

## Migrating Win 3.1 Help to Win 95/NT

By James Hobart, President, Classic System Solutions

May 1996



Moving your help system from Windows 3.1 to Windows 95/NT involves several steps that take into consideration content, search, and programming issues. This is especially true if you are going to take advantage of the new features provided with the Win 95 help engine.

### **What is Win 95 help and how can I best use it?**

The formal name is WinHelp 4. It is the primary method for developing both Windows 95 and Windows NT help systems. Essentially, WinHelp 4 provides several multimedia features, previously available only with add-on products like Multimedia Viewer™. It also provides a much more robust keyword search engine. Windows 95 fully supports Windows 3.1 help files without any rework or recompilation. The help files migrated straight from Windows 3.1 will essentially look the same; however, they will have the added functionality of the help browser index and find features. You may want to consider moving to WinHelp 4 in three stages.

### **Stage 1: Move to Windows 95/NT, and don't change the help file.**

To configure your Windows 95 help application to use the new features, you need to create a contents file (.cnt) that accompanies the help file. Using the contents file, you can add the following features to a Windows 3.1 help file without recompiling.

- Display a hierarchical 'books and topics' contents page in the help browser
- Dynamically include contents from other help files and seamlessly integrate help files for each user at run-time. Hint: Use Include statements in the contents file to insert the contents of another file into the current help file. Only the contents of installed help files will appear.
- Combine keyword indexes from multiple help files to create a "master" index. This is useful if you have several help files and want to give users just one place to search.
- Group topics to take advantage of the new built-in multiple topic printing feature. This allows users to print out the entire help file without resorting to the source document.

### **Stage 2: Move to Window 95/NT, and create associative links and secondary windows.**

You can display each topic in its own secondary window. This can be useful to compare pieces of information. However, it can be confusing to some users who are not comfortable with a multi-window environment. This approach is best used when you migrate your help content from reference material to more clear and concise task-based content. With this approach, you can provide task information very quickly in the main help file and provide detail to support the task as a topic in a secondary window. In addition, you can add associative links that appear in a secondary window. These associative links are excellent for providing the ability to search for topics across help files.

### **Stage 3: Move to Windows 95/NT, and tie online help more closely to the application.**

Classic System Solutions has always been a fan of clear and concise help within the application. Now there are some decent tools that come with WinHelp 4 to implement this technique. What's This? is pop-up context-sensitive help that appears when the user right-clicks on a specific dialog box control. This requires some significant rework of your existing help file; however, it provides clear context-sensitive help without launching the user into a cumbersome help application. You can also implement "shortcuts" that invoke a dialog box or window from within the application by clicking a button in the help window. This is a little tricky to implement, and you need to be very clear on whether the window invoked is modal or modeless. Essentially, this is the next logical step of implementing actions such as "Do it!" buttons within your help application.

Whichever approach you take, plan out your strategy, and become familiar with the features WinHelp 4 has to offer.

#### ***About the author:***

James Hobart is an internationally recognized user interface design consultant based in California, USA. He specializes in the design and development of large-scale, high-volume client/server and web applications. He is an expert in GUI design for transaction processing systems and strategies for migration to thin-client graphical user interfaces. He can be reached at [jimh@classicsys.com](mailto:jimh@classicsys.com)

Find more articles on Usability at [www.classicsys.com](http://www.classicsys.com)