

Our Little Help Machines and Their Invisibilities

David K. Farkas
Dept. of Technical Communication
University of Washington
Seattle, Washington
farkas@u.washington.edu

Abstract

This paper examines the four kinds of invisibility Johnson-Eilola associates with minimalist help systems: fast information access that reduces user reflection and questioning, impersonal writing style that assumes the Shannon-Weaver communication model, narrow scope that leads to training but not teaching, and interface designs that oversimplify user tasks. For each of the four, the paper questions Johnson-Eilola's conclusions. Ultimately, the problems with truncated online help systems may disappear, as help systems are increasingly linking to the web, where adequate conceptual information is often supplied and opportunities for a social context for help are available.

H.5.2 User Interfaces—training, help, documentation

Keywords: online help, documentation, user interface

Johndan Johnson-Eilola's "Little Machines" is wide-ranging and provocative. It serves the field of user assistance by drawing us well beyond our usual concerns about usability and efficient production.

The starting point of the essay is the general recognition that documents, software products, and all other cultural artifacts necessarily reflect the outlook and agenda of those who create and disseminate these artifacts. Certain things are left in; certain things taken out; certain things are made more and less prominent. The agenda may well be benign, but it may not be. These artifacts, furthermore, very often present themselves to us in a neutral, self-effacing manner. The essay's central argument is that help systems and software user interfaces, through their neutral personas and the ease with which they are used, have now achieved the condition of invisibility. This invisibility is pernicious: It subtly restricts thought and leads people to limit their horizons of potential action.

As I read "Little Machines," invisibility has four aspects. I offer commentary on each and on the calls for change ("re-articulations") that make up the second half of the essay. Johnson-Eilola writes in a spirit of speculation and intellectual adventure, and his complex argument is composed of numerous strands. My commentary, therefore, does not closely map the structure of the essay. Rather, I try to contribute strategically to the dialog, focusing on points about which I have reservations.

Four Kinds of Invisibility and Their Consequences

The first kind of invisibility has to do with information access. In contrast to print manuals, which are rapidly becoming obsolete, online help enables users to access information so quickly and easily that the information users want just seems to be there for them. Although users benefit from this efficiency, their normal instincts to reflect and question are diminished.

In regard to this claim, I offer an observation that I think will be made both by many documentation professionals and computer users: Help is simply not that good. Johnson-Eilola shows us balloon help and tool tips, probably the most directly and easily accessible forms of online help. (He also cites wizards in an interesting part of the essay I turn to later.) For many of their needs, however, users consult forms of help (HLP files, CHM files, and the like) that are the online equivalent of print books.

With these forms of help, users typically browse through an online table of contents and an online index, much as they would in a print manual. If a natural language interface is available, the user still has to formulate a query and then pick from a list of suggested help topics, not always pertinent ones. Help technologies may advance until access to desired information is nearly instant and automatic, but Johnson-Eilola's claim for invisibility is not borne out by the help systems in use today.

Another form of invisibility is a function of writing. Help procedures are written in a trim, streamlined style designed for easy comprehension by busy users. These procedures speak in a quiet, even-toned, authoritative corporate voice. They retreat from human consciousness, but leave users with intellectual amnesia, and invite users to assume a passive role in regard to the documentation and, by extension, the corporation that wrote and issued it.

The essay ties this kind of writing and its consequences to Shannon and Weaver's outdated transmission theory of communication:

Among other things, when both designers and users accept invisibility as a goal in these systems (when, for example, users recognize themselves as the unspoken "You" of the command "Press the enter key") we participate in what's popularly known as the Shannon and Weaver model of communication from the 1940s, also

sometimes called the transmission or conduit model: Information passes down a channel from sender to receiver. The receiver's job is this: *Present yourself as a target.* (pp. 4-5)

It is not clear, however, that impersonal documentation has this effect on computer users or how salient this effect might be. As Johnson-Eilola himself acknowledges, people inevitably make meaning as they read, and their use of documentation is a "recursive, active, creative process." (p. 8) Indeed, users of documentation regularly form judgments (often harsh ones) about the documentation and the company that made the software, and they regularly depart from the guidance offered by the documentation.

It may well be that many technical writers, influenced by Shannon and Weaver, view communication as a mechanical transmission process, and certainly, we would have these writers (and everyone else) gain more sophistication about communication and rhetoric. But how people respond to streamlined, impersonal documentation, how people read and think, does not depend on whether the writer accepts the Shannon and Weaver model—or even whether readers themselves do.

It is worth noting that when computer users turn to third-party books—as they continue to do—they often encounter authors who project lively personas and who are often more than willing to point out shortcomings in the software they are writing about. Many users, then, are in a position to notice this difference in how they are addressed. When impersonal documentation can be seen as a rhetorical choice, it is not invisible.

Johnson-Eilola's argument extends to software interfaces. Much as standard procedures subtly project corporate values, so do the standardized, highly functional interfaces of most software applications. Citing Dickie and Cynthia Selfe, Johnson-Eilola notes how "contemporary interface mappings rely heavily on Eurocentric, corporate ways of seeing and working." (p. 19). True as this is, practical alternatives are not easy to find (though one might imagine interface "skins" that accord with the values of certain cultures). Perhaps the best solution—and this is one of the re-articulations proposed in the essay—is education, raising the awareness of technical writers and everyone else in regard to the agendas and outlooks that lie behind all cultural artifacts.

The third form of invisibility stems from the narrow scope of online help, the insistence on teaching com-

puter tasks (“training”) while largely excluding domain information (“learning”). So, for example, online help explains how to create a hanging indent but does not offer domain knowledge such as explaining the rhetorical implications of hanging indents in page design. Narrowly focused documentation appeals to users because of its brevity and to software companies because it costs less to produce. But in an important respect it does not ultimately serve people well: It does not accord with the complex circumstances in which we all live, and it does not help computer users envision what is possible in their lives. As a remedy to this form of invisibility, the essay endorses enriching help with domain information and the production of both online tutorials and printed books geared for broad-based learning.

I believe, along with Johnson-Eilola, that online help should include enough conceptual information for users to understand the consequences of carrying out tasks. On the other hand, the issues of how much domain knowledge to provide and how to provide it are long standing and difficult ones. It is not at all clear how much domain knowledge users want to view while trying to figure out basic computer tasks or how designers can integrate domain information with core documentation. Another problem is that in many instances no single body of domain information will serve the needs of diverse audiences. To return to Johnson-Eilola’s example, while the rhetorical implications of hanging indents in page design should certainly be explained in online tutorials and third-party books, explanations of this kind may or may not work well as a part of online help.

A significant part of the discussion of domain information is a negative critique of wizards. Johnson-Eilola’s stance is that wizards are most often used to artificially reduce complexity, to automate “things that probably aren’t very amenable to automation.” (p. 11) Two wizards in Microsoft Word—the resume wizard and the legal pleading wizard—are singled out for not providing sufficient domain information and, hence, for inviting users to create documents without really understanding the communication task at hand.

I find this position problematic on several grounds. To begin with, a great many software wizards automate basic computer tasks (e.g., configuring a hardware device) and are fully adequate for their purpose. Furthermore, both the MS Word resume and legal pleading wizard can be used advantageously simply to get a

head start on typing and formatting. (The legal pleading wizard is especially convenient; it lets me import contact names and addresses from Outlook Express.)

Granting that a wizard may lull users into creating a document thoughtlessly, users themselves should, I think, accept significant responsibility for the resumes and pleadings they create if they have no experience with such documents and do not seek guidance beyond what the wizard offers. Finally, I question whether we want (potentially) hundreds of millions of people to receive comprehensive instruction on the creation of resumes, legal pleadings, and other documents from Microsoft or any other single source.

The fourth kind of invisibility pertains to the design of software user interfaces.

Designers necessarily make some features more prominent than others. But this is problematic in a culture conditioned to the idea that “the distance between desire and result should be near zero.” (p. 15) Users gravitate to what is visible, and the relative invisibility of certain functions discourages the use of those functions. Button bars and toolboxes are cited as encouraging users to content themselves with the functions immediately in view.

The issues here parallel several that I’ve raised earlier. Perhaps Johnson-Eilola underestimates people’s willingness to reflect, form goals, plan, and then to seek out both the computer functions and the documentation they need. And perhaps there is no ready way to change people who are truly lazy and thoughtless.

The End of Our Little Machines

Finally, I suggest that with the emergence of the Internet our little machines of online help are actually disappearing, being absorbed into something much larger and more complex. Help is becoming a portal, a quick and convenient source of core information but also a gateway to very extensive resources. Many of these resources reside on the software vendor’s server, but others do not.

Increasingly, a user who is dissatisfied with the available help topics can access an extensive “knowledge base” of highly specialized information (e.g., known bugs and fixes). A library of tutorials, software patches, and other resources is often available. The Microsoft Office Template Gallery, for example, offers a surprisingly complete and specific set of sample resumes (e.g., “Demonstrating how your military expe-

rience translates into corporate skills”) along with links to books on resumes and other resume-related resources. Johnson-Eilola calls for adding a social dimension to online help, and the help-as-portal paradigm includes directing users to bulletin boards and live chat sessions. This richness of options works against invisibility.

This new paradigm for user support does not address all of Johnson-Eilola’s concerns, and it raises

new ones. Indeed, one risk in the new paradigm is that users will be led from one resource to another by clusters of software vendors, media corporations, and other powerful interests working in close partnership. Users, then, may think they are being shown links to the resources that best meet their needs when profit is the main criterion for providing these links. We will continue to need the kind of alertness to broad societal issues exhibited in this essay.