

The state of Help today

By Justin Darley. Thursday, 04 September 2003

The state of Windows Help today

In common with most areas of computing, online Help technology does not stand still. This article aims to take stock of the current situation, to look at how we got here and to glance tentatively in to the future. Help formats tend to be long-lived, consequently as new formats are developed the number of formats to choose between gets ever larger. The last section of this article provides some pointers for Help authors attempting to choose the right format for their documentation.

History of Help formats

Before we start to look at Help as it currently stands, and at where it may go from here, it makes sense to take a brief look at the history of Help and of the Help formats that have been and gone (or more accurately often *not* gone) over the last decade or so.

It will, I'm sure, come as no surprise to read that Microsoft® has been, until recently, far and away the most influential body in designing and implementing Help formats. However, as Cheri Lockett Zubak pointed out in her Microsoft Help Update session at the 2003 WinWriters Online Help Conference, Microsoft does not develop Help technologies for the Help community at large (Martin 2003). This is well worth bearing in mind when considering Microsoft's Help strategy and the formats that result from it. We will return to this point later. This trend has changed somewhat with the advent of the Web and it's fair to say that in this arena at least Microsoft is no longer setting the pace. The following sections are not exhaustive. Help did exist (I'm reliably informed) before Microsoft threw its hat into the ring. Earlier products from Microsoft and products from other vendors did include some form of Help. However the following are all Help formats:

- That I have some knowledge or experience of
- That most Windows® users will have seen or heard of.

WinHelp®

WinHelp has been released in two distinct and very different versions. WinHelp 3.1 (sometimes referred to as WinHelp 1.0) first shipped in 1990 with Windows 3.0™. Navigation in WinHelp 3.1 was purely by means of hyperlinks and a search dialog with Index and full-text search (Find) tabs. There was no table of contents - the Contents toolbar button simply took users to the main navigation topic in the Help.

Figure 1: WinHelp 3.1 contents topic

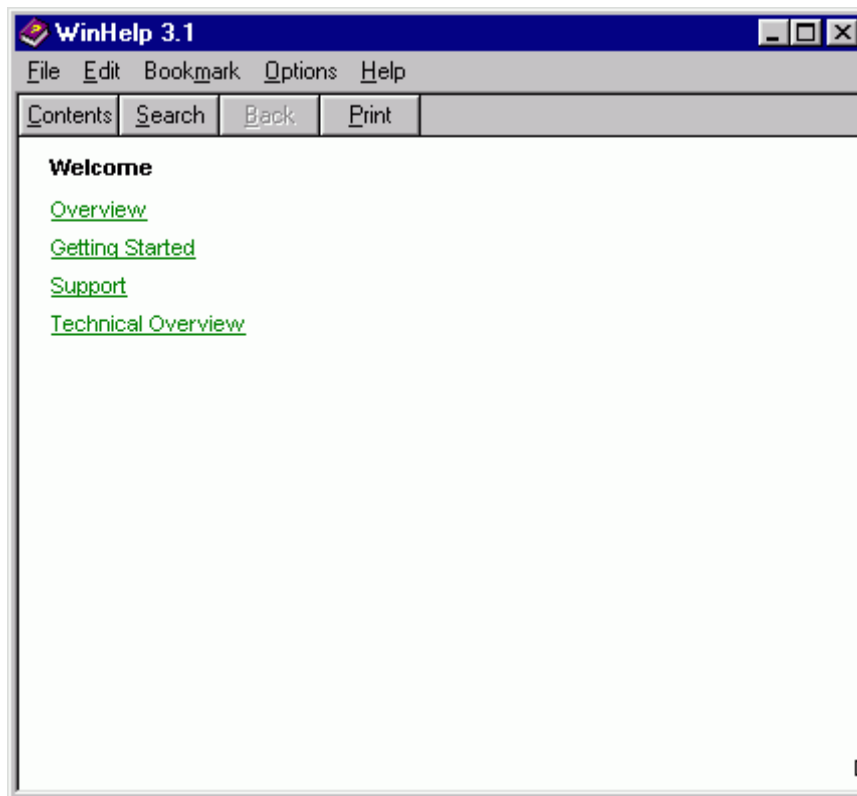
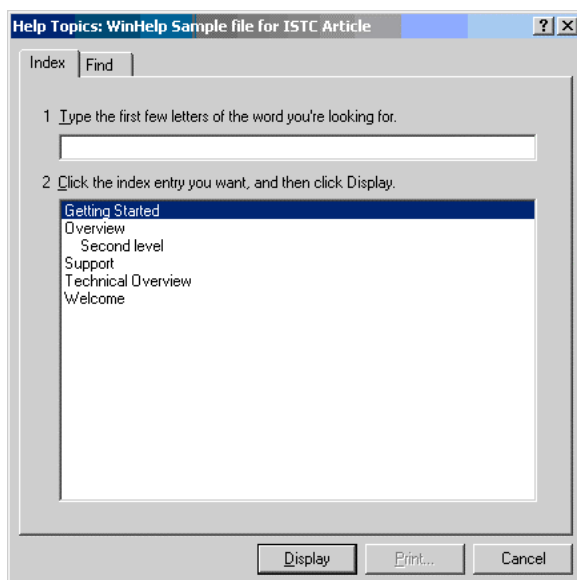
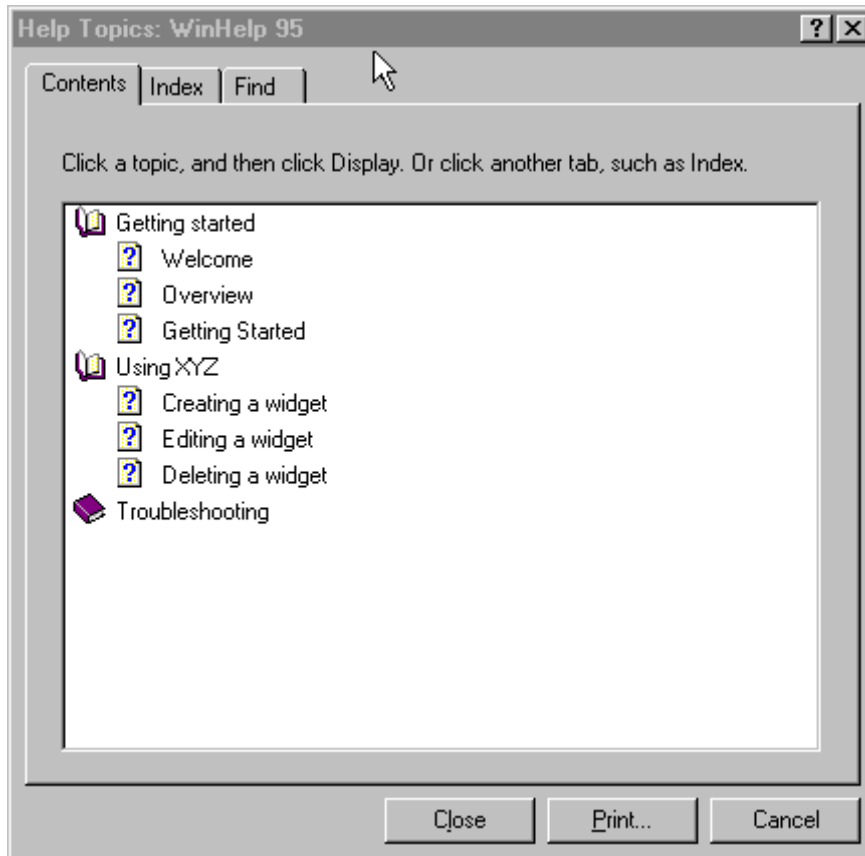


Figure 2: WinHelp 3.1 index



WinHelp 4 (also known as WinHelp 95 and sometimes as Classic WinHelp) shipped in 1995 with Windows 95® and Windows NT® 3.51. The most obvious change was the inclusion of an expanding table of contents.

Figure 3: WinHelp 95 table of contents



Both versions of WinHelp were compiled from RTF files and viewed via a custom viewer:

- WinHelp.exe (Windows 3.x) or
- WinHelp32.exe (Windows 95).

Microsoft announced in 1996 that they would cease development on WinHelp and start development on HTML Help.

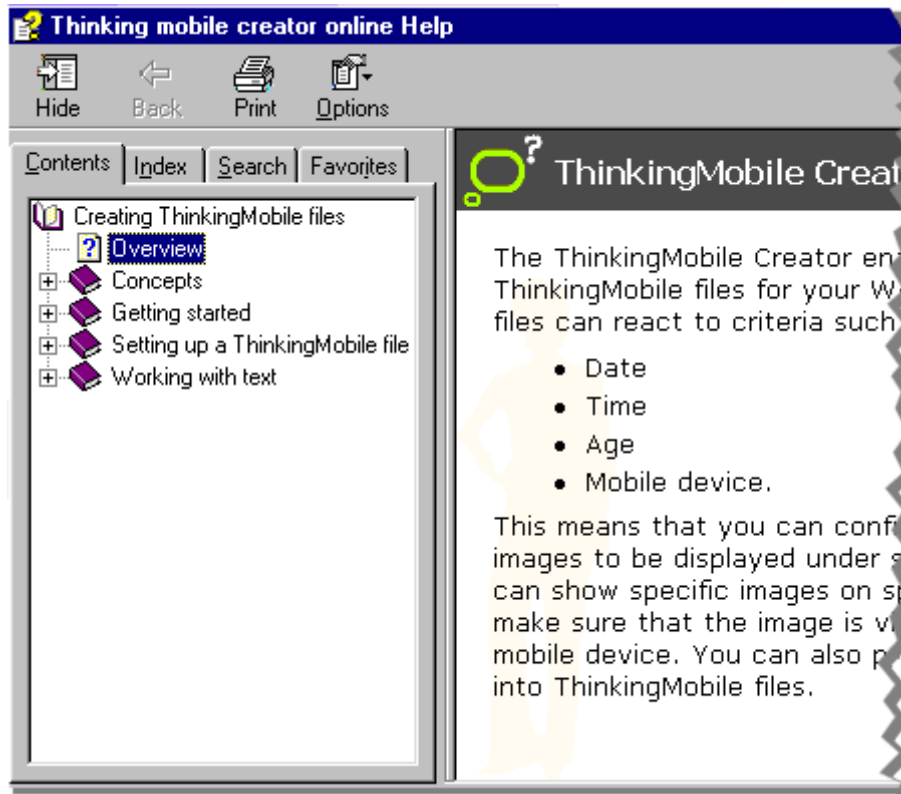
The WinHelp story was not quite over however. The eHelp modification known as WinHelp 2000® provided a much improved tri-pane view very similar to that described below under HTML Help.

HTML Help

Microsoft HTML Help, the ‘replacement’ for WinHelp, shipped initially with Internet Explorer 4® and was then released with Windows 98® in February of that year. HTML Help implemented changes both in the user interface and in underlying technology.

The standard HTML Help interface takes the WinHelp ‘Help Topics’ view and moves it up a step in terms of usability. WinHelp 95 displays this view in a separate window obscuring the content of the topic. It is also invoked by a button with the less than intuitive caption: ‘Help Topics’. By contrast the HTML Help navigation pane (containing TOC, index and search) is displayed in the left hand pane of a tri-pane view. Users don’t need to find a button to turn on navigation, and it can be viewed simultaneously with the topic content.

Figure 4: Microsoft HTML Help



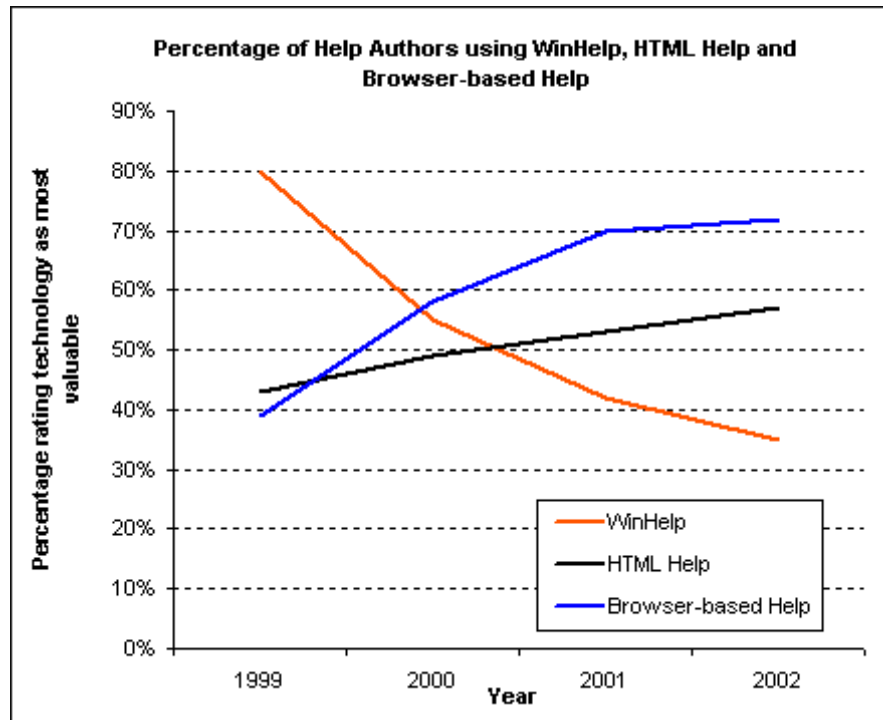
In terms of technology, the big change here is a move from RTF files to HTML. This change opens up Web-like formatting and interactivity options to Help authors. Styling can be achieved using cascading stylesheets (CSS) and dynamic effects can be coded using dynamic HTML (DHTML). HTML Help is currently at version 1.4 and that would seem to be the end of its development. In 2001 at the WinWriters Online Help Conference Microsoft announced a new Help technology Microsoft Help 2.0 (Martin 2003). More on this later, but between 1998 and 2001 a new development in Help took place which seemed to my mind to reduce the significance of Microsoft's latest developments and to move the focus to a new arena: the Web.

Web-based Help

Interestingly, despite its release in 1998, HTML Help has been surprisingly slow to be adopted. Some of this can perhaps be accounted for by a relatively slow drop off in the use of WinHelp. However whilst WinHelp dropped from 80% usage in 1999 to around 35% in 2002. HTML Help has only risen from 43% to 57% over the same period. Some of this discrepancy may be accounted for by the marked increase in interest in Web- (or browser-) based Help which rose from 39% in 1999 to 72% in 2002.

(1)

Figure 5: Help authors using WinHelp, HTML Help and Browser-based Help.



As the Web grew in popularity it began to be used as a medium for accessing applications and for disseminating user assistance. Help distributed over the Web resides on one central server. Updates to the Help therefore don't need to wait for the next release of the software. In addition, it is possible with centrally- stored Help to analyse usage statistics: we can see which topics are read and which aren't.

It's not all good news however, delivering Help over the Web brings with it certain specific problems. These problems for the most part boil down to the heterogeneity of the Web. In other words, we don't know:

- What operating system, or
- What Web browser our users are using to view the Help.

HTML Help is not designed for this new uncertainty. HTML Help:

- Cannot be used over an HTTP connection
- Will only work on the Windows Operating System
- Requires Internet Explorer and cannot be viewed using Netscape browsers.

As mentioned above, Microsoft create Help formats for their own products not for the Help community at large (Martin 2003). As a result, the development of a Help system for cross-browser applications or for applications running under UNIX® is unlikely to be one of Microsoft's primary goals.

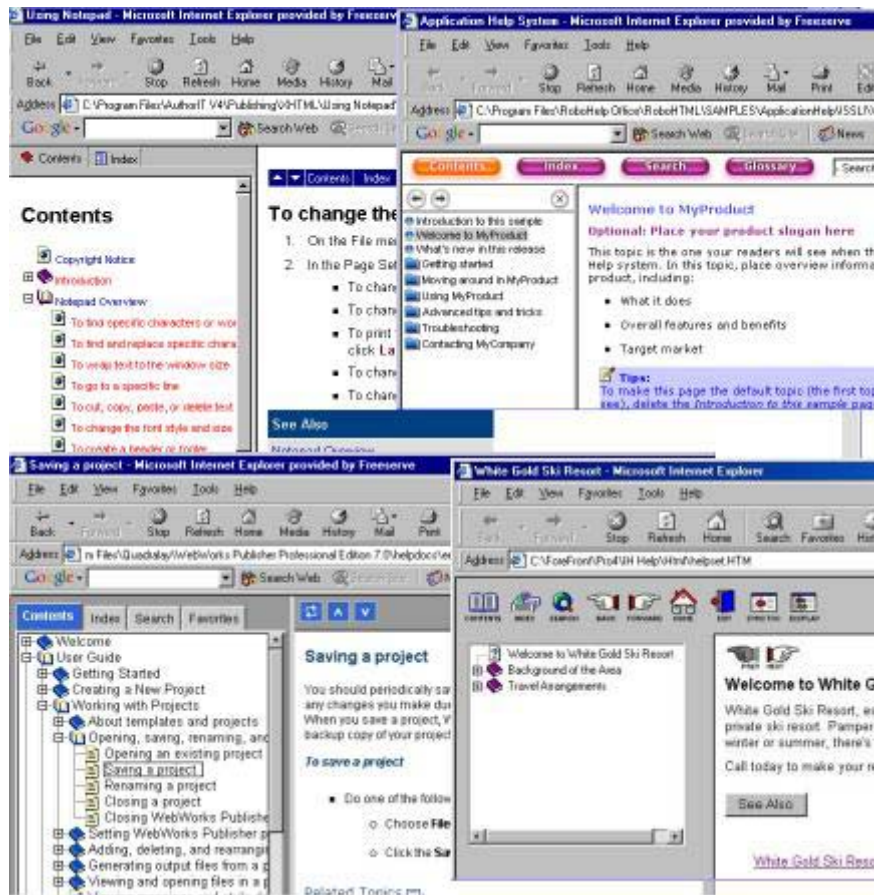
In the absence of a Microsoft solution, the focus of Help format development has moved towards the developers of Help authoring tools. Quadralay®, eHelp®, the late lamented ForeHelp® and AuthorIT™, have all developed and released Help formats that (unlike HTML Help) will run:

- Over the Web
- On more than one operating system (often referred to as cross- platform)
- In more than one browser (often referred to as cross-browser).

These formats deal with the browser issues above in several, differing, ways, including the use of Java and JavaScript to provide tables of content and indexes. In addition, both Sun® and Oracle® have produced Java™-based cross-platform Help formats (JavaHelp™ and Oracle Help for Java respectively). These formats are however limited in terms of functionality. JavaHelp, for example, limits even simple features such as selecting a font and will not support useful advanced features such as Dynamic HTML or image maps.

Microsoft, therefore, no longer sets the pace of Help format development to quite the same extent. A situation that has perhaps been re-enforced by the tale of Help 2.0.

Figure 6: A cornucopia of Web-based Help formats. Clockwise from top-left: AuthorIT XHTML output, eHelp’s WebHelp®, ForeFront’s InterHelp (now owned by Component One), Quadralay’s WebWorks® Help.



(1.) Statistics from WinWriters surveys (Welinske 2000, 2001, 2002). Interpretation author’s own. Respondents were asked to: ‘rank the importance to your current development efforts’ of each of several ‘delivery technologies’. The figures quoted are for the percentage of respondents who rated these technologies as 4 or 5 on a scale of 1 to 5.

Help 2

The story of Help 2.0 is an odd one to say the least. In 2001 at the Winwriters Conference Microsoft announced the release of a new Help format: Help 2.0. This new Help format, however, was never to see a full public release. This year, again at the WinWriters Conference, Cheri Lockett Zubak announced that Help 2.0, whilst still being used for Visual Studio® and .NET®, would not be publicly released. So what are Microsoft’s future plans? It seems that those waiting for a new Microsoft Help format may need to wait until the release of the next Windows® operating system, code-named Longhorn. (Martin 2003)

Longhorn

So what do we currently know about Microsoft's future Help plans? Here's what seems to be trickling along the grapevine:

- Longhorn may not be released until 2005
- Internally, Microsoft is using an XML schema based on DocBook
- Microsoft will be moving towards task-based assistance

Microsoft is moving towards an 'inductive user interface' (an interface structured around tasks) with embedded user assistance. (Martin 2003)

New Help formats

The Help World is not standing still. New Help formats are being developed. OmniHelp and FlashHelp™ are two of the newer formats approaching the problem of cross-browser, cross-platform Help support from very different directions.

OmniHelp

OmniHelp is a new, open source, Help format development.

Open source, put simply, means that the Help format is available free to use, provided that the source code, with any modifications, is made available for others to use and modify. In other words this is a totally non-proprietary system that you can obtain and modify to your heart's content, as long as you extend the same privilege to everyone else (1).

Open source also means that the format is developed collaboratively. Although currently all work has been carried out by staff of Omni Systems, Inc. they welcome collaboration from others: particularly Help authoring tool vendors.

Technologically, OmniHelp is interesting and seems to be aiming to keep the technology involved simple and transparent. Apparently the only code needed to integrate a standard HTML file with an OmniHelp system is one line referencing a JavaScript® file. The whole system is made up entirely of XHTML and JavaScript files.

In terms of design, two things stand out. One is the (unfortunately) slightly dated and clunky look and feel. The other is the lack of an expanding and contracting table of contents: something seen in virtually every Help format. This functionality is replaced by a choice between a "Full", "Medium" and "Short" table of contents.

Figure 7: OmniHelp



The developers are very up-front about this design feature:

- Unlike WinHelp and HTML Help, the OmniHelp Contents list does not expand and contract; it is always fully expanded, so that every entry is an immediately clickable link. This design concept is carried through consistently in the other navigation methods. We consider this a major OmniHelp benefit. (OmniHelp Design Report .)

I'm not sure I agree. The expanding and contracting table of contents is now a given in Help design. Of itself that doesn't make it a good thing: but to remove a navigation feature that pretty much every user of Help will be familiar with and able to use requires:

- Solid evidence that the standard table of contents doesn't work,
- A convincing replacement.

In my opinion, a full table of contents (in other words fully expanded) will be too long and overwhelming for users. The OmniHelp alternative: choosing a "Short" or "Medium" version, while an interesting idea, will, I think, confuse users who have not come across it before.

That said, as the developers themselves point out:

- OmniHelp is very much a work in progress, and all those interested are cordially invited to participate in its development. (OmniHelp Design Report .)
- OmniHelp is currently only supported by one Help authoring tool: Mif2Go. For the more technically minded, however, it would seem to be possible to create OmniHelp by hand. In the future more vendors may begin to support it. The OmniHelp developers think this would make economic sense:

Participation in this open-source project provides significant benefits for authoring-tool vendors. The lack of an open standard for cross-platform Help has meant that major resources had to be assigned to the task of creating a proprietary solution. For some of the smaller vendors, that cost has been prohibitive, and so they have not created such a cross-platform solution. But even for larger vendors who have done this, the costs of code maintenance and of updates to deal with new browser and OS versions are burdensome. OmniHelp offers all vendors the opportunity to work together solving common problems, while still preserving their own proprietary value-added in the way they generate the project- specific OmniHelp files. (OmniHelp Design Report .)

Personally, I wonder if the benefits of providing a proprietary solution that can only be produced by one tool outweigh the costs of developing and maintaining such a solution. eHelp are certainly willing to put in the effort to create proprietary Help systems as we can see in the following section.

(1) It is the underlying, OmniHelp, technology that must be made available (including any modifications) to others to use and modify. To the best of my knowledge this shouldn't affect the copyright of the content.

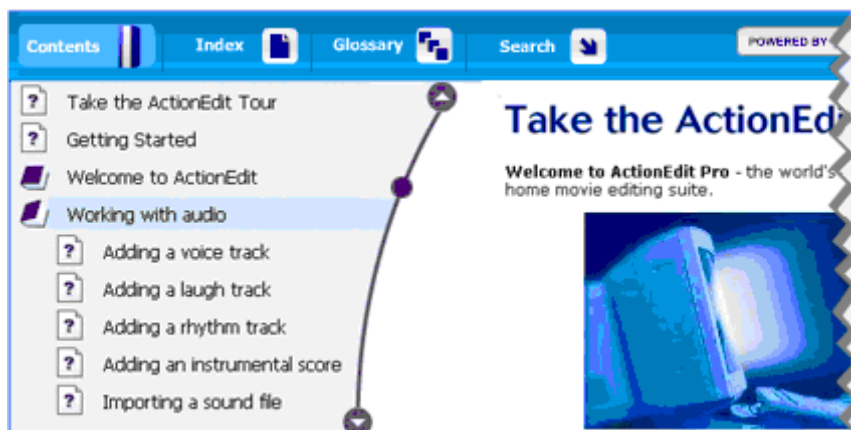
New Help formats

The Help World is not standing still. New Help formats are being developed. OmniHelp and FlashHelp™ are two of the newer formats approaching the problem of cross-browser, cross-platform Help support from very different directions.

FlashHelp

In sharp contrast to this simple (and I mean simple as a compliment) open source solution comes the latest offering from eHelp. FlashHelp is marketed on its apparent ability to provide a consistent look and feel across multiple platforms and browsers. It also claims to be quick to download, to avoid the security issues associated with navigation provided by Java and to be highly customisable. Combined with the possibility of animated and even interactive interface elements, it certainly sounds like an interesting development.

Figure 8: FlashHelp (still taken from a Flash® demonstration at www.ehelp.com)



If all of the claims are true (and FlashHelp has only recently become available, so time will tell) then FlashHelp may very well be a very good alternative to the current range of Web-based Help formats. It would seem able to do away with worries about browser and platform differences (currently Help systems can look significantly different between older versions of Netscape® and current versions of Internet Explorer for example) and may offer a quicker download. Navigation that does away with Java and JavaScript would also provide a significant improvement.

Customisable interfaces have their place and provide the opportunity to create Help that is aesthetically integrated with the software. It is important however that Help authors do not become over-enamoured with the possibilities. Help files are often used in a hurry and should provide answers as quickly as possible. Users will often not have the patience for graphically rich movies and intros and will, I suspect, become quickly infuriated with flashing icons and the like.

Flash is often criticised on the following fronts:

1. It is a proprietary solution
2. Users need a client-side installation
3. There may be problems with accessibility.
4. It is expensive to produce in both time and money
5. It is often used inappropriately (to quote a friend of mine: 'forcing you to use the navigation system that came from some wacky cartoonist-wannabe's diseased brain').

Of these points, only number three gives me cause for concern. On the others:

- Microsoft HTML Help is a proprietary format: if that doesn't bother you, then neither should FlashHelp.
- According to Macromedia®, 'Macromedia Flash Player® is currently available to over 98% of web users'.(1) This is not perfect, but if you feel that 98% is high enough, or you are confident that your users have Flash installed, then FlashHelp should be a viable option.
- With the caveat that I have not yet been able to experiment with FlashHelp, I see no reason why FlashHelp (produced without too many custom features) should be much more time consuming to produce than RoboHelp®'s other outputs.
- We, the Help authors, have replaced the cartoonist! FlashHelp's navigation conforms to the standard tri-pane structure with a table of contents, index and search. The structure of these features should be in the hands of technical authors.

It remains to be seen how FlashHelp will deal with accessibility concerns. I will be interested to see how it fares on this score. Flash has been heavily criticised on this point, but, with the release of Flash MX® and the Flash 6 player, accessibility features such as alternative text have been introduced (Perry 2002). If eHelp have taken advantage of these features then FlashHelp may well be at least as accessible as WebHelp, eHelp's current Web- based Help format.

(1.) I have a concern here over the word "available". I hope this means that 98% of users have Flash installed. It could be taken, however, to mean that 98% of users can download it if they want to. Clearly this is quite different and not satisfactory. However, according to Joe Welinske: 'A survey conducted by NDP Research showed that 98% of Web users can experience Macromedia Flash content without having to download and install a separate player.'
(<http://www.winwriters.com/flash.htm>).

So which format is right for you?

I hope the above is a useful synopsis of some of the major developments in Help over the last few years. As you can see, quite a few different formats have come into being over the last decade or so. It is both interesting and problematic that most of the Help formats above are still very much with us. I will admit that I haven't seen a WinHelp 3.1 system written after 1997, but Help files are still being written using WinHelp 4 (I last taught a course on doing so in March of this year). One reason for this tenacity is that software developers can be unwilling to alter the context-sensitive calls in their application to interface with a different online Help format.

This persistence of Help formats combined with the lack of a single standard format for Web-based Help leaves us to choose between 10 or more formats in which we could present our online Help. So how do you choose?

It would take a whole article to list all of the pros and cons of each of these different versions of Help, but there are some fairly simple pointers, which should get you looking in the right places:

1. I would be loathe to suggest that anyone starting a Help system from scratch used WinHelp. Aside from several improvements in functionality provided by most other Help formats, to start writing in WinHelp now is an investment in an outdated technology. The technical skills gained are not as transferrable as the Web-based skills behind other Help formats, some Help authoring tools have already (tacitly or openly) moved away from supporting WinHelp and it is becoming difficult to find information to help with any problems you may have. To my mind these considerations are at least as important as the differences in capabilities between the systems.

Having said that - WinHelp is a good Help format, particularly when implemented as WinHelp 2000 using the eHelp modifications. If you currently have a WinHelp system and:

- Your developers are not willing to change the context-sensitive calls, or
 - You lack the time or resources to undertake a conversion to a new format
- then sticking with WinHelp is far from the end of the world.

2. There are 2 or 3 factors dictating the choice between Microsoft HTML Help or a cross-browser, cross-platform solution. If:

- Your users may not be using Windows
- You need to provide the Help over the Web

then compiled Microsoft HTML Help is, essentially, not an option. You will need to consider one of the several cross-browser, cross-platform solutions. If these two points are not an issue for you, then you are at liberty to use Microsoft HTML Help. You may well find that doing so will limit some possible complications.

However, even if you are providing locally installed Help to Windows users, you may still want to look into a Web-based solution. You may find them more flexible, at least in terms of look and feel.

3. If you have decided on a Web-based solution, the difficult decision is: which one! Listed below are some of the points you need to consider:

- Which Help authoring tool do you want to use? As many of the Web-based solutions are tool-specific this is a fundamental decision.
- Do your users have any technology or security restrictions? Some organisations restrict the use of JavaScript and Java. These technologies are often used to implement navigation in Web-based systems. If your users, for example, cannot download Java applets (usually for reasons of security), you need to be sure to pick a solution that doesn't rely on Java.
- Which browser do your users use? 'Cross-browser' doesn't necessarily mean 'works on all browsers'. More often, it means that certain named browsers are supported. You need to ensure that your users' browser is on the list.
- Unless your application is coded in Java, I would advise against either of the two Java-based options: they are more limited in functionality than some of the alternatives, and visually less attractive.

Conclusion

In conclusion, it seems to me that the main feature of the Help world today is its variety. We have a choice between 10 or more Help formats ranging from the 8-year-old WinHelp 95, through to OmniHelp and FlashHelp: still in their infancy. This leaves us with some hard thinking to do when it comes to choosing a format. I hope I've managed to shed some light on what's available. You have plenty of options. In brief, you need to:

- Research these formats thoroughly
- Find out as much about your users as you can
- Try to match the solution to the needs of your users.

References

- Martin, C (2003) Microsoft Takes Its Help Technology in a New Direction. [Internet] Seattle. WinWriters. Available from:<<http://www.winwriters.com/resarch.htm>> (sic)
- Northern California STC WinHelp SIG WinHelp Survival Kit [Internet] Available from:<<http://www.geocities.com/Area51/6793/helpsurv.htm>>
- Welinske, J (2000) 2000 WinWriters Skills and Technologies Survey. [Internet] Seattle. WinWriters. Available from:<<http://www.winwriters.com/resarch.htm>> (sic)
- Welinske, J (2001) 2001 WinWriters Skills and Technologies Survey. [Internet] Seattle. WinWriters. Available from:<<http://www.winwriters.com/resarch.htm>> (sic)
- Welinske, J (2002) 2002 WinWriters Skills and Technologies Survey. [Internet] Seattle. WinWriters. Available from:<<http://www.winwriters.com/resarch.htm>> (sic)
- Ulli, M (2003) Historische Entwicklung der Microsoft online Hilfesysteme. [Internet] Available from:<<http://www.help-info.de/de/welcome.htm>>
- Chandler, R (2003) HTML Help - Tech Info. [Internet] Melbourne. The Helpware Group. Available from:<<http://www.helpware.net>>
- OmniHelp Design Report [Internet] Available from:<<http://www.omsys.com/>>
- Perry, JM (2002) Flash MX Accessibility Issues. [Internet] O'Reilly. Available from:<<http://www.oreillynet.com/pub/a/javascript/2002/11/08/flashmx.html>>
- Northern California STC WinHelp SIG WinHelp Survival Kit [Internet] Available from:<<http://www.geocities.com/Area51/6793/helpsurv.htm>>