tiles of a mosaic. I move them about physically, dragging, shrinking, or enlarging them with my hands, pushing and pulling them through the information space. Text entry is primarily by voice recognition. The keyboard, when needed, is a holographic projection coupled to a motion tracker. Data are stored on a solid state memory stick I keep on my keychain, or else uploaded directly to secure network servers.

All of this may sound like over-indulgent science fiction – "Hamlet on the holodeck," to borrow a phrase from Janet Murray. But in fact, most of the elements listed here – wireless networking, voice recognition, keychain data storage, touch-screen and tangible user interfaces – are already in common use. And the rest – the motion trackers, paper-thin LCDs, and holographic input devices – are all already at or past the initial development stage. The "magic" carpet is actually just an extrapolation of real-world research that is ongoing at places like the Tangible Media Group (and elsewhere) in the Media Lab at MIT, the Interactivity Lab at Stanford, the Metaverse Lab at the University of Kentucky, the GVU Center at Georgia Tech, and the Human–Computer Interaction Lab at the University of Maryland. While the fortunes of any one individual technology will invariably prove volatile, even a quick scan of these sites leaves no question that the next decade of research in interface, usability, and HCI will take us for quite a ride. One of the major challenges for the digital humanities in the coming decade will therefore be designing *for* interfaces (and designing interfaces themselves) outside of the 13- to 21-inch comfort zone of the desktop box.

WORKS CITED OR SUGGESTIONS FOR FURTHER READING

The Blake Archive for Humans. Accessed November 15, 2002. At <http://www.squibix.net/blake/>.

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

Carroll, J. M. (2002). *Human-Computer Interaction in the New Millennium*. Boston: Addison-Wesley.

Cooper, A. (1995). *About Face: The Essentials of User Interface Design.* Hoboken, NJ: John Wiley.

CSS Zen Garden: The Beauty of CSS Design. Accessed December 30, 2003. At <http://www.csszengarden.com/>.

Curran, S. (1999). Review of the *William Blake Archive. TEXT* 12: 216-19.

Drucker, J. (1994). *The Visible Word: Experimental Typography and Modern Art, 1909-1923.* Chicago: University of Chicago Press.

Fuller, M. (2003). *Behind the Blip: Essays on the Culture of Software.* Brooklyn, NY: Autonomedia.

Gelernter, D. (1998). *Machine Beauty: Elegance and the Heart of Technology*. New York: Basic Books.

GVU Center, Georgia Tech. Accessed May 16, 2003. At <http://www.cc.gatech.edu/gvu/>.

Hawthorne, N. (1948). *The Portable Hawthorne,* ed. M. Cowley. NewYork: Penguin Books.

Human-Computer Interaction Lab, University of Maryland. Accessed May 16, 2003. At <http://www.cs.umd.edu/hcil>.

Ivins, W. M. (1953). *Prints and Visual Communication.* Cambridge, MA: MIT Press.

Johnson, S. (1997). *Interface Culture: How New Technology Transforms the Way We Create and Communicate.* San Francisco: HarperEdge.

Laurel, B. (1991). *Computers as Theatre.* Reading, MA: Addison-Wesley.

Manovich, L. (2001). *The Language of New Media.* Cambridge, MA: MIT Press.

McGann, J. (1993). *Black Riders: The Visible Language of Modernism*. Princeton, NJ: Princeton University Press.

Metaverse Lab, University of Kentucky. Accessed November 15, 2002. At <http://www.metaverselab.org/>.

Negroponte, N. (1995). *Being Digital.* New York: Knopf.

Nelson, J. (2000). *The Last Machine with Moving Parts.* Accessed May 14, 2003. At <http://www.heliozoa.com/ending3.html>.

Nielsen, J. (2000). *Designing Web Usability: The Practice of Simplicity*. Indianapolis: New Riders Publishing.

Norman, D. A. (1990). Why Interfaces Don't Work. In B. Laurel (ed.), *The Art of Human-Computer Interface Design* (pp. 209-19). Reading, MA: Addison Wesley.

____ (2002). *The Design of Everyday Things.* New York: Basic Books.

Paley, B. *TextArc.* Accessed November 15, 2002. At <http://www.textarc.org>.

Raskin, J. (2000). *The Humane Interface: New Directions for Designing Interactive Systems.* Boston: Addison-Wesley.

Rockwell, G. Eye-ConTact: Towards a New Design for Text-Analysis Tools. *Computers and the Humanities Working Papers.* November 15, 2002. At <http://www.chass.utoronto.ca/epc/chwp/rockwell/index.html>.

Samuels, L. (1997). Poetry and the Problem of Beauty. *Modern Language Studies* 27, 2. At <http://epc.buffalo.edu/authors/samuels/beauty.html>.

Shelley, M. (1992). *Frankenstein,* ed. M. Hindle. London: Penguin Classics.

Shneiderman, B. (1998). *Designing the User Interface: Strategies for Effective Human-Computer Interaction.* Reading, MA: Addison-Wesley.

____ (2002). *Leonardo's Laptop: Human Needs and the New Computing Technologies.* Cambridge, MA: MIT Press.

Siegel, D. (1995-6). Severe Tire Damage on the Information Superhighway. At <http://www.dsiegel.com/damage/index. html> .

Stephenson, N. (1999). *In the Beginning Was the Command Line.* New York: Avon.

Takahashi, T. *Word Perhect.* Accessed May 14, 2003. At <http://www.e-2.org/commissions_wordperhect.html>.

Tangible Media Group, MIT Media Lab. Accessed May 16, 2003. At <http://tangible.media.mit.edu/>

Tog on Interface. Reading, MA: Addison-Wesley.

William Blake Archive. Eds. M. Eaves, R. N. Essick, and J. Viscomi. Accessed November 15, 2002. At <http://www.blakearchive.org>.