TRUEyesnononoyesyesyesyesyesyesThis file © Edward Guy, CDP 1993/4Help for HELLLP!yesyesyesMTS001helllp!

Table of Contents

The items below are intended as PopUp boxes.

The Up Arrow button	Creating a Topic The T button	Creating a Hypertext Jump The J button
The Down Arrow button	Setting the first screen to display The I button	Creating a Pop Up Box ("Definition") The P button
The Curved Arrow button The Spectacles button	Browse Sequence The B button	Context Labels (Context Strings) The C button

DISCLAIMER
Acknowledgments
Hypertext
Opening Window Size
Topic Window Size
SHED incorporation PopUp
Pop Up Box for Opening Title
Question Mark Button
Test Sound
Include Sound
Exit Sound
Change Topic Title 1
Change Topic Title 2
Change Topic Title 3
Change Topic Title 4
Change Topic Title 6
Welcome to the world of HELLLP!
size of secondary windows
Secondary Windows (example)
Customising lines in the HPJ file (for experts)
Secondary Windows
Strange Coloured Window
Background Colours
SideBar Table of Contents
Choosing an Icon for the help file
Upgrade Policy
Title Bar Text Setting
Pop Up Box

Help file produced by **HELLLP!**, a product of Guy Software, on 06/03/94

Close Exit

SideBar Table of Contents

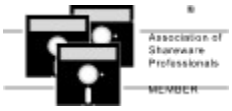
(An example of a secondary window)

The Windows™ Help Authoring Tool

Takes care of all the drudgery and makes authoring Windows Help Files an enjoyable and rewarding experience.

Allows you to use color (or colour) to make sure that text you wish to emphasize gets read by the users.
Different sizes and fonts may be used to give additional visual interest.

To get help on the following topics, just click the mouse pointer on the text that is THIS COLOUR.



Getting Started With HELLLP!

The setup sequence is critical for successful use of the system. You might want to print this topic for reference (click the Print key on the button bar).

1. Set up a subdirectory for your project files, we suggest C:\HELPCOMP\PROJECT.
2. Install the Microsoft Help Compiler - for a source of this see below. Note its fully qualified pathname (e.g. C:\HELCOMP\COMPILER\HCP.EXE).
3. Install the template file HELLLP!.DOT and the help file HELLLP!.HLP in the same subdirectory as your other Word for Windows templates.
4. Within Word for Windows select File, New and choose HELLLP! as the template.
5. Respond to the Setup questions on first use.
6. When prompted either type your help text or import it with Insert File.
7. Ensure that you can see Word 2.0 pattern Toolbars
8. Follow the instructions in the topic "Using HELLLP! to make a Help File"

HELLLP! is designed to be as easy to use as possible, while still allowing the expert user to use it to "jump start" the creation of complicated help file projects. For a simple project you will probably only need to accept its suggestions in the choices it offers you in order to achieve success.

HELLLP! is all contained in the template **HELLLP!**.DOT which you copied to the directory in which you keep your templates, and it has its own HELP file which you are reading now, and which may be invoked from the Help menu item, the Help button on the initial setup screen and the Question Mark button on the toolbar.

If you are storing your templates on a shared network drive, please review the topic "HELLLP! from a Network Server".

In order to do its job, though **HELLLP!** does need to find out a few things about your system and your preferences.

During initial setup, **HELLLP!** asks for a directory name in which to save the files it creates during its operation.

To actually produce a HELP (HLP) file, **HELLLP!** needs access to one of the Microsoft™ help compilers. These have names like HC.EXE or HCP.EXE, or HC30.EXE or HC31.EXE or something similar.

One of these compilers is included in most tool kits sold for producing Windows programs. They are also available from the Microsoft software library on CompuServe. The HC300B version of the compiler (which writes HELP files for Windows 3.0) was available as the file S13290.ZIP and the HCP version (uses extended memory for speed, and writes for Windows 3.1) was file S13436.ZIP in August 1993.

If using version 6.0 of Word for Windows you will need a version of the help compiler which is compatible with RTF files produced by that version. As of January 1994, a suitable version of the compiler was available for download from the Microsoft Windows SDK forum on CompuServe (GO WINSDK) as HCP505.ZIP in the WinHelp section.

During the setup dialog, you are asked for the name and location of the compiler you wish to use. Please provide the full path name e.g. **C:\HELPCOMP\COMPILER\HCP.EXE** if that is what you are using. If you do not have a copy of the compiler, **HELLLP!** will still work, but only to the point of producing the input files that the compiler needs. You may take these to a computer that has the compiler on it.

If you later wish to change any of your initial settings, for example to use a different version of the Microsoft Help Compiler, you may do so with the "Change HELLLP! Setup" option on the Format menu. If you change your project directory but wish to work on an existing project you should move **ALL** files associated with it (they all have the same filename but different extensions) to the new project directory.

There will be files with various extensions in your project directory, all with the same name as your project but with different extensions. They are all maintained by **HELLLP!** and you should not edit them unless you are an expert. You can do everything

that you need from within **HELLLP!** using the *.DOC files.

If you are using an unregistered (that is not paid for) version of **HELLLP!** you will get a registration reminder popping up when you open a file produced by **HELLLP!**. You are entitled to 30 days free trial before either paying for it or ceasing to use it. For details of how to do this see Registration and Payment for **HELLLP!**.

The help file you are reading was built using* HELLLP!*, so anything it does, you can do! In fact some of the things in it were done to demonstrate the effects you can achieve.

HELLLP!

from a network server

You may use **HELLLP!** from a network server provided you take the same steps as you would with any other template located on the server and provided you pay for the correct number of users after your 30 day free trial period.

You will not be able to make any changes to the template while someone else is using a document attached to it.

This means that you should only do the following when no-one else can access the template:

Change registration information
Change the Help compiler name or location
Change the project directory path

since these all require the template to be re-saved and this cannot be done when two users have it open in shared mode. Of course they may be performed only from a user ID which has write access to the template.

The full path name of the compiler must be one that every user can access, as must the project directory path. The users do not need to access the same directory, just one with the same name. e.g. if the project directory path name is C:\HELPPROJ then every user machine would need to have a \HELPPROJ subdirectory on its C: drive.

Using HELLLP! to make a HELP file

Windows Help presents "TOPICS" (or Screens) to the user based on searches, the browse buttons (**<<** and **>>**) and jumps. To create these screens, various codes have to be inserted into the file containing the text. **HELLLP!** creates and inserts these codes for you. If you are an expert, you may use **HELLLP!** to do a "first cut" and then fine tune the result yourself before compiling. For most people, **HELLLP!** will do it all for you with no hassle.

The process is controlled by the lettered buttons on the toolbar.

To ensure that everything is set up properly, it is necessary to create a file by doing a FILE NEW and choosing HELLLP! as the template. Once that file is created and saved, you may open it like any other file. If you are working with an existing document file, you will be prompted to insert it at the appropriate place. If you just attach your document to HELLLP!.DOT using FILE TEMPLATE, it will not work properly.

To turn a section of text in your file into a "topic" which will become a screen, just place the cursor (the "insertion point" as the MSWord documentation calls it) in the first line you want included and press the T button. You will be asked for a title for the topic, and will also indicate via a check-box whether you wish the title to appear emphasized at the top of the topic screen. If there are words in that first line which you would like to use as the topic title, select (highlight) them before you press the T button and **HELLLP!** will present them as a proposed title, which you may then edit (e.g. to capitalize letters).

If you intend to set non-scrolling topic titles, do not ask HELLLP! to put the emphasized title on any topic which you intend to use as a PopUp (if you do, only the title will pop up).

Notice that **HELLLP!** inserts a page break ahead of the topic. This is the only circumstance in which there should be a page break, since a page break is what defines a new topic. Text following a page break that is not set up as a topic will never be seen by your users.

HELLLP! will also take the words of more than one letter in the title and present them as proposed "keywords" to be used in WinHelp searches. You may edit these or add or delete words.

HELLLP! also takes care of "Context Strings", the means by which the Help Compiler identifies topics. **HELLLP!** will create a unique context string for each topic and keep track of it. You do not need to remember it because when you set up your Hypertext jumps, you will be selecting from a list of topic titles.

You will always be given the opportunity to override the automatic choices which **HELLLP!** suggests.

Go through your text in this way until you have marked all of your topics. At this stage, you may compile your help file and it will work. It will have a single table of contents with all of your titles listed and you may jump to them as with any other help file. You may test your newly compiled HELP file by clicking on the "Spectacles" button on the toolbar.

However, to make the file really useful you will probably want to go through it again and add Hypertext jumps and PopUp boxes using the J and P buttons. Although the same topic can be reached by both PopUp and Hypertext is unwise to do this, since there may be side effects. In particular, if you have HELLLP! incorporate a title at the head of a topic to be used in a PopUp, and then set non-scrolling topic titles only the title will pop up!

You have the option to set up "Browse Sequences". This just means that the **<<** and **>>** buttons will be able to page your help file from topic to topic.

For more sophisticated applications, you may create your Help file in several sections, you may assign context numbers for use in other applications (called HelpContextIDs in Visual Basic), you may add sound effects and you may start other processes from your Help file to give your users a true **Multi-Media Help Experience**.

Tables of Contents

HELLLP! generates a table of contents in the sequence in which topics are identified. It uses the names used during topic definition and inserts the Hypertext Jumps to them automatically. This table is used by **HELLLP!** to keep track of all of the context labels (known in the Microsoft manuals as context strings). You are taken to this table during the generation of hypertext jumps to pick the place you want to jump to, or the text you want in a PopUp box.

It is recommended that you minimize any editing changes you make to this table of contents. You may move items around, change the indentation, the size, the font color but do not change the text as to underlining or hidden/not hidden because these will prevent the compiler from working. You must also move entire lines or groups of lines, not parts.

The standard table of contents will appear in your help file unless you override it with the **I** button. You may include your own table of contents, just like a regular page and have it displayed when Windows Help is invoked.

Context Labels (Context Strings) are the means by which topics are identified. **HELLLP!** generates them automatically, guaranteeing that they are unique. However a user has the opportunity to override the automatically defined label.

HELLLP! uses a three letter/three numeral scheme taking the initials of the first three words in the title (padding with underscores if there are less than three words) and adding a three digit number to identify topics with the same three initials. The number starts at 001.

The automatically generated table of contents is TOC000.

To ensure that context strings are unique for multiple project file builds, the project filename (less any characters not legal in a context string) and a period are prepended to the string (for example this topic would have a context of **HELLLP.CLC001**).

Although not needed by the HELP compiler, **HELLLP!** inserts the contexts as hidden text in dark red ahead of the topic title (if any) or text for the convenience of those users who need to know their values.

The automatic table of contents generated by **HELLLP!** provides a ready cross reference list of topic names and their context strings.

Another list of the context strings is maintained in an INI file, for use by those who need to ensure that particular values of context string are not used with the current file (some applications are produced from multiple files). This INI file provides the means to "busy" the context strings to avoid their re-use and is also used by the External Jump feature of **HELLLP!**

The symbols +#!K$ appearing at the start of each topic are codes used by the help compiler and should not be edited unless you **really** know what you're doing.

The HELLLP! Button (Tool) Bar

You may click on one of these buttons for help about its function.
The **HELLLP!** button bar is a customized version of the regular Word For Windows 2.0 toolbar with some buttons added in place of those you are not so likely to use in making a help file. The actions commanded by the ones omitted may still be performed by the pull down menus. If you cannot see these buttons on your toolbar, please ensure that you have **Word for Windows 2.0** selected as your toolbar.

Where the user has options associated with the buttons on the button bar (tool bar) they are presented after the button has been clicked.

The Up Arrow button and The Down Arrow button move you quickly to the table of contents, or to the last place in which you added a topic, jump or PopUp.

T Will create a "Topic".

J Will create a hypertext jump from the highlighted text to a topic of the user's choice.

P Will create a PopUp box containing the text of the desired topic.

C Used only in conjunction with the P and J buttons. Used to identify the objective of a hypertext jump or PopUp box.

The Speaker Button incorporates sound into the topic.

The Screen Button permits an external process to be started by the topic (e.g. multimedia effects).

I Used to determine which topic will be displayed first when the file is opened by Windows help.

B Will create automatic browse sequences.

The Curved Arrow button ("Give it a whirl") prepares the files for the compiler, then calls the compiler to produce a help file that you can test.

The Spectacles button lets you "take a look" at your newly compiled help file.

? Brings up the Help screens for **HELLLP!** (the ones you are looking at). An alternative to using the menus.

T Will create a "Topic", which involves generating a topic title from the text currently highlighted (or entered from the keyboard if nothing is highlighted). The title will appear as a result of a Windows help search whenever any of the words of more than one letter in it are entered as keywords. The title will also be used in **HELLLP!**'s automated table of contents. The user is given the option to edit the automatically generated keywords, adding new ones or deleting any not desired.
The user has the option to have the title typed or selected included at the head of the topic in the help screen, in addition to the table of contents and search screens. When the box selecting this option is checked, the title will appear in the style selected as Topic Title in the style box. The topic title style may be changed using the **Change Topic Title Style** function on the **Format** menu, which includes an option to make the topic title non-scrolling.

J Will create a hypertext jump from the highlighted text to a topic of the user's choice. After highlighting some text and pressing the J button, the user is taken to the **HELLLP!** automatic table of contents to identify the other end of the JUMP. The user places the cursor in the table of contents entry for the desired end of the jump and presses the C button. All work is done automatically by **HELLLP!**
(The cursor may in fact be placed in any double-underlined text which is followed immediately by dark red, dotted underlined, hidden text, or directly in any dark red, dotted underlined, hidden text which directly follows underlined text - you are taken to the automatic table of contents for your convenience because all topics are listed there)
The J Button also provides the option to Jump to a topic in another project file using the External Jump button.

P Will create a PopUp box containing the text of the desired topic, instead of a hypertext jump to it. After highlighting some text and pressing the P button, the user is taken to the **HELLLP!** automatic table of contents to identify the topic to put in the PopUp box. The user places the cursor in the table of contents entry for the desired end of the jump and presses the C button. All work is done automatically by **HELLLP!**
(The cursor may in fact be placed in any double-underlined text which is followed immediately by dark red, dotted underlined, hidden text, or directly in any dark red, dotted underlined, hidden text which directly follows underlined text - you are taken to the automatic table of contents for your convenience because all topics are listed there)
The J Button also provides the option to Jump to a topic in another project file using the External Jump button.

Text to go in a PopUp box may also be accessed via a hypertext jump, because it is a topic like any other. It is a good idea to put text which is intended only as PopUp text at the end of the file since it may also be accessed by the browse buttons. Alternatively, text for PopUps may be placed in separate file and compiled without browse buttons. The text may then be incorporated using the External Jump feature offered as an option when the **[P]** button is clicked..

C Normally used only in conjunction with the P and J buttons to identify the objective of a hypertext jump or PopUp box. When you have placed the cursor (the "insertion point") in the double underlined title of the topic you wish to jump to or pop up, you click the **C** button to allow **HELLLP!** to process the context string. Instead of putting the cursor in the title, you may put it in the dark red dotted underlined text ahead of a topic title indicating the context string. If you wish to include a jump to the automatic table of contents you must use this method, since it does not appear in its own list. For your convenience, the **P** and **J** buttons position the screen to the automated table of contents, and the **C** button restores it to where you were.

What the button actually does: If the insertion point is in hidden text, it backs out of it. Then it looks for the next hidden text (which might be the piece it backed out of) and copies it to the location the cursor was at when one of the J or P buttons was pressed. It therefore has copied the context string for the topic to a new location after the highlighted text which became underlined or double underlined as appropriate when P or J was pressed with it highlighted. If the J or P button had not been pressed, the C button assigns the topic for context string it finds as the topic to be displayed initially.

I Used to determine which topic will be displayed first when the file is opened by Windows help. When you click the **I** button you are taken to the automated table of contents, just as with a hypertext jump or PopUp. After this, place the insertion point (the cursor) in the title of the topic you wish to display first, where it occurs in the **HELLLP!** automatic table of contents, then press the **C** button and that topic will appear first when help is selected. In that case the **HELLLP!** automatic table of contents will be displayed only if you insert a hypertext jump to it. This way you can hide it from the users, replacing it with your own in your final HLP file, but retain it for your own use in keeping the file you produced it from updated. If you change your mind you may repeat the process to select another starting page. To restore the **HELLLP!** automated table of contents as the initially displayed topic, place the cursor in its topic title and press the **C** button.

You may also display your own title screen by creating it as a topic, setting it with the **I** button to display first, and supplying just one hypertext jump from it to the table of contents, your own or the **HELLLP!** automated one.

The I button was used in this way to make the Windows Help application start with the title screen you saw when you started it this time.

If you do not make a selection with the **I** button, the **HELLLP!** automated table of contents will be the first screen to display when Windows Help calls the file.

B Used to generate browse sequences automatically. Use of this button gives the option to delete all previously defined browse sequences and to generate them for all topics except the automated table of contents. These browse sequences will enable the **<<** and **>>** buttons on the Help Screens so that the user will be able to read through the document in the same sequence as it is in the file. For any topic which you do not wish to be accessed by browsing (for example any topic you wish to appear **only** as a PopUp box), delete the **+**footnote reference at the top of the topic.

The Up Arrow button will take you to the last place in which **HELLLP!** added an automated table of contents entry.

The Down Arrow button will take you the last place in which **HELLLP!** did work in your text, adding a topic heading, a PopUp or a hypertext jump.

The Browse Sequence

The "Browse Sequence" is the order in which topics will appear when the user presses the **<<** and **>>** buttons at the top of a windows help screen.

If a browse sequence is not defined for a topic, the **<<** and **>>** buttons are disabled when that topic is displayed.

To remove a topic from the browse sequence, simply delete the + footnote reference mark ahead of the topic title (don't delete any of the other marks or the topic may not be accessible to your users).

With the use of the B button, **HELLLP!** will insert browse sequence numbers so that the **>>** button presents the screens in the same sequence as the topics in the file. The automatic **HELLLP!** table of contents is not included in the browse sequence so that a user who wishes to use a different format table of contents, selected with the I button can prevent its display to the users, but still preserve it in the file as a means of keeping track of HELP context strings.

> The sophisticated user, who wishes to use a different browse sequence, may edit the footnotes to change the browse sequence. Browse sequences are created in the form *group:sequence*. Browsing may only occur between topics in the same browse group. **HELLLP!** creates a single browse group for each project file, with the same name as the project, and appends a five digit number advancing in tens. To create more than one browse sequence in a single project file, edit the *group* in the + footnotes to create different groups.

The Microsoft Help Compiler permits a help file to be built up from several files, and allows Hypertext Jumps and PopUps between them. Some people like to keep their "Jump" topics in one file and their "PopUp" topics in another one. Some projects are large or decentralized so that sections of the help file are built separately and linked later. The External Jump feature of **HELLLP!** is designed to assist in the use of these features.

When you click on **[J]** or **[P]** to create a jump or a PopUp you are offered the option of selecting an External Jump. If you select this, you are presented with a list of other help files in the project directory. When you select one, you are then presented with a list of topics and their context strings. You select one and the jump or PopUp is set to reference that topic in that file. HELLLP! also sets a flag to ensure that the file referenced will be included when a help file is created. The initial screen displayed by WinHelp will be that for the file from which the compile was run.

If you prefer to select topics for jumps or PopUps from a list box, instead of by using "point and shoot" at the table of contents, you may use the [External Jump] button to pick topics from the current file. Just choose the current file instead of a different one from the list of help file project names.

The other files may still be worked on and updated, provided that the context strings for the externally referenced topics are not deleted or changed.

To compile your HELP file

 You may compile your help file by clicking the curved arrow, "lets give it a whirl" or by selecting "Compile the Help File" from the Tools menu. This will save the file as Rich Text Format and will also save the .DOC file. Before proceeding, you may wish to convert any "Smart Quotes" and to update the browse sequence so that the **<<** and **>>** keys page you through the text in the same sequence as the file. If you have installed one of the Microsoft Help compilers and have used **HELLLP!** setup to tell **HELLLP!** where it is, then **HELLLP!** will prepare a Windows Help (.HLP) file which may be used on any machine running Windows. If you do not have the compiler installed, simply take the files with the same names as your project to someone who does, and they will be able to compile it for you.

Compiler error messages will appear. The various versions of the HELP compiler require different commands, and will sometimes reject the command for a different version. Because **HELLLP!** cannot predict the behaviour of the particular version you are running (the name does not identify the compiler, it could have been renamed), it issues several versions of the command. For example, some versions expect the command INDEX to tell them which screen to start with, some expect the command CONTENTS for the same thing. **HELLLP!** issues both, because one will be right. However the other will trigger an error message. The error messages do not indicate a problem with your file, which you may test by clicking the "spectacles" button or by selecting "Test Project Help File" on the Help menu.

IF YOU RECEIVE A COMPILER ERROR WHICH STOPS THE COMPILE YOU MAY NEED TO OBTAIN THE CURRENT VERSION OF THE COMPILER, as explained in Error Messages.

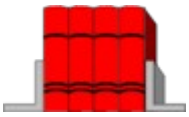
To try out your newly compiled help file

You may try out your newly compiled help file by clicking on the "Spectacles" button to "take a look" at it. As an alternative you may select the item "Test Project Help File" from the Help menu.

User Copyright Line in WinHelp ABOUT box

Help files produced by registered copies of HELLLP! will "brand" the help files they produce with a copyright title to appear in the WinHelp ABOUT box. By default they will use the name of the registered user for this. The default may be changed with the **Insert User Copyright** function on the **Format** menu.

Unregistered evaluation copies will include identification to that effect in the ABOUT box of WinHelp.



Using HELLLP! and Winhelp to produce corporate manuals

The Windows HELP application is an excellent tool for providing access to corporate manuals, provided you have word processor text for them. The Hypertext and PopUp definition boxes, and the easy incorporation of colour make them more valuable in this form than on paper, and if you keep the HLP files on a server you can be **sure** that everyone is using the same version and that it is up to date.

The users do not need to have a copy of Word For Windows™, since to display help files all you need is the WinHelp application which is included free with every copy of Microsoft Windows™

All you have to do to make this possible it to import the file into **HELLLP!** and go through it marking up topics and jumps, then compile. **HELLLP!** keeps track of things like context strings. It doesn't even matter if you don't know what they are, just uncheck the box that asks if you want to continue confirming them and forget about them.

There are very few restrictions. You should stay away from using dark green in your text (although you can use any other colours, even Dark Cyan gives enough contrast), because it will confuse your users who will not know where to click for hypertext jumps. You should also stick to fonts that you know all of your users have. Graphics are tricky and best avoided by the beginner, but if you are knowledgeable enough to include bitmaps (including those produced by Microsoft's "SHED.EXE" hot spot graphics editor), they will work.

Once you have produced the HLP file, just set up an item in Windows with the command line WINHELP FILE where FILE is the name of the HLP file that the compiler produces, including the path to find it if necessary. Your users will now only have to double-click on the icon to access your manual. You don't need to train them to use another hypertext tool, because they all know how to use Windows Help.

Add Help screens to your programs, templates and macros

Add Help screens to your programs, templates and macros? That's how this help screen was displayed for you. The screens were produced with **HELLLP!** and then a macro called **HelllpHELP** (listed below) was added to the Help menu, and to the [?] on the toolbar.
WinHelp needs to know where to find the .HLP file, so it is easiest to keep it with the template it refers to and use the following macro in **Word for Windows**:

```
Declare Function GetActiveWindow Lib "User"() As Integer
Declare Function WinHelp Lib "User"(hWnd As Integer, lpHelpFile$, wCommand As Integer, dwData As Long) As Integer
Sub MAIN
dotpath$ = GetProfileString$("DOT-PATH")
If dotpath$ = "" Then dotpath$ = GetProfileString$("INI-path")
File1$ = Files$(dotpath$ + "\helllp!.HLP")
ActiveWindow = GetActiveWindow
wCommand = 3
n = WinHelp(ActiveWindow, File1$, wCommand, wData)
End Sub
```

If you wish to copy this code, click on the Copy button below and select the text you wish to use. It will be copied to the clipboard and you may then paste it into your application. (This button was added with the **Insert Copy Button** function on **HELLLP!**'s Format menu).

Copy

If you want to fire up WinHelp to display a particular topic, change wCommand to 1 and set wData to the context number of the topic you wish to display.

The SHELL command can be used in **Visual Basic** from version 1.0 onwards for a simple way to display a help file like this:

```
Sub ShowHelp_Click ()
n = Shell("WinHelp d:\winword\helllp!.hlp")
End Sub
```

Here an object called ShowHelp is created (typically a command button) and the Shell statement is used to call WinHelp when it is clicked. In this example, the Help file called is that for **HELLLP!** being called from d:\WinWord.
Later versions of **VB** have more sophisticated methods built-in, including context-sensitive help where the help topic associated with the control which has the focus comes up when you press F1. You have to tell VB the name of the help file on the menu **Options.., Project** and you have to give each control a number as the HelpContextID property. In HELLLP!, you associate that number with the topic by using the item **MapTopic for Application** on the **Insert** menu.

The only real disadvantage of using the **SHELL** command rather than the API calls (in either VB or WordBasic) is that the application will not close Help when it terminates, and you could have several instances of WinHelp running (but you might want that).

"Busying" a context string

"Busying" a context string may be accomplished for those who wish to use **HELLLP!** to assist them in building multi-file projects. A context string must be unique in the files being built, so **HELLLP!** must know about the strings it cannot use. It must also be able to save externally the strings it has used.

HELLLP! accomplishes this by making reference to a file in the project subdirectory called *FILE*.INI, where "*FILE*" is the name of the project. This file is in the same format as a regular Windows INI file, and under the heading [Contexts] it keeps the context strings used as keywords, and the associated topic titles as their values. **HELLLP!** both reads and writes this file when setting context strings and will not duplicate one that it finds there.

To use the INI file to prevent the use of its strings with another project file (e.g. PROJECT2.DOC), just copy it to the appropriate file name (e.g. PROJECT2.INI).

HELLLP! will automatically handle this if you use its facility for Multiple Project File Builds.

Multiple Project File Builds

It is frequently more convenient to build up a large help file in sections. **HELLLP!** is able to provide assistance with this feature. First, create and test the help files for the individual sections to ensure that the topics are set up and tested. Then create a "top level" help file linking into the individual files by using **HELLLP's** "External Jump" Feature.

When a topic from another file is referenced by a Jump or PopUp, **HELLLP!** sets a flag to ensure that the file referenced will be included in future "Builds" of a help file using the file FROM which the reference is made (the "top level" file). Builds based on the referenced file may be performed without the need to link back to the top level file, unless references go both ways.

A second file referenced in this way is not affected in any way and may be independently updated and compiled. After update ***and compilation*** of such a file, a new build of a file referencing it will incorporate the updated version.

It should be remembered that if even a single topic of a secondary file is referenced, then all topics it contains will be included in the Help file - even they cannot be reached by jumps, because the Microsoft Help Compiler will incorporate the whole file.

Note that if there are references back **from** the secondary file, there may be compiler warning messages generated since some topics in the primary file may need references to other secondary files which the current one knows nothing about.

External Jumps are accessed using the **[J]** and **[P]** buttons on the tool bar.

The easiest way to link a multiple file build is to make External Jumps from the "top-level" file to the tables of contents of the other files, although any topic of any file may be referenced from any other file.

"Boiler Plate" files of definitions etc. may be created and automatically linked in when referenced.

Compression of Help File

The Windows Help Compiler is able to compress the help file it produces, which both saves disk space and speeds loading when HELP is invoked. However, compilation is slower when compression is turned on. **HELLLP!** provides an option permitting compression to be turned on and off. It is invoked by selecting **HELLLP! Compression Options** from the Format menu. For a new file being created by **HELLLP!** compression is initially turned off.

Differences can be significant. Without compression, one version of the help file for **HELLLP!** compiled in about half a minute, but resulted in a file size of 376 kbytes. With compression it took 9 minutes to compile, but the file size was reduced to 213 Kbytes

HELLLP!

for the

Expert User

HELLLP! is designed to be easy to use, and to produce help files rapidly. However, even if you are an expert user of the Help compiler, **HELLLP!** can still help you. It produces a DOC file with all of the codes in that you need to compile a help file, and you can edit it to customize it further. Since the expert user may wish to produce a help file from several source files, **HELLLP!** provides the means to "Busy" context strings to ensure that they are unique across several files.

Alternatively, the expert user may choose to employ **HELLLP!**'s automated multiple project file build feature.

The expert user may also be interested in **HELLLP!**'s ability to create the association between Context Numbers for applications and Context Strings by the Map Topic For Application feature.

The following topics may be of particular interest to the expert user:

- "Busying" a context string
- Compression of Help File
- Hotspot graphics and the SHED editor
- HELLLP! from a network server
- Error Messages
- Multiple Project File Builds
- External Jumps
- User Copyright Line in WinHelp ABOUT box
- Mapping Context Strings to Context Numbers and HelpContextID
- Adding Sound by Playing *.WAV files
- Spawning an External Process from a Help File
- Setting Help Window Size and Screen Position
- Button Selection and Control
- Smart Quotes
- Non-Scrolling Region
- The symbols +#!K$

Hotspot graphics and the SHED editor

It doesn't have to be like this!

You may embed graphics by capturing them on the clipboard from another application (like PAINTBRUSH) and then using EDIT, PASTE SPECIAL to paste them in as bitmaps. You may still make them HotSpots by selecting the graphic before clicking the **[J], [P]** or other action button.

HINT: If you want your graphics to appear beside your text, create a table and insert the graphic in one of the cells. If you display the topic as a PopUp it will be automatically set to the correct size and shape. You can even add a pictorial business card this way. For another example of this effect, see the topic Setting Help Window Size and Position.

A good but frequently overlooked source of graphics for use in Help files is the clip-art library supplied with Microsoft power point.

Open PowerPoint, under **File** select **Open Clip Art**, select and display the graphic your are interested in; using the mouse, select the area of the graphic which you want and use Edit Copy to place it on the clipboard. Return to Word and paste it to your file.

<< This one was pasted into a table cell as explained above. It was then selected again, and the Add Sound to Topic function on the Insert menu was used to associate it with the SystemAsterisk sound, which you will hear if you click it.

Graphics with "hotspots" produced by the SHED editor may be incorporated into help files produced by **HELLLP!**

The SHED editor requires some expertise to use, but may be downloaded free from CompuServe's Windows SDK forum section WinHelp, file SHED.ZIP. It has an extensive HELP system.

Since the SHED editor must be told the context strings to use for particular hotspots, the topics to be accessed by the graphical hotspots should be created first. The context strings that **HELLLP!** creates will be found adjacent to the topic titles, both at the head of the topic and in the automated table of contents. They are also listed in the project's INI file.

Once the topics are created, the graphic may be created and edited with SHED, then saved as a file (with a .SHG extension). Note that these files usually take much less space than PCX files.

HINT: Run the SHED editor in the left of your screen, run Winword in a window on the right, displaying the automated table of contents at the top of your file.

To incorporate the graphic in your help file, include a line like this at the location where you want the graphic to go:

```
{bmc c:\helpcomp\project\helllptb.shg}
```

Remember the curly brackets. This line was used to create the hotspot graphic of the **HELLLP!** toolbar in this file. To place the graphic at the left, use bml instead of bmc (which centres). To place it at the right, use bmr.

Even if you do not wish to use HotSpots within the graphic, using graphics created by the SHED editor is efficient since SHED's *.SHG files are smaller than most. If you incorporate graphics directly in the DOC file, it becomes very large and may reach the limit of the compiler. You may make the whole graphic a hotspot by highlighting (selecting) the command which incorporates it (including the curly brackets) and using the **[T]** or **[P]** toolbar buttons.

If you wish to embed graphics and you do not have access to the SHED editor, you may embed regular bitmap files (*.BMP) in the same manner as *.SHG files.

Choosing an Icon for the help file

The default icon used when a help window is minimised is the yellow question mark. You may choose a different icon by using the **Choose Icon** function on the **Format** menu. You will be presented with a dialog box asking for the full file name of the icon file you wish to use.

Mapping *Context Strings* to *Context Numbers* and *HelpContextID* for Applications

For access by context-sensitive help in applications, a topic must be identified by a context number (called the HelpContextID property in Visual Basic) instead of by the context string. The association of a context number with a context string is made by the Help Compiler. **HELLLP!** has the ability to insert the necessary instructions to the Help Compiler via a procedure accessed by **Map Topic For Application** on the **Insert** Menu.

This allows you to associate a context number with any Help topic which you have set up in the current file or in any file which you have created and compiled in the same project subdirectory. If the topic is in a different file, then that file will be included in future builds of a *.HLP file from the current file. The other file is not affected in any way, it is just copied into the *.HLP file.

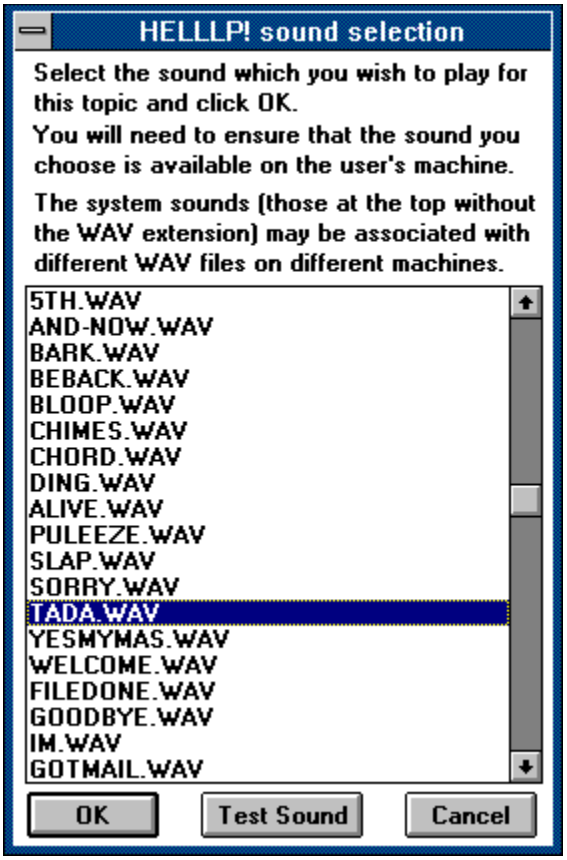
Adding Sound by Playing *.WAV files

Click on the bell. The Windows help system is able to play sound (*.WAV) files, provided they are in the WINDOWS subdirectory and provided that the machine is able to play them. A machine without a sound card can play *.WAV files through the built in speaker with the aid of the Microsoft Speaker Driver, available as the file SPEAK.EXE from the Microsoft Software Library (GO MSL on CompuServe).

HELLLP! implements the ability to play a WAV file when a topic is displayed, or when HotSpot text is clicked. To invoke the feature, use the item **Add Sound To Topic** on the **Insert** menu or click the "Speaker" icon on the toolbar. If text is selected at the time, that text will become HotSpot text (looking like a hypertext jump) and the sound file will be invoked when it is clicked (this one plays your "SystemHand" WAV file). If no text is selected the sound file will be played when the topic is displayed, as it did when this topic displayed. If you select this item several times, each sound will be played in turn in the order of selection. Additionally, if graphics are included in the file, they may be linked to WAV files just like text. Highlight the graphic or the statement in curly brackets (including the brackets) which incorporates the graphic and then choose Add Sound To Topic. The graphic then becomes a HotSpot and the WAV file will play when it is clicked.

The "speaker" graphics were incorporated that way and linked to the System Sound called "SystemExclamation". That sound event is associated with a WAV file in the **[sounds]** section of **WIN.INI**, and that WAV file will play when it is clicked.

Sounds are selected using the dialog box below, which appears when the **Add Sound To Topic** function is selected or the speaker button on the toolbar is clicked.



Of course these buttons are "live" - click one and see!

The HotSpot buttons here were set up using the SHED editor.

To remove sounds from the opening display of a topic, use the function **Clear Topic Opening Effects** on the **Edit** menu. Please note that it will also clear any window size change set up as an opening effect for this topic.

If you merely wish to use sounds to gain attention, it is probably wise to use the standard SystemSound names since most machines will have them defined and you do not need to distribute WAV files. If, however, you need specific sounds you should use WAV files and distribute them with your application. Windows 3.1 came with the files CHIMES.WAV, CHORD.WAV, DING.WAV and TADA.WAV. Other WAV files are available widely from bulletin boards.

Spawning an External Process from a Help File

The Windows help system is able to start external processes, provided that the machine is able to find them. In this way, various multi-media applications may be controlled from a help file.

 As an example this graphic will start the Calculator application when you click it. So will this text.

HELLLP! implements the ability to spawn an external process when a topic is displayed, or when HotSpot text is clicked. To invoke the feature, use the item **Spawn External Process** on the **Insert** menu. If text is selected at the time, that text will become HotSpot text (looking like a hypertext jump) and the external process will be invoked when it is clicked. If no text is selected the external process will be invoked when the topic is displayed.

Additionally, if graphics are included in the file, created by the SHED editor or directly embedded, they may be linked to external processes just like text. Highlight the graphic of the statement in curly brackets (including the brackets) which incorporates the graphic and then choose **Spawn External Process**. The graphic then becomes a HotSpot and the process will start when it is clicked.

Spawn External Process will display a dialog box asking for the command line which you wish to be invoked. You should enter the command line exactly as you would enter on the Windows Program Manager **File Run** menu.

 This graphic, when clicked, will start up the Sound Recorder acting on the file \WINDOWS\TADA.WAV (if the current machine does not have that file, the sound recorder will give its normal error message). The "Command Line" entered in the **Spawn External Process** dialog box was **SOUNDREC.EXE C:\WINDOWS\TADA.WAV**.

Setting Help Window Size and Screen Position

You may specify the size and position which your Help window will use when it first displays (your users can move it and resize it as usual afterwards). You may also define up to five secondary windows using this function.
The function is found on the **Format** menu as **Set Help Window Size**. When invoked, it presents a dialog box giving you the ability to set the margins around your help windows as a percentage of screen size. It will not allow you to set impossible sizes and will override your choice if you attempt to. The values pre-set in the dialog box are those which are set for the initial display of help. This makes it easy for a topic to force a restore to the initial window size if reached after the size has been changed. It is a good idea to set the initial display screen topic to this size, as well as the topics in the browse sequence before and after one which sets a different size.

CLICK ON THIS GRAPHIC

This function was used to cause this size change. To restore to the size and position which this help file had on opening, click on CONTENTS or one of the browse buttons (those topics also have a size and position setting).

This graphic is a hypergraphic, also to demonstrate an effect. This interesting effect was created by setting up a three column table and including the bmc statement explained in the topic "SHED EDITOR" inside the left hand cell(which is very narrow). This text was placed in the right hand cell.

The shadowed overlay was created by including that text in the centre cell.

You could use this to "Watermark" example graphics in your help file to prevent copyright violation by screen capture.

You may choose to fill the screen by using zero margins all round if you wish (your user will still be able to resize the window as usual).

You may elect to have the window change to a specific size and position when a topic is displayed. To do this, ensure that the cursor is in the topic concerned before selecting **Set Help Window Size** and then click the lower option button in the dialog box. Please note that a Help window size, once set, remains in effect until it is explicitly changed either by the user or by a topic with a different setting. If windows sizes are critical to your desired effect, you might want to do an explicit window size setting for every topic - but this may annoy a user who keeps altering it, only to have it switch back the next time the topic is displayed.

To remove window size changes from the opening display of a topic, use the function **Clear Topic Opening Effects** on the **Edit** menu. Please note that it will also clear any sound effects set up as an opening effect for this topic.

Contents Copy Close

Button Selection and Control

The Windows help system can have a variety of control buttons just below its title bar. The selection of which buttons will appear there is made at the time of help file compilation. **HELLLP!** supports the selection and placing of several of these. The function is found as **Set Button Selection** on the **Format** menu.

To change the buttons on the main window button bar, choose **Button Bar for Main Window** in the initial dialog.

Options are provided to add:

- Browse Buttons (on as default)
- Print Button
- Copy Button
- Close Button (Closes the main help window)
- Exit Button (Closes all help windows)

An option is also provided to change the topic accessed when the **Contents** button is clicked. In this way, it is possible to set one topic for initial display (using the I button) and a different one to be displayed when the **Contents** button is clicked. This help file is set so that the title screen is initially displayed, but the contents button invokes the SideBar Table of Contents.

A special set of working buttons is provided in the Template Glossary (as hotspot graphics, sometimes with associated macros), from which they may be inserted in the text using **Edit, Glossary, Insert**. This permits the insertion of a working Copy button as a graphic directly in the text, like the one below. This feature is useful if your help file has examples of text or code which you may want to provide a user-friendly copying method. An example may be found in the help topic "Add Help screens to your programs, templates and macros".

Copy

Buttons with an associated macro, like the **Copy** button are inserted with a dark yellow macro definition at their right. Some buttons, like the **Yes** button have no associated macro. These are intended as hotspots to jump or PopUp from. Simply select (highlight) the button graphic like any other hotspot graphic and use the **[J], [P],** Speaker or Screen button on the toolbar to determine the action to be taken when it is clicked.

The glossary buttons also provide a means to add **Contents**, **Close**, **Exit**, **Copy**, and **Print** buttons to the non-scrolling region of topics in main or secondary windows. For an example of **Close** and **Exit** buttons used this way, see the **Sidebar Table of Contents** of this help file.

To create a working *horizontal* button bar for inclusion in topic text, create a table and insert the glossary buttons in the cells.

HELLLP! provides an automated means to insert, change or delete a horizontal button bar in this manner in a non-scrolling region at the head of a topic. Select the option **Button Bar For Current Topic** from the initial dialog for **Format, Set Button Selection** in order to achieve this. To delete a topic button bar previously defined, simply uncheck all of the boxes in the dialog box.

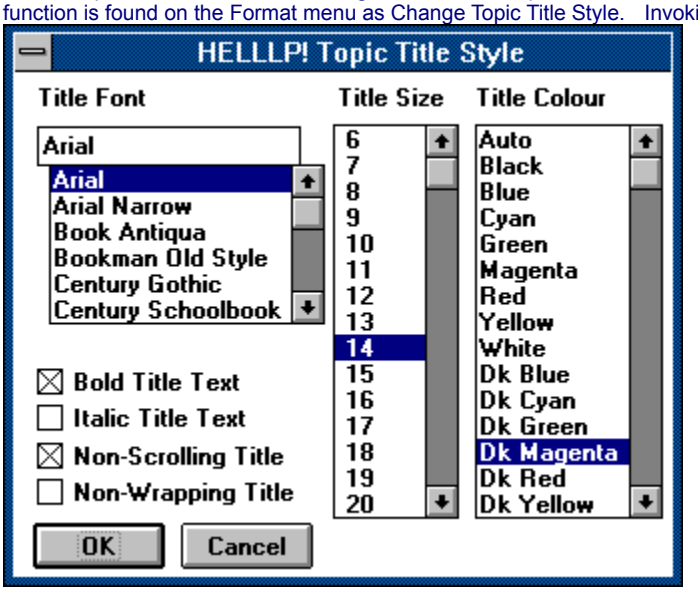
This option was used to insert the extra **Contents, Copy and Close** buttons at the head of this topic. Notice how the extra **Contents** button is connected to the main window table of contents, while the main one brings up the SideBar Table of Contents.

Buttons in a Topic Button Bar for **Contents**, **Help**, **Back** and **Browse** permit (and require) the help file author to determine which topic will display when they are clicked. This is necessary, both to make them topic-sensitive and because the regular functions will not work if called from a secondary window.

The Microsoft help compiler will warn of a "hidden paragraph" when this feature is used. This is normal and the warning may be ignored.

Changing Topic Title Style Colours and Fonts

HELLLP! provides the means to change the standard topic titles which it has generated to another font, colour and size. The function is found on the Format menu as Change Topic Title Style. Invoking it brings up the following dialog box:



About Shareware

Shareware is copyrighted software that is distributed by authors through bulletin boards, on-line services and disk vendors.

Shareware allows you to try the software for a reasonable limited period. If you decide not to continue using it, you throw it away and forget about it. You only pay for it if you continue to use it. Shareware is a distribution method, not a type of software. You benefit because you get to use the software to determine whether it meets your needs, before you pay for it.

The shareware system and the continued availability of quality shareware products depend on your willingness to register and pay for the shareware you use. It's the registration fees you pay which allow authors to support and continue to develop our products. Please show your support for shareware by registering those programs you actually use.

Even if the person who supplied you with the software tells you that registration is not necessary, be assured that if the copy of the program presents a screen reminding you to register, then the author has not been paid for that copy and registration is necessary.

This program is produced by a member of the Association of Shareware Professionals (ASP). ASP wants to make sure that the shareware principle works for you. If you are unable to resolve a shareware-related problem with an ASP member by contacting the member directly, ASP may be able to help. The ASP Ombudsman can help you resolve a dispute or problem with an ASP member, but does not provide technical support for members' products. Please write to the ASP Ombudsman at 545 Grover Road, Muskegon, MI 49442-9427 USA, FAX 616-788-2765 or send a CompuServe message via CompuServe Mail to ASP Ombudsman 70007,3536.

Using a copy that has been made from someone else's registered copy is contrary to copyright law.

Only the copyright owner, Ed Guy, can give permission for copies to be made. He has given permission for copies to be made freely for 30 day evaluation purposes, provided that they are so marked internally.

To make an illegal copy into a legal evaluation copy, select the item "UNREGISTER" on the TOOLS menu, and the registered user's name will be changed to "Unregistered User". The copy may then be used for 30 days of evaluation, and copies may be made for distribution to others for the same purpose. Copies of **HELLLP!.DOT** may only be distributed together with **HELLLP!.HLP.**

After an evaluation period of 30 days, you are required to either pay for the continued use of the product, or to cease its use. "Registration" involves payment, and the receipt of a registration number. The registration number is an encryption of the user's name which will be displayed in the copyright box when **HELLLP!** starts.

Registration
and
Payment

Registration supplies the right to the continued use of the product after the 30 day free trial, and the elimination of the reminder screens. It does not rescind the legal disclaimer of liability under which this product is used.

Additionally, Windows HLP files produced by an unregistered version of **HELLLP!** are so identified in the "About Help" box (where registered copies permit a user's copyright notice to be placed) and they have a topic at the very end noting that they were produced by an unregistered version. That topic is seen by the users only if they use the browse buttons to go to the very last topic, or if they go to the search topic UNREGISTERED. It does not appear in the table of contents.

Credit card orders may be submitted to **PsL**

A cash order form may be obtained by printing the topic INVOICE

For use on a single computer, registration costs $30 US, paid to:

Guy Software,
1752 Duchess Avenue
West Vancouver
British Columbia
Canada, V7V 1P9 (CompuServe 71750,1036)

If the user provides an electronic means for the passage of the registration number (Electronic mail accessible via CompuServe or a Secure Fax number), then the cost is $20 US per copy in recognition of the lower costs involved. Please ensure that any electronic route is secure, since anyone with your registration number can make it appear that you have illegally given them a copy registered to you. This special electronic registration may be performed entirely within CompuServe using their Shareware Registration Database.

If the route chosen permits the transmission of a binary file, an MsWord DOC file will be transmitted which, when opened in Word automatically performs the registration. In other cases it will be necessary to type in the registered user's name and registration number.

For network use, the price is $100 US if no more than six people may access the product at once, and $10 US per additional permitted user for numbers in excess of that. If the user's network has the ability to limit the number of people accessing a file, that number may be used instead of the number of nodes on the network.

CASH INVOICE (Please send credit card orders via PsL)

Name: _____

Address: _____

_____

_____

_____

_____

Supply of the **HELLLP!** System as below
Special "Electronic Registration" package
for one computer only $20.00
Mailed Package-for one computer; $30.00
Mailed Network Package first six users $100.00
Additional users on network @ $10.00 each _____

Payment Enclosed:..... **$**_____

For electronic registration,
please give Email or Fax address _____
to send registration code.

Name for identification on
copyright screen (if different) _____

Fold here first

AFFIX POSTAGE

GUY SOFTWARE
1752 Duchess Avenue
West Vancouver
British Columbia
CANADA V7V 1P9

Fold Here second and then tape edges

Registration and Payment via CompuServe

CompuServe provides a convenient route for the registration of Shareware, and the special "Electronic Registration" of **HELLLP!** may be performed by that route. Note that the electronic registration price of **HELLLP!** is only dependent on your providing an electronic route for us to send you your registration code number. That price is available whether or not you use CompuServe to contact us.

To access CompuServe's registration database, **GO SWREG** in CompuServe. **HELLLP!** is **Registration ID 1346** in the CompuServe database and its price is **$20.00(US)**. Only the "Electronic Registration" is available by this route. Once CompuServe informs us of your registration, we will transmit to your CompuServe ID a WordForWindows document called **REGISTER.DOC**, and when you open that in Word your registration will occur automatically. That same document will permit you to register the next few upgrades as you download them (none are planned, but inevitably there will be maintenance releases for bug fixes and minor enhancements).

The following instructions are provided by CompuServe on the use of the Shareware Registration Database.

Register Shareware+

INSTRUCTIONS TO REGISTER SHAREWARE

The Shareware Registration database contains descriptions of hundreds of shareware programs designed for a variety of platforms and purposes. To register a program, you must first find its description in the Shareware Registration database.

To begin, select "Register Shareware" from the previous menu. A list of search criteria will be displayed to you. You may search for your program's description by its TITLE, FILE NAME, AUTHOR'S COMPUSERVE ID, and AUTHOR'S NAME. You may also search for your description by its KEYWORDS (categories). For example, to find the description of a Macintosh utility, you might use the keyword: utility. (For information about registering by your program's REGISTRATION ID, see below.)

After you enter your search criteria, you are usually displayed a list of programs that meet the criteria in the database. Select the program that you wish to register from the list.

Sometimes after you enter search criteria, you are displayed a menu of choices. This occurs when your search yields a large number of "matches" in the database. The menu offers you the ability to NARROW your results further, to DISPLAY the list of "matches," or to BEGIN AGAIN. After you select your program from the list, you are displayed the program's description, with a prompt: "Would You Like to Register? (Y/N)."

Read the description carefully to verify that you wish to register the program. The description details the program, its capabilities, and the benefits of registration. Be sure to note the program's registration fee. When you register any program, the registration fee is charged to your CompuServe account.

Enter "Y" at the prompt to register; enter "N" at the prompt to ensure that you do not register the program, and to return to the previous menu. When you enter "Y" to register, you will be prompted to enter your full name, company name (optional), your complete address, your phone number (optional) and the total number of copies of the program that you wish to register. When you finish entering this information, you can VIEW, CHANGE, SEND or CANCEL it.

Select VIEW to confirm that the information you entered is correct. Select CHANGE to edit any of the information you entered. Select SEND to complete your registration, receive a detail of your total charges, and to send the information to the author. Select CANCEL to cancel registration of the program. When authors submit their descriptions to the Shareware Registration database, their descriptions receive a REGISTRATION ID. If you know the REGISTRATION ID (for **HELLLP!** the registration ID is **1346)** of the program description in the database, you may enter it under option #1 on the search criteria menu. You will then navigate directly to the program's description. This is the easiest and fastest method of registering a program.

If you have any questions or concerns about shareware registration, send a message to the Shareware Administrator by selecting the "Provide Feedback" option at the main SWR menu.

Upgrade Policy

Most shareware authors are appreciative of suggestions for product improvement made by their customers. We are no exception and have found that most ideas for improvement have come from users who have said "It would be nice if ...". Our only caveat has been that we do not add features in a way which will harm the user-friendly interface of **HELLLP!**.

This sometimes results in unpredictable releases to add valuable new user-suggested features. Our upgrade policy is that registration for a previous release will be honoured automatically for a new release. **HELLLP!** handles this automatically in most circumstances. The first time a new unregistered release is used on a machine where it replaces a registered release, it will present the registration dialog. The previous User Name and Registration Number will be honoured. In most cases they will be pre-entered in the dialog box, having been stored on the machine in a parameter file (HELLLP!.INI) by the previous version. Upgrades are thus free for the downloading from any source carrying the unregistered version.

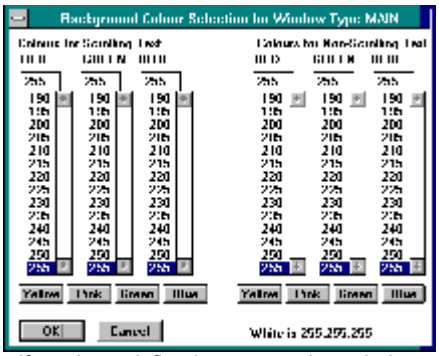
Colours you can use

Colours you can use. You probably won't want to use some of them, and if you aren't looking at them on a colour screen you are having to imagine them.
You may change the colours of the topic titles all at once with the **Change Topic Title Style** item on the **Format** menu.

Black
Blue
Cyan
Green
Magenta
Red
Yellow
Dark Blue
Dark Magenta
Dark Red
Dark Yellow
Dark Grey
Light Grey
Dark Green looks too much like Hypertext or PopUps.
For some really wild effects, you can change the background colour.

Background Colours

You may change the colour of the background for the main and secondary windows independently. Click here for a sample**.**You may use different colours for the scrolling and non-scrolling areas if you have set non-scrolling titles.



The **Set Background Colours** function is on the **Format** menu, and presents a dialog box allowing you to set colours in terms of RGB (Red Green Blue) values, with four buttons to preset basic pale colours which you can further vary. To find the RGB values for a given colour, you may use the **Color** control of the Windows **Control Centre**.

If you have defined any secondary windows, you will first need to respond to the dialog box determining for which window you are setting the colours.

If you do not set a background colour, the WinHelp application will use whatever background colour the user has set as his/her windows default. If you are using colours for your text or graphics, they may not show up against the background the user has set. You might even want to force a white background using this function, to ensure that the help topics look the same to your users as they do to you. This help file has background colours set this way, both for the reason given and (as with many things in this file) to show what can be done using **HELLLP!**

Error Messages

Error Messages may sometimes appear. They probably mean that you accidentally deleted a hidden code of some kind. **HELLLP!** uses these MsWord "Bookmarks" to hold values and as place markers. There should be several of them and a page break before the Automatic Table of Contents. There are also two within the text, one called "IwasHere" in the main text - used to keep track of where to come back to after doing something elsewhere, and one called "nextTOC", marking the location for the next entry in the Automatic Table of Contents.

HELLLP! tries very hard to restore any bookmarks that it finds missing, but it will not always succeed.

If you get a **"Bookmark not found"** when **setting up a topic**, you have probably deleted the "nextTOC" bookmark in the table of contents. To restore it, position the insertion point (The cursor) in the last entry of the Automatic Table of Contents and click the UPARROW button on the toolbar.

If your new Help File has lost the title from its title bar, you may re-create it with the **HELLLP! Project Title** selection on the Format menu.

The other bookmarks will be re-created when you perform an appropriate action. The "IwasHere" will be recreated when you create a new topic, jump or PopUp. Those which hold settings from checkboxes will revert to default values, and will be re-created when the values are changed again.

There may be other errors not listed here, no program is free of bugs including MS Word for Windows (As documented in the Hacker's Guide to Word for Windows - strongly recommended reading.)

Should you receive error messages from the Microsoft help compiler which stop the compile, you should check the version of the help compiler and possibly download an updated version as described in Getting Started. Each version change in WinWord seems to require an updated compiler to handle the Rich Text File (RTF) form which it produces.

DISCLAIMER: The supplier and author bear no liability to the licensee or any other person or entity for any damage or loss, including special, incidental or consequential damages caused by this product directly or indirectly. The software is supplied "as is" without warranty of any kind, either expressed or implied. Warranties of merchantability or of fitness for any purpose are specifically disclaimed.

HELLLP! was inspired by a series of articles by Ray Duncan in PC Magazine in the spring of 1993, and by Woody Leonhard and Vincent Chen's excellent book the Hackers Guide to Word For Windows.

This is a Pop Up Box. The Microsoft literature calls it a "definition box" because it's useful for definitions of terms, where it may be popped up without taking the user away from the context he or she was looking at. It's useful for lots of other things, though. Your imagination is the only limitation.

That's right, you just clicked on some text that will cause another topic to be displayed. Where it has a dotted underline, a box like this pops up, where it has a solid underline you are taken to the topic. Don't worry, you won't get lost, because pressing the Back button will retrace your steps. If you are using Windows 3.0 the Back button is a pair of footprints.
To make this box pop down again, click again.

Non-Scrolling Region

The top section of a topic may be set as a non-scrolling region - useful for titles and headings. This is done by setting the paragraph formatting for those paragraphs as Keep With Next and Keep Lines Together. For the topic title, this is an option on the **Change Topic Title Style** function on the **Format** menu. Setting a non-scrolling section after a scrolling section is not permissible and will result in a compiler error message. Note that any topic which may be called as a PopUp must not have any non-scrolling text or text below it will not be displayed. If you are setting up a topic which may be called as a PopUp, either do not have **HELLLP!** insert a topic title, or manually reset the paragraph formatting afterwards.

The Question Mark Button brings up the Windows Help File for **HELLLP!**. That is the file you are looking at now.

Word for Windows is often set up so that it automatically incorporates typesetters "curly quotes" instead of the familiar " and '.

However, the Help system cannot display the curly quotes. **HELLLP!** therefore provides the means to convert them back to " and ' by using the **Undo Smart Quotes** feature on the **Tools** menu.

Hypertext is the term used for a method of moving around an on-line document to follow cross references, etc. In Windows Help, whenever a user clicks on an underlined green section of text, the application jumps to the page of the cross reference, but it keeps track of where it was. Users are able to retrace their steps using the Back button. Much better than keeping a thumb in the page, because you quickly run out of thumbs! If the green text has a dotted underline, then the reference "pops up" in a box like this. Think how useful this will be if you have lots of terms that mean things to your experienced staff, but not to newcomers.

This line of text is, of course, a graphic because if the text were used the picture would be here instead of the sample line!

This button lets you check the sound.

This button incorporates the sound into your file.

This button exits.

This box lists all of the fonts on your system. Some may not be suitable for titles, and you should ensure that you choose a font which your users will have on their machines.

This box list the font sizes available for topic titles.

This box lists all the colours available. You probably won't want to use the paler colours and you certainly won't want to use white.

These boxes allow you to choose these attributes.

This button implements the change.

This option sets the window size when WinHelp opens this file.

This option sets the window size when this topic is displayed.

These option buttons set the size of secondary windows, and turn on the options which deal with them.

Credit Card Orders Only

You can order with MC, Visa, Amex or Discover from Public (software) Library by calling 800-2424-PsL or 713-524-6394 or by FAX to 713-524-6398 or by CIS Email to 71355,470. You can also mail credit card orders to PsL at P.O.Box 35705, Houston, TX 77235-5705.

The above numbers are for orders only.

Please quote **item/product code # 11177**

Any questions about the status of the shipment of the order, refunds, registration options, product details, technical support, volume discounts, dealer pricing, site licenses etc. must be directed to:

1752 Duchess Avenue
West Vancouver
BC Canada V7V 1P9
(604)926-1370 CIS 71750,1036

To ensure that you get the latest version, PsL will notify us the day of your order and we will ship directly to you.

Secondary Windows

The Windows help system has the ability to display secondary windows, in addition to main windows and popups.

To initiate the use of secondary windows, you use the **Set Help Window Size** function on the **Format** menu, to set the size and position for one or more Secondary Windows Types (identified as **windowA** through **windowE**).

Once you have defined secondary windows, it will be necessary to respond to this dialog box when performing some other functions, in order to identify which windows they apply to.

You may differentiate your secondary windows from your main window by giving them different background colours, using the **Set Background Colour** function on the **Format** menu.

You may choose to have a secondary window (one only), as well as the main window, displayed when Windows Help opens a help file. To do this use the **[I]** button and select the secondary window to be employed, and the topic to be displayed in it. **If you use this option you may not change the file name of the help file after compilation.** It is frequently convenient to display a "Sidebar Table of Contents" in a secondary window, and use it to control topic display in the main window.

Welcome to the world of HELLLP!

To get help about the use of the **HELLLP!** system at any time, just click on the question mark button on the toolbar.

To view the **HELLLP!** help table of contents now, click on the **[Contents]** button of this window.

The **HELLLP!** system needs to know where you want it to keep its working files and will ask for the name of a subdirectory for that purpose. You should have set up a directory for that purpose separate from any other files, to keep all of your **HELLLP!** files in. **HELLLP!** will create files with the extensions **DOC, RTF, INI, HPJ, BAT** and **HLP** for each help project in this directory. If you have not set up a directory, you will not be able to complete this setup process and you should click **[CANCEL]** now, to give yourself the opportunity to create one.

HELLLP! also needs to know the name and location of the copy of the Microsoft Help Compiler on this machine, and will present a dialog box asking for that information.

Contents Close Exit

This is a secondary window.

Unlike a PopUp, it will not go away when you click again. It must be specifically de-activated.

To use secondary windows in your application, define a size and position for it/them using the **Set Help Window Size** function on the **Format** menu. After you have done this, you will be required to tell **HELLLP!** which window type (main or one of the secondaries you have defined) you wish to use for every jump you define and for certain other functions. A dialog box will be presented for this purpose.

You may specify sizes and positions for up to five secondary windows, but only one may be displayed at once. Click to bring back the sidebar table of contents.

If you are creating a multi-project file build, the window types you are going to use should be defined in each of them. During the compile, the sizes and positions set for the file from which the compile is called will be the ones used.

Customising lines in the HPJ file (for experts)

The expert user may wish to add lines to the **HPJ** file which are not generated by **HELLLP!**
This may be achieved by creating a file in the same format as the **HPJ** file, but with the extension **HPK**. When **HELLLP!** creates the **HPJ** file, it checks for the existence of an **HPK** file and merges the lines from it into the appropriate sections of the **HPJ** file. These lines are added after any lines which **HELLLP!** puts into each section.
Before using this option, you should look at the **HPJ** file created by **HELLLP!** to ensure that any lines you put into the **HPK** file do not conflict with them.

Contents Close Exit

This is a secondary window to show this rather wild effect. You will have to close it to get rid of it (or else click CONTENTS). For more on secondary windows, click here.

Title Bar Text Setting

You may set the text to be displayed in the title bar when Windows Help displays your help file. The function is found on the **Format** menu as **HELLLP! Project Title Bar.**

yesyesyesyesHELLLP! Generatedwindsize

Table of Contents

Help file produced by **HELLLP!**, a product of Guy Software, on 01/02/94 for Edward Guy.
The above table of contents will be automatically completed and will also provide an excellent cross-reference for context strings and topic titles. You may leave it as your main table of contents for your help file, or you may create your own and cause it to be displayed instead by using the I button on the toolbar. This page will not be displayed as a topic.
HINT: If you do not wish some of your topics to appear in the table of contents as displayed to your users (you may want them ONLY as PopUps), move the lines with their titles and contexts to below this point. If you do this remember to move the whole line, not part.

PopUpSection
Left
Right
Top
Bottom


HELLLP!


Program Manager
File Options Window Help
HELLLP!


Program Manager
File Options Window Help
HELLLP!


Program Manager
File Options Window Help
HELLLP!

The Window Size Dialog Box

It is easier to show how this box works than to explain in words. Therefore the buttons themselves are live to show which one affects which dimension.

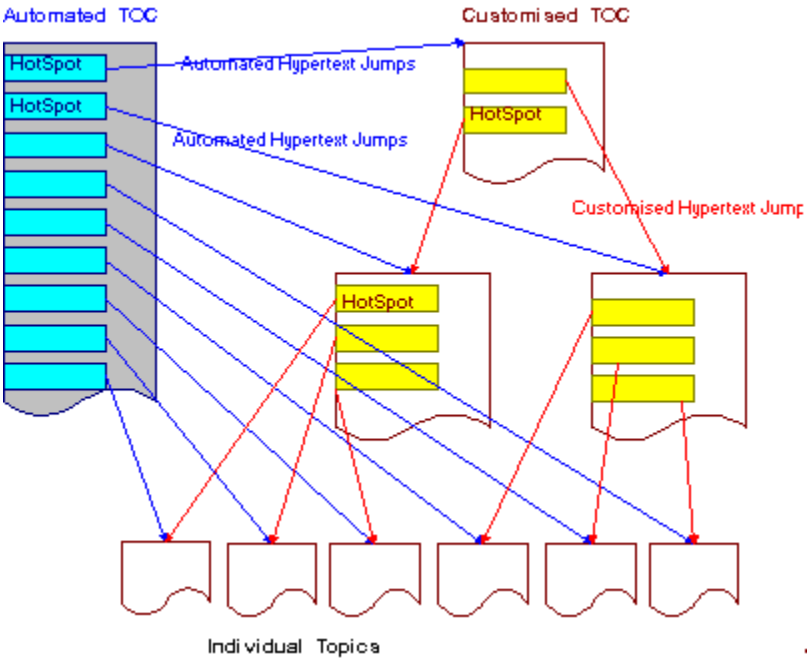
FALSEyesyesyesyesHELLLP! Generatedhypertxt

Table of Contents

Help file produced by **HELLLP!**, a product of Guy Software, on 01/02/94 for Edward Guy.
The above table of contents will be automatically completed and will also provide an excellent cross-reference for context strings and topic titles. You may leave it as your main table of contents for your help file, or you may create your own and cause it to be displayed instead by using the I button on the toolbar. This page will not be displayed as a topic.
HINT: If you do not wish some of your topics to appear in the table of contents as displayed to your users (you may want them ONLY as PopUps), move the lines with their titles and contexts to below this point. If you do this remember to move the whole line, not part.

Hypertext Document Structure

A windows help file usually contains several levels of hypertext "menus", in which there are jumps from a top-level menu to one or more other levels of menu. **HELLLP!** makes customization of this structure very easy. The cool colour "blue" items on the structure below are automatically created by **HELLLP!,** the hot colour "red" items are inserted or imported by the author.
This drawing is a hypertext picture. Just click on an item for an explanation of its place in the scheme of things.



This table of contents is created by HELLLP!, and it has automatically created hypertext jumps (shown in blue) to every other topic set up by HELLLP!. Unless the Help file author specifies otherwise, this is the table of contents which is first displayed when the help file is invoked. For a simple help file, the author may be satisfied with this. However most authors will want to create their own structures. The automated table of contents is able to assist in this, since it has every topic listed along with its context string, all in one place.

When an author clicks on the **[J]** or **[P]** buttons, he/she is taken to this table of contents to select the target of the jump or PopUp, since titles are easier to remember than context strings.
The red jumps are those inserted by the author, the maroon and yellow topics are those inserted or imported by the author.

This is the Author's Table of Contents or other opening screen. Since no automatic hypertext jumps go **to** the automated table of contents and it does not appear in the keyword list or browse sequence, there is no way in which it can appear to the ultimate user if the author uses the button to set this as the initially displayed topic.

These are the lower level topics, but even they may have hypertext jumps or PopUps leading from them.

These are intermediate levels of "Menu" set up by the author. There may be many of these (in the thousands if needed).

These are hypertext jumps which are automatically set up by HELLLP!

These are HotSpots, either text or graphics, included in the text. The jump occurs when the user clicks the mouse pointer on one of them.

These are hypertext jumps which are set up by the author, using the Point & Shoot techniques provided by HELLLP!.

Edward Guy ***P. Eng., CDP***

Information Technology Consultant