Seven golden rules of online Help design

Leaving the tools to one side, Matthew Ellison proposes seven guiding principles on which to base a better Help system.

Back in 1995, the brief for designers of online Help was quite clear-cut. Windows was the overwhelmingly predominant computing platform for business and home applications, and there was a single standard Help format: WinHelp 4. Microsoft published design guidelines for creating WinHelp systems, and most organisations chose to follow them. As a trainer in online Help techniques, the most common question I received from students around that time was 'How can I make my own Help look exactly like the Help for Microsoft Word?'.

Almost ten years later, the situation is not nearly so cut-and-dried. The web is rapidly becoming an established platform for delivering server-based applications, and other operating systems such as Linux are on the rise. The current Microsoft Help format (HTML Help 1.4) does not provide a solution for this new generation of web-based and cross-platform applications, and so organisations have turned instead to a wide range of other formats for user assistance. These formats include:

- Turnkey browser-based systems (such as WebHelp and WebWorks Help)
- Sun's JavaHelp
- Customised embedded user assistance, where the Help forms part of the user interface itself.

Since each of these Help formats has its own particular layout and presentation style, it would seem on the face of it that a set of global design guidelines is no longer appropriate or workable. However, if you cut through the technological jargon and hype, the overall objectives of online Help are the same as they ever were: to answer users' questions and to support the completion of users' tasks. Coupled with these unchanging goals are certain fundamental design principles, or 'golden rules', that can be applied whatever format of Help you are creating.

The following sections summarise briefly the top seven design rules that guide me in my own online Help development.

Rule 1: Don't force users to move between topics to solve a problem

One of the most common mistakes in online Help design is to structure the information in the same way as you would for sequential reading. With the trend towards generating online user assistance from the same source as printed documentation, this is becoming an even more significant issue. The fact is that, unless users are feeling at an extremely loose end, they don't read Help systems from start to finish: they dip in, selecting individual topics as and when they need them. This means that, as you write any given Help topic, you need to guard against taking prior knowledge for granted.

I think that users are best served by including within a single topic all the information they need to answer their question or solve their problem, even if that means duplicating information that appears in other topics. This saves users the time and trouble of navigating between topics to piece together the full solution, and enables them to return back to productive work more quickly. I realise that this strategy sometimes needs to be tempered by practical considerations such as maintenance and translation costs. However, with today's increasingly sophisticated content management systems, it is becoming possible to write small chunks of content once and have them show up as normal text within multiple Help topics.

Even if you haven't adopted content management, there are a number of information-layering techniques that you can use to reduce the need to navigate between topics. These include pop-ups, expanding glossary definitions (which can be stored in a single file) and Dynamic HTML drop-downs (as shown in Figure 1).

Rule 2: Only index the main theme of each Help topic

In my view, until natural language search really comes of age, the index is the most useful device for accessing specific information within an extended Help system. I have heard Shane MacRoberts, Help Program Manager at Microsoft, quote the fact that full-text search is more frequently used; however, I think this is probably due in part to the poor quality of many of today's indexes. (Have you noticed that the Help for Microsoft Office has not had a true index since version 2000?)

Although it is a good idea to include as many useful entries as you can in an index, it is still possible to 'over-index'. The most common mistake is to index terms within a topic without actually considering whether the topic provides useful further information about that term. This can result in users being frustrated by following false leads. Remember that one of the key goals of online Help is to answer users' questions. As you index each topic, it is a good idea to think of all the questions for which the topics provides the answer and to use the words and phrases from those questions.

You might even consider basing your index keywords on only the title of the
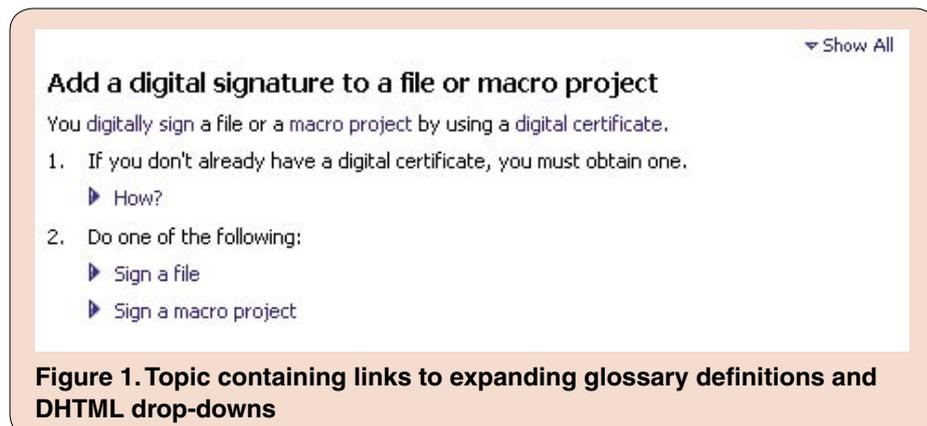
Figure 1. Topic containing links to expanding glossary definitions and DHTML drop-downs

topic, since that provides a sharper focus. However, don't misinterpret this as a licence to use only the wording of the titles themselves as index entries; my own usability tests have shown this to be worse than having no index at all. It is, therefore, essential to consider synonyms, phrase inversions, and other terms associated with the topic titles.

Rule 3: Don't require users to make a conscious decision to access Help

As authors of user assistance, it is sometimes easy to fall into the trap of assuming that our users are far more interested in the Help that we provide than is actually the case. We fondly imagine them seeking out the Help option and patiently browsing through our carefully crafted topics in their quest for knowledge. However, the hard reality is that many users have low expectations of Help and will do all they can to avoid using it. For that reason, they are unlikely to be tempted by a 'question mark' icon or a button labelled 'Help' since neither of these provides any direct evidence of the relevance or usefulness of the assistance that is available.

I prefer that links to online Help from the application user interface should be:
- Appealing
- Clear in terms of the information afforded
- In tune with the needs of users.

The link to user assistance shown in Figure 2 (and highlighted by me with an ellipse) is highly intuitive and, because it echoes the question that users are likely to have in their minds, it provides a compelling incentive for selecting it. What is at issue here is not the design of the Help itself, but the design of the link *into* the Help.
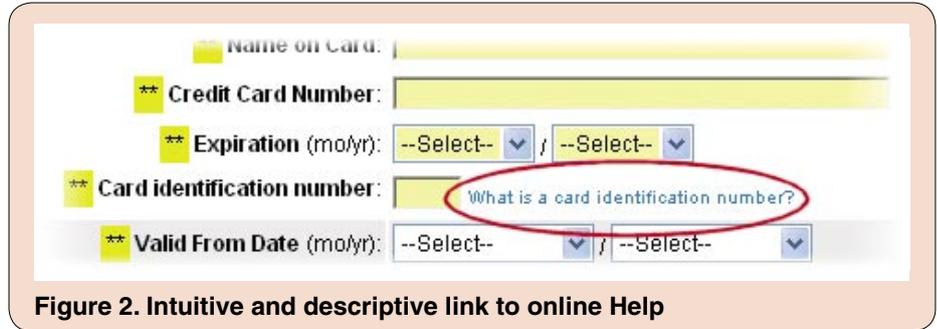
Figure 2. Intuitive and descriptive link to online Help

You can potentially use this technique for displaying any Help format, including a traditional HTML Help window.

Rule 4: Only include images when they add value

It is possible to overdo the use of graphics (especially screenshots) within online Help. I sometimes hear their inclusion justified by the fact that 'users like pictures' or 'graphics make the Help look more friendly'. Including images for no other reasons than these can lead to unnecessary additional maintenance costs, and can sometimes even be counter-productive. The fact is that users like pictures only when they are conveying useful information.

In general, online Help should contain far fewer screenshots than does paper-based documentation, if only because the user can often see the real screen side-by-side with the Help window — it makes sense for the Help to complement the application user interface, not duplicate it.

A problem with large images is that, due to the relatively small size of most Help windows, scrolling is required to view the images in their entirety. This means that, when the topic is first displayed, important information can sometimes be hidden out of view below the image.

The topic in Figure 3 shows the effective use of two small images that enable users to compare side-by-side the two contrasting ways of displaying a set of options.

Rule 5: Write topics that answer users' questions

This sounds obvious. However, it is common mistake for developers to write Help that is based on their own knowledge and understanding of an application, rather than on the questions and problems that users are likely to have.

I recommend writing a separate topic that addresses each potential question, and organising the Help table of contents by question types (such as 'how do I complete a task?', or 'what is this dialog box for?') rather than by the structure of the application.

Rule 6: Don't be a slave to consistency

As writers and editors, we sometimes allow ourselves to be inappropriately pre-occupied with consistency. How many times have you found yourself documenting a totally self-explanatory procedure (such as filling in a name and address) just for the sake of completeness? The important thing is to be guided only by the needs of the user as you decide which Help
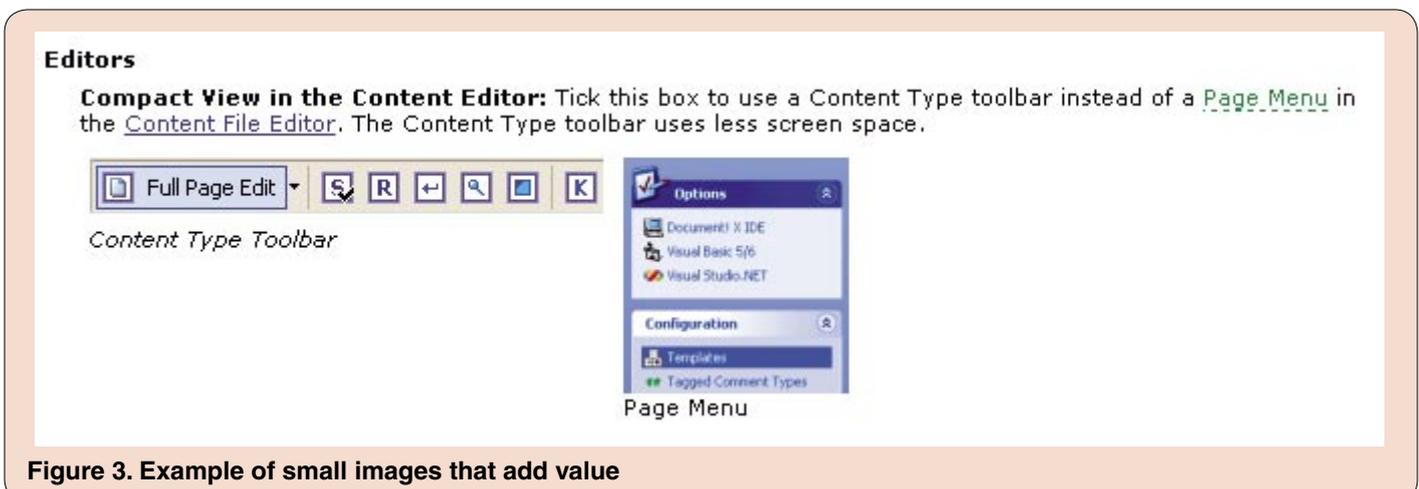

Figure 3. Example of small images that add value

Figure 4. Example of user assistance provided only where required

information to include and how to present it. The web-based application shown in Figure 4 provides a good example of this approach — all the column headings in this screen have links to context-sensitive Help except the Amount column, which is self-explanatory and does not require Help.

Rule 7: Don't be tempted to provide too much information

With the laudable intention of favouring the user with as much assistance as possible, we can sometimes run the risk of information overload. In general, the overriding dictum for online Help is 'less is best'. So, after finishing the research phase of your next Help project, be ruthless as you plan your content: focus on the questions that users will ask, and vigorously prune out anything that users don't need to know. **C**

Matthew Ellison is an independent consultant and trainer specialising in online Help design and technologies. He has 18 years' experience of working in software user assistance, and is a respected speaker at conferences and events in Europe and the USA. E-mail: matthew@ellisonconsulting.co.uk.

Useful resources

Books
Designing and Writing Online Documentation
by William Horton
ISBN 0-471-30635-5
Minimalism Beyond the Nurnberg Funnel
by John M Carroll (Editor)
ISBN 026203249X
Hot Text: Web Writing that Works
by Jonathan and Lisa Price
ISBN 0735711518
The Microsoft Manual of Style for Technical Publications
Published by Microsoft Press
ISBN 1572318902
Indexing: A Nuts-and-Bolts Guide for Technical Writers
by Kurt Ament
ISBN 0815514816

Websites
WritersUA User Assistance Resource Directory
www.winwriters.com/resource.htm
An up-to-date collection of tools, web resources, books, training providers, reviews, and articles relating to software user assistance.
Usable Help
www.g2meyer.com/usablehelp/about.html
Articles about the usability aspects of documentation and help systems for software and consumer products.
MSHelpWiki
www.mshelpwiki.com
A source of the latest news and information from Microsoft Help MVPs (Most Valuable Professionals).
Yahoo! Groups Help Authoring Tools & Techniques (HATT) mailing list
http://groups.yahoo.com/group/HATT/ (free subscription required)
A forum for online Help authors to share questions, views and information.