



Why Simple Is Better

BY GEOFFREY J. S. HART | *Fellow*

THE CONCEPT OF “working memory,” also called “short-term memory,” is important because it’s a limited resource. The human brain can only handle so much information simultaneously; when there’s no more room, something must be dropped to make room for something else. The information design goal of simplicity reminds us to seek ways to avoid overwhelming readers’ working memories.

A Simple Example

If you think of working memory as analogous to physical space on your desk, and long-term memory as analogous to your bookshelves, it’s easy to understand how the two forms of memory cooperate. Let’s



This column focuses on the principles of information design, the art and science of understanding how humans process and comprehend information, and using that knowledge to develop more effective ways to communicate. Please send comments and suggestions to ghart@videotron.ca.

say you’re planning a vacation and get out your reference books to start researching details:

- ▶ an atlas that provides maps of the area
- ▶ a short history to help you appreciate the culture of your destination
- ▶ your *Lonely Planet* guidebook
- ▶ your *Rough Guide* guidebook, to fill in the gaps in *Lonely Planet*
- ▶ your checkbook because (alas!) you must consider your budget
- ▶ a travel writer’s exposé that provides the dirt most guidebooks miss
- ▶ your laptop—every destination has a website (and books are so last century)

You’d love to put more things on the desk, like a notepad, but there isn’t room. If you want more resources, you must remove something from the desk to make room for another book from the shelves.

It’s not a coincidence that I listed seven items. George Miller’s famous “magical number seven” is much abused, but it’s a useful way to think about working memory. Don’t get hung up on the number: the important point is that the more things you ask someone to deal with simultaneously, the less space that remains in working memory. Each addition removes a space ... until there is none left. At that point, old things must disappear to make room for new ones.

A Technical Communication Example

How might this relate to technical communication? Consider the task of helping readers to follow a procedure. The items on their mental desktop probably include:

- ▶ their overall goal, of which the current procedure is only a small part
- ▶ the current procedure’s goal, which constrains how they perform the procedure
- ▶ the overall characteristics of the software interface they’ll use (this is “orientation” information: we must know where something is before we can act upon it)
- ▶ the number of the step they’re about to perform, so they can return to that step, move down a line, and continue with the next step
- ▶ the name of a menu item or button you’ve told them to look for

- ▶ its location within the overall interface (e.g., which menu or toolbar)
- ▶ details of what to do with or to that menu item or button

We're already up to seven items, and we haven't yet accounted for distractions (that fascinating conversation in the next cubicle or the ominous dentist appointment that afternoon); watching the software or hardware for feedback that something good or bad is happening; wondering whether we're about to make an irreversible error; and many other possibilities. No wonder so many people ignore our documentation and go straight to the local expert.

Implications

Clearly, people learn to cope with this complexity, otherwise nobody would ever get anything done. Coping strategies abound, such as learning the hardware (keyboard and mouse) and software (menus and toolbars) so well they become subconscious parts of how we work, and thus no longer need to be numbered among the seven items we can juggle simultaneously. Again, the important point is not the number seven, nor is it the things we can't control (such as distractions)—it's the fact that even seemingly simple situations are far more complex than they appear at first glance.

Understanding that this complexity exists gives us tools to solve it, starting with creating lists such as the two I presented earlier. By understanding the size of the list that must be held in working memory, we can seek ways to minimize the burden. For example, instead of using a single step to tell readers to find a menu, open it, select an option, select a tab in a dialog box, navigate to a heading in that tab, then select three options, we can break this into at least two separate steps:

- ▶ opening the dialog box and selecting the correct tab
- ▶ finding the relevant options and setting them
- ▶ any option that requires elaboration (when there are multiple alternatives that must be explained), should be treated as a single additional step

I'll describe three more approaches in the following paragraphs.

USE THE SAME MENTAL MODE. To facilitate locating part of an interface, provide a screenshot with that part highlighted. This lets readers move a visual image of what they're seeking into working memory; when they turn to the software, they don't have to translate a verbal description into a visual target, thereby freeing up the memory space that would hold that translation. The more visual the thing they're seeking, the more important it is to provide an image. Icons should always be presented as graphics, not described.

JUXTAPOSE RELATED THINGS. When readers must look back and forth between software and documenta-

tion, working memory is consumed by the positions of the current step in the documentation and the location of the interface parts used to accomplish that step. (This is why you'll often see people hold a finger on the page while trying to operate the software with their free hand.) Making it easier to glance back and forth would free up some working memory. For example, adding "affordances" in the interface, such as clear button names or tooltips, reduces the number of times readers must return to the documentation for basic information on how to use the interface. Providing documentation as context-sensitive online help eliminates the need to navigate through the help just to find the correct topic; moving the help window beside the software makes it easier to consult the documentation without becoming disoriented. If we use embedded help, for example, by organizing the interface from left to right and top to bottom in the same order as the procedure's steps, we further reduce the need to consult the documentation (here, to learn the sequence).

REORGANIZE THE INTERFACE. Understanding how people progress through the steps in a procedure gives us knowledge that a product's designers often lack. Programmers and engineers most often focus on the details of implementing features, not on integrating them within a workflow. Moreover, most either never use the products they produce or never ask anyone outside the design team to use the products, so they don't understand why an interface is unusable. By providing them with that knowledge, we help them stitch the feature list together into something that supports the product's users. This may involve simple things, such as eliminating irrelevant or rarely used options from a dialog box (such as placing them behind an Options button instead) or grouping related tasks into a single tab of a dialog box. It can also involve more complex processes, such as reconsidering the overall workflow.

Simplify!

This article can be boiled down into a single rule: Before you begin writing or designing, think about how your audience will use what you create. In the context of working memory, this means we should count the number of things users must keep in their head while they work, then look for ways to minimize that number. We won't always have the luxury of redesigning a product to reduce the burden on its users, but we can at least lighten that burden. **■**

Based on decades of writing, editing, and information design, Geoff (ghart@videotron.ca) has published more than 300 articles and the book Effective Onscreen Editing. Geoff has given workshops on writing, editing, information design, audience analysis, cross-cultural communication, and workplace survival. He works as a freelance scientific editor, specializing in ESL authors.