

Q D H E L P
Q D H E L P is a tool for generating help files for
MS Windows. With that said, the following mandatory legal mumbo jumbo is necessary.
Q D H E L P is distributed "as is" without any warranty of any kind.
IN NO EVENT SHALL PHIL ALLEN BE LIABLE OR RESPONSIBLE FOR ANY LOSS, DAMAGE,
OR OTHER PROBLEM CAUSED BY USING Q D H E L P. THE ENTIRE RISK OF USING Q D H E L P
IS YOURS.
ct described in this manual whenever without notice.

I have made every effort to be sure that QDHELP is an effective tool for generating help files for
MS Windows. With that said, the following mandatory legal mumbo jumbo is necessary.
QDHELP is distributed "as is" without any warranty of any kind.
IN NO EVENT SHALL PHIL ALLEN BE LIABLE OR RESPONSIBLE FOR ANY LOSS, DAMAGE,
OR OTHER PROBLEM CAUSED BY USING QDHELP. THE ENTIRE RISK OF USING QDHELP
IS YOURS.

1. Intro

QDHELP has many features that make your task of writing help files easier than ever before. This document presents detailed information on all of them. Try QDHELP and I'm sure you will be more productive as you generate MS Windows help files the Quick and Dirty way!

QDHELP is shareware. Users can use an unregistered copy of QDHELP for thirty (30) days. After that time users must either register their version of QDHELP or delete QDHELP. Help files created with an unregistered version of QDHELP may not be distributed in any form.

2. Registration

2.1 How To Register

To register send \$49.00 plus \$3.00 S&H to the following address. Overseas orders add \$2.00 for shipping and handling. Checks must be drawn on an American bank.

Phil Allen
2161 Fawnwood Dr. S.E.
Kentwood, MI 49508

Registered users receive technical support via Compserve and phone.

2.2 Bug Reports/Enhancement Requests

Please send all bug reports and enhancement requests to the above address. If you prefer to contact me via electronic means I can be reached at:

Compuserve 72047,2134
internet allen@isgtec.com

2.3 Distribution

Since QDHELP is shareware you are encouraged to distribute it to friends, bulletin boards, information services, etc. as long as all the files in the original are distributed.

XX
XX

3. Getting Started

3.1 Running QDHELP

QDHELP is run from the DOS command line. The command has the following syntax:

```
qdhelP INPUT.QDH [OUTPUT.RTF]
```

The name of the input file is the first argument to the command. The second argument is the output file name and is optional. If the output file name is omitted the default file name input.rtf will be used, where input is the name of the input file (e.g. qdhelP TEST.QDH will produce an output file named TEST.RTF).

To begin you will need to use an ASCII editor to generate an input file for QDHELP. Any editor or word processor that can output straight ASCII text can be used. MS Windows Notepad is a good choice for starting. The default file extensions that QDHELP uses are .QDH for input files and .RTF for output files.

Once you have an input file that you want to generate a help file from simply type:

```
qdhelP FILENAME.QDH [ OUTNAME.RTF ]
```

where FILENAME.QDH is the file name and extension of your input file. QDHELP will process the input file. If no errors are found a file named OUTNAME.RTF will be produced. This is the file that you will run the Microsoft help compiler (HC) on. Your documentation on HC will explain the steps necessary to run HC.

3.2 Testing Your Help Files

To test your help file after compiling it with HC do the following:

- Enter MS Windows and from any application bring up the MS help.
- In help choose the File menu and the Open option. A dialog box will come up. Move to the appropriate directory and choose the help file that you just compiled. This will load your help file for viewing.

If you are going to be working on a help file in one directory for a while do the following:

- Open one of your window groups, select the New option from the File menu in the Program Manager.
- Select the Program Item from the dialog box that comes up. Now in the input dialog input line labeled Command Line enter the winhelp command followed by the directory and name of your help file.

For example, if you have a help file in a directory C:\MYHELP and the name of the help file is TEST.HLP you would enter the following in the Command Line input line:

```
winhelp C:\MYHELP\TEST.HLP
```

□□

```

XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX
XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX

```

This will cause a program icon to be put into your current Windows group which should be a question mark (Like the WinHelp Icon) but with your input filename as the name under the icon. If you double click the icon it will activate winhelp and read in your help file all in one shot. Pretty neat, eh!

3.3 Hints and Tips

Read the WHATSNEW.DOC file in your distribution. This file contains the changes between this and the previous versions of QDHELP. You will definitely want to know about these things.

Read the README file in your distribution. This file contains the latest information that may have not made it into this manual.

QDHELP **is** case sensitive. This means you must enter the commands in lower case except where noted in this manual.

Do **not** put tab characters into your QDHELP text. You can tab to where the text will start for indentation purposes. If you want to put tabs into your text use the RTF command \tab. This will insert a tab into the help file.

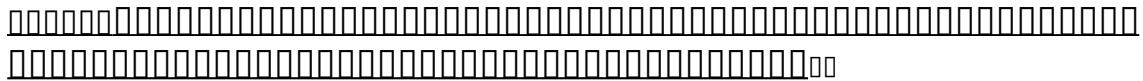
QDHELP comes with extensive on-line documentation in the file QDHELP.HLP. You can load this file into the MS Windows help for viewing. See Section 3.3 Testing Your Help Files. The QDHELP file that generated this help is included in your distribution with the name QDHELP.QDH. This is a good place to start looking to see how to lay out a QDHELP file. To generate QDHELP.HLP from source do the following:

QDHELP QDHELP.QDH

This should output a file named QDHELP.RTF.

HC QDHELP.HPJ

This runs the help compiler and generates the file QDHELP.HLP. This, of course, is the file that you can load and run under MS Windows.



5.1 /hpjoption command

Syntax:

/hpjoptions OPTION, value

The /hpjoptions command is used to set one of the option variables available in the .hpj file used by HC. The valid options are the following:

BMROOT	Root directory for finding bitmaps.
BUILD	Define build criteria.
COMPRESS	Select type of Compression used.
CONTENTS	Select context of contents screen.
COPYRIGHT	Add copyright string to About dialog box.
ERRORLOG	File to write HC compilation messages to.
FORCEFONT	Force use of specific fonts.
ICON	Specify help minimized icon.
LANGUAGE	Sort order for Scandinavian language.
MAPFONTSIZE	Map fonts to different sizes.
MULTIKEY	Select alternate keyword mapping for topics.
OLDKEYPHRASE	Use old keyphrase table.
OPTCDROM	Optimize for CDROM use.
REPORT	Select display of build messages.
ROOT	Root directory to find topic and data files.
TITLE	Specify help window title bar text.
WARNING	Select level of warning messages.

5.2 /hpjfiles Command

/hpjfiles filename

```
/hpjfiles myfile.rtf
```

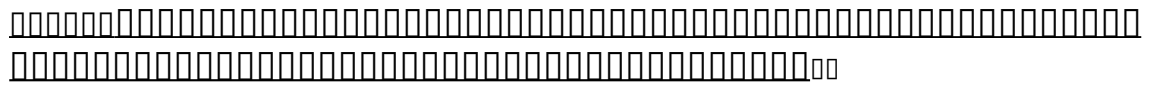
00

5.4 /hpjconfig Command (Win 3.1)

/hpjconfig CONFIG DATA

```
/hpjconfig BrowseButtons()
```

00



5.5 /hpjbitmaps Command

Syntax:

hpjbitmaps c:\path\file.bmp

This command specifies bitmap files to be included in the build. Bitmap files need only be specified if they can not be found in the directories in BMROOT or ROOT.

5.6 /hpjmap Command

/hpjmap TOPIC_NAME Value

The `/hphjmap` command allows you to map a value to a topic name for calling context sensitive help. If you are having QDHELP automatically generate your `.HPJ` file and generate an `.HHH` file (header information on context sensitive help) QDHELP will put something like the following into the `.HPJ` file:

```
[MAP]
#include <helpname.hhh>
```

What this does is automatically pull in the .HHH file being generated by QDHELP. If you are going to use context sensitive help then you might as well let QDHELP do all the work for you!

If you want to manually place a context sensitive value into the .HPJ file use the /hpjmap command. As an example, if you want to be able to call help and get to a topic named EXAMPLE_TOPIC add the following to your .QDH file:

```
/hpjmap EXAMPLE_TOPIC 0x0001
```

If you call WinHelp for context sensitive help and pass a value of 0x0001 for the value, WinHelp will bring up EXAMPLE_TOPIC.

5.7 /hpojalias Command

/hpjalias TOPIC_NAME=ALIAS

□□

5.9 /hpjbaggage Command (Win 3.1)

/hpjbaggage file

□□

5.11 /defformat Command

/defformat format commands

Example:

Once a `/defformat` is in place all text in the document will have the specified formatting attributes applied to it. You can, however, override the `/defformat` commands. For example, if you set the `defformat` to the following:

the font size is set to 20 and the foreground color to color number 5 for the whole document. Now if inside a topic you use another `/defformat` command

/topic

```
/defformat \fs40
```

/endtopic

You will override the document font size of 20 with a font size of 40, but because you did not change the foreground color it will remain color 5. Using this information it is possible to lay your /defformat out so as to make your help writing as easy as possible. If most of your paragraphs in a topic need a space of 100 after them but some need as space of 0, set your /defformat in the topic as follows:

```
/defformat \sa100
```

Now on the paragraphs that need 0 space do the following:

/para \sa000
/endpara

This causes the paragraph to have no spacing after it., which is just what we wanted.

00

```

XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX
XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX

```

5.12 /topic /endtopic Commands

Syntax:

/topic TOPIC_NAME,[BUILDTAG1;BUILDTAG2;...]

...

/endtopic

The topic command signals the beginning of the definition of a new topic. The name of the topic is TOPIC_NAME and it may also include the build tags which this topic should be built for. (If you do not know what build tags are just leave them out for now and all will be well). There are several commands that are only valid from within a topic definition. The following commands can only be found inside a topic definition: /title, /keywords, /browse, /para, and /endpara. You also will find that there are commands which are only legal inside of the /para .. /endpara commands.

Placement:

The /topic commands can be placed anywhere in an input file except inside another /topic command.

Limitations:

Topic definitions cannot be nested.

Example:

More information about the commands used inside the topic command can be found later in this manual.

For information about command placement see section 6.1, Command Placement.

```
/topic TEST_TOPIC
```

```
    /title Test Topic Search Title
```

```
    /keywords test;topic;search;title
```

```
    /browse test:AUTO
```

```
    /para
```

```
    This is the first and last paragraph in the test_topic help topic.
```

```
    /endpara
```

```
/endtopic
```


5.14 /title Command

```
/title search title text
```

Placement:

Limitations:

See the example for the `/keyword` command.

Syntax:

```
/keywords word1[;word2...]
```

The keyword command is used to set the keywords that will find a particular topic. The keywords are used by the Microsoft Windows 3.0 help engine. These are the words you will find in the Search For window of the dialog box that appears when you press the search button while using help.

Placement:

The keyword command is only valid within a topic definition before any paragraphs have been defined.

Limitations:

None.

Example:

/topic KEYWORD_EXAMPLE

/title Copying and Pasting Text

```
/keywords cut;document;Document menu commands;double space;edit
```

/para

This is just some text in an example

/endpara

/endtopic

In the above example five (5) keywords are defined for this topic. The keywords are cut, document, Document menu commands, double space, and edit. In the search command of the MS Windows help engine these three words would be available in the Search For list in the Search window (Figure 1). Choosing one and performing the search would cause the title string "Copying and Pasting Text" to be shown in the Topics Found list in the Search window (Figure 2). Also note that the keyword cut must have also been defined in the topics with titles "Edit Menu Commands" and "Pasting, Copying, and Cutting Pictures".

Search For:

cut	
cut	
document	
Document menu commands	
double space	
edit	

00

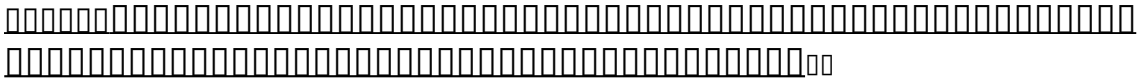


Figure 1.

3 Topics Found

Edit Menu Commands	
Copying and Pasting Text	
Pasting, Copying, and Cutting Pictures	

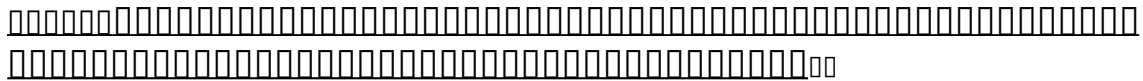
Figure 2.

5.16 /browse Command

00

Using the above defined topics, while topic1 is displayed by the help engine the browse backward key will be off and the browse forward key will be on. Pressing the browse forward key would bring you to topic2. At this point the browse backward key would be on as well as the browse forward key. Hitting the browse forward key again brings us to topic3. At this point the browse backward key is on and the browse forward key is off because this is the last topic in category1.

00



5.17 /para /endpara Commands

Syntax:

/para [paragraph format commands]

...

/endpara

The /para /endpara commands mark the start and end of a paragraph, respectively. There are several commands that are only valid inside a paragraph definition. They are /text, /link, /popuplink, /bitmap and /bitmaplink. For more information see section 6.1, Command Placement.

The paragraph format commands are RTF commands that are passed through to the help compiler unchanged. Below is a list of the most useful RTF commands for paragraph formatting and their meanings. Many commands deal with the unit called twips. A twip is 1/1440 of an inch (that is not very much!).

NOTE: that RTF commands begin with a backslash (\) whereas QDHELP commands begin with a slash (/).

RTF Command	Category	Description
\ql	Justification	Paragraph left justified.
\qr	Justification	Paragraph right justified.
\qj	Justification	Paragraph plain justified.
\qc	Justification	Paragraph centered.
\fiXXX	Indentation	First line indented XXX twips
\liXXX	Indentation	Left margin all lines indented XXX twips
\riXXX	Indentation	Right margin all lines indented XXX twips
\saXXX	Spacing	XXX twips after the last line of the paragraph
\sbXXX	Spacing	XXX twips before the first line of the paragraph.
\slXXX	Spacing	XXX twips space between lines of the paragraph
\brdrt	Border Placement	Border on top of the paragraph
\brdrb	Border Placement	Border on bottom of the paragraph
\brdrl	Border Placement	Border on left edge of the paragraph
\brdrr	Border Placement	Border on right edge of the paragraph
\box	Border Placement	Border on all sides of the paragraph
\brdrs	Border Style	Single line border
\brdrth	Border Style	Single thick line border
\brdrsh	Border Style	Shadow border (try it, you'll like it!)
\brdrdb	Border Style	Double line border
\brdrdot	Border Style	Dotted line border
\keep	Word wrap	Turn off word wrapping.
\keepn	Scrolling Region	Create a non -scrolling region

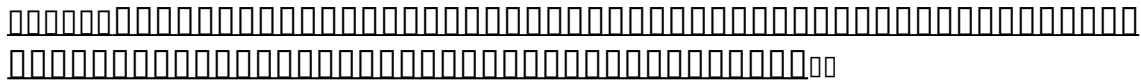
The paragraph format can contain any combination of these. Of course, if you give it two (2) different border styles for one paragraph the results will be unpredictable.

□□

5.18 /topicmacro Command (Win 3.1)

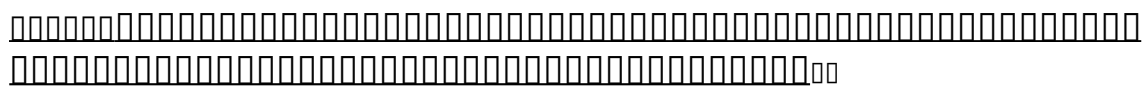
5.19 /helpid

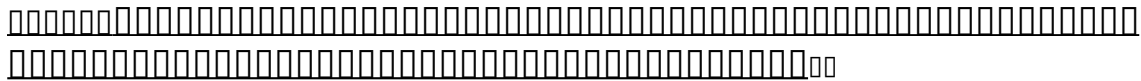
5.21 /link Command



Syntax:

The `bitmaplink` command is another way that you make hyperlinks between help topics. This command will cause a link from the current topic to the `TOPIC_NAME` topic. The link will be activated by pressing the mouse button while over the bitmap that is contained in the file name specified. The bitmap is stored as a standard Windows `.BMP` file. The file name should **not** contain any DOS path information. The path information about where the bitmap file is located on your disk drive should be placed in the `.HPJ` file used by HC. Your Help Compiler manual should answer any questions you have regarding the `.HPJ` file's contents. See the `/link` command for a description of the `,line` option.





Syntax:

The /bitmap command places a bitmap into the help file at a position. The bitmap is in the file named FILENAME.BMP. The file name should **not** contain any DOS path information. The path information about where the bitmap file is located on your disk drive should be placed in the .HPJ file used by HC (MS Help Compiler). Your Help Compiler manual should answer any questions you have regarding the .HPJ file contents. The position of the bitmap is determined by the value placed into the position parameter. The valid values are shown in Table 1.

Table 1.

This command can only appear inside a `/para /endpara` group. If found outside an error will be generated.

Text does not always align the way you would expect when using this command. You must try the command and judge the results on a case by case basis.

/topic BITMAP EXAMPLE

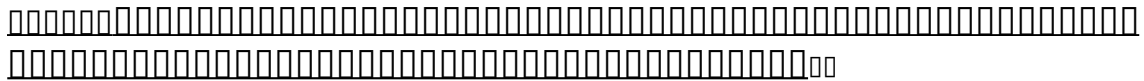
```
/title Bitmap Example
```

```

/para
Left sided bitmap
/bitmap I,MYBITS.BMP
To the left!
/endpara

```

```
/para
Right sided bitmap
/bitmap r,MYBITS.BMP
To the right!
/endpara
```



```

    /para
    before the bitmap
    /bitmap c,MYBITS.BMP
    after the bitmap
    /endpara

/endtopic

In the above topic the same bitmap would be displayed three (3) times. First the bitmap would be
on the left margin of the help topic, then on the right margin, and then in mid sentence. The right
and left margin justified bitmaps do not always come out before the text that is after them in the
paragraph. We are working to find out if this is a limitation of the help engine or QDHELP.
```

```

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

```

5.26 // Command

Syntax:

// comment text

The // is the comment character sequence in QDHELP. Any text found after a // will be ignored. The // must be the first 2 characters on the line.

Example:

```

/topic COMMENT_EXAMPLE

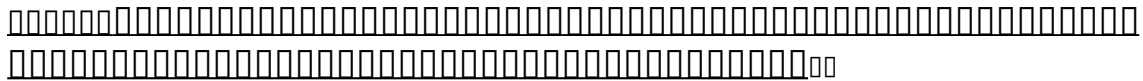
//
// this is the title comment
//

        /title Comment Example

/endtopic

```

This is just a way to add some info to your QDHELP input file that may be worth remembering.



6. Language Structure

6.1 Command Placement Example

Some commands in QDHELP must be positioned in certain places. This section graphically describes the relationship of commands and placement in QDHELP.

The `//` command and the `/include` command can be used anywhere.

```
//  
/include SOMEFILE.QDH
```

The `/topic` command has a specific order in which the commands should be placed in it.

```
/topic TOPIC1
```

```
-----  
|  
|    /title This is the title      | <-- These commands must be between the /topic and  
|    /keywords one;two;three      |         the first /para command. They can be in  
any order.  
|    /browse category,1          |  
|  
-----
```

```
    /para  
    Some text  
    /endpara  
  
    /para  
    -----  
    |  
    | /text \b,BOLD TEXT          |  
    | /link TOPIC2,Go To Topic 2  |  
    | /popuplink TOPIC2,Popup Topic 2 | <-- These commands must be used inside  
the   | /bitmaplink TOPIC2,TOPIC2.BMP |         /para . . . /endpara commands.  
They can | /bitmap c,PICTURE.BMP        |         be used multiple times in any order.  
    |  
    -----  
    /endpara
```

```
/endtopic
```


7. An Annotated Example

11

/text \b\i\fs50,Template #1

/endpara

```
/para ifi200 \sa150
This is template #1 for the QDHELP system. This template
has a large title followed by an introductory paragraph. Following
that come bold topic headings and paragraphs for each
topic.
/endpara
```

/para \sa150
/text \b\fs30,Topic Heading
/endpara

ANNO: This paragraph contains our first /link command
/para

/link SUBTOPIC1,Sub Topic 1
/endpara

/endtopic

11



Glossary

RTF	Rich Text Format
HC	Microsoft Help Compiler
twip	1/1440th of an inch
MS Windows	Microsoft Windows