HLPDK System Overview

ISoft D&M, POB. 5517, Coralville IA 52241, U.S.A

The Help Engine Development Kit is a system that allows you to create hyper-text database help systems with topics and hot links, using a simple text editor and the help compiler.

The help databases can be distributed (free of charge) with the HELPENG or HE help engine display programs.

The help database format is release to the public domain, if you own the Paradox Engine you can use the help databases with your own programs.

Borland Pascal Users can purchase the Help Engine Source Code even for Applications that do not use the paradox engine, using the Native Mem format.

The helpC compiler that is part of this development kit includes the ability to create Windows Help source files from the same source the dos help sources are created!. Write your help once, and create help databases for Dos, Windows, OS/2 and DESQview/X !

Supported Formats : Native(PX), Native (Mem), Win30, Win31, QuickHelp, THELP, TVHC, POPHELP, DESQview/X, OS/2, Text, Word Processors (RTF)

From the same source you created your hypertext database, you can create a text document with automatic table of contents and index sections!

Using the RTF target you can prepare your professional documentation by generating word processors documents, all from the same source!

Contact ISoft D&M if you want to purchase the WINTEXT UI library for Turbo Pascal programs, and use the help engine window in your pascal programs.

Related Topics :
Registration
Contact
File List
Credits
Help Engine
Help Compiler
Quick Start
Related Products

Registration

The Help Engine Development Kit (HLPDK) is a shareware product, if you find this product valuable, please register it. This section describes the reasons you should register.

By registering you will receive a complete discussion of the help engine features, and many examples of usage. You will receive a diskette with the latest HLPDK version, and DPMI (DOS protected mode) versions of the help compiler and engines!. (The memory version of the help engine can display huge databases only in the protected mode version).

Your registration will help us to create the next versions of HLPDK, that will include more options, and features, some of them might even be your enhancement requests!.

A point to consider - If you want to create Windows Help source files, you will have to purchase a word processor that can output RTF files, the WORD for Windows 2.0 package, for example, is listed at around $500!.

To order your registered copy of HLPDK please do one of the following:

1.   Order from ISoft D&M:
You can order directly from ISoft D&M. Please refer to the supplied
HLPDK.REG file. Edit it, print it and mail it (with a check) to:
        ISoft D&M,
        POBox 5517
        Coralville Iowa 52241, U.S.A

2.   Order from PsL:
You can order from Public (software) Library with your MC, Visa, AmEx
or Discover card by calling 800-242-4PsL (from overseas: 713-524-6394) or
by FAX to 713-524-6398 or by CompuServe to 71355,470.
These numbers are for ordering only. ISoft D&M can NOT be reached at those
numbers. Please contact ISoft D&M directly for any other type of problem
(technical or other), information or request.

Related Topics :
Contact
Overview

CONTACT

Please contact :

ISoft D&M,
P.O.B 5517
Coralville IA 52241,
U.S.A

ISoft D&M e-mail address : CompuServe - 76350,333

To contact the author directly :

Contact :   Loewy Ron,
9 Haneveem st.
Herzeliya, 46465
ISRAEL.

e-mail address : CompuServe - 100274,162

Related Topics :
Regsiteration
Overview

HPLDK Distribution File List

This package contains the following files in the following archives :

Archive A :

```
READMEA.HDK      - File List for Archive A.
HELPC.EXE        - Help Compiler.
HLPDK.DOC        - This file.
HELPC.DOC        - Help Compiler documentation
HLPDK.REG        - Registration file.
HLPDK.HDF        - Help Development Kit hypertext help database source.
HLPDK.HLP        - WinHelp Version of the hlpdk hypertext source.
HLPDK.INF        - OS/2 IPF Version of the HLPDK hypertext source.
PROGRAMS.TXT     - ISoft D&M shareware products description.
HELPDEMO.HDF     - Demo Help Project.
2NDDEMO.HDF      - Demo Help Project - file 2.
```

Archive B :

```
README B.HDK     - File List for Archive B.
HELPENG.EXE      - Help Engine (Native Paradox).
HE.EXE           - Help Engine (Native Memory).
HELPDEMO.HLP     - Demo Help Project file for Windows WINHELP.
HLPDK.HDT        - Help Development Kit help database file. (Native Paradox)
HLPDK.SET        - Help Development Kit help database file.
HLPDKTOP.*       - Help Development Kit help database files.
HLPDKLNK.*       - Help Development Kit help database files.
HLPDKKEY.*       - Help Development Kit help database files.
```

Related Topics :
Overview
Quick Start

CREDITS

HELPENG, HELPC were written using Turbo Pascal 6.0, and Borland Pascal 7.0, Paradox Engine 2.0, and Paradox Engine 3.0, 3.01. (Trademarks of Borland International).

Windows, Microsoft, HC and Quick Help are trademarks or copyrights of Microsoft Corp.

The HELPENG program was written using the WINTEXT UI library, (c) 1991,93 Loewy Ron.

THELP, HL and TVHC are Trademarks or copyrights Borland International.

POPHELP is a copyright of TurboPower Software.

TVHC support, and other ideas were suggested by MunLeong Chan.

Parts of the Help Engines were generated using Ron Loewy's WTGEN and dbGen CASE code generators.

Parts of the Help Compiler were generated using Ron Loewy's dbGen database code generator.

Yaniv Golan helped me with beta-testing, debugging and suggestions. He is also the   author of the HPCHDK and POPHDK packages.

John Murphy notified me of the paragraph formatting difficulties that lead to the .PARAGRAPH .END-PARAGRAPH formatting capabilities.

DESQview/X is a trademark of Quarterdeck Office Systems.

MDVXHELP is a copyright (1992) of Daniel J. Bodoh. I have used his excelent documentation of DESQview/X help format to generate code to this environment.

The HC31 problem with characters with ordinal values biggen than 127 was solved thanks to Mr. Lars Fosdal of Norway.

OS/2 is a registered trademark of International Bussiness Machines.

The Text Cursor restore problem in the help engines was reported by Prof. Timo Salmi of the Univ. of VAASA, Finland.

D.J. Murdoch suggested the Cross Reference Report enhancements, and winhelp Go To Search list box title problem.

Word for Windows is a trademark of Microsoft corp.

WordPerfect, WordPerfect for Windows are trademarks of WordPerfect corporation.

Lotus, AMI PRO are trademarks of Lotus corp.

Reuben Halevi from ISoft D&M did a lot to make this product real.

Slren Pingel Dalsgaard notified me of the small scroll box bug in helpeng, as well as some spelling mistakes in helpC.

Related Topics :

<u>Overview</u>

Quick Start

From the command line enter HELPC HELPDEMO to compile the demo help database.

After the compilation process is ended, enter HELPENG HELPDEMO to browse the demo help database.

If you want to compile for windows :

from the command line enter HELPC HELPDEMO /W31
and after that HC31 HELPDEMO
(HC31 is the Microsoft Help Compiler)
from windows, open the HELPDEMO.HLP file from the windows help engine. (Notice - if you do not have the HC help compiler from windows, we have included a compiled version of HELPDEMO (HELPDEMO.HLP) so you will be able to feel the quality of the output that can be generated, and decide if you want to purchase the HC31 program (Included with the SDK, BC++, TPW, BP and probably some other packages), or download it from one of the sites Microsoft had uploaded it to).

If you use windows 3.0, or want your help file to be compatible with winHelp 3.0 use :
        HELPC /W30 HELPDEMO
        HC HELPDEMO

To compile for THELP use :
        HELPC /TH+ HELPDEMO
        HL -x -e100 -p -i -oHELPDEMO.TPH HELPDEMO.HL

To compile for QuickHelp :
        HELPC /QH+ HELPDEMO
        HELPMAKE -e15 -oHELPDEMO.QH HELPDEMO.SRC

To compile for TVHC :
        HELPC /TV+ HELPDEMO
        TVHC HELPDEMO.TXT

To compile for POPHELP :
        HELPC /PH+ HELPDEMO
        MAKEHELP /Q HELPDEMO

To compile for DESQview/X :
        HELPC /XD+ HELPDEMO
        MDVXHELP HELPDEMO.HLP @HELPDEMO.LST
        COPY HELPDEMO.HLP d:\DVX\HELP

To compile for Native Mem Format :
        HELPC /MT+ HELPDEMO
(Use HE HELPDEMO to view that database).

To compile for IPF (OS/2) format :
        HELPC /OS2 HELPDEMO
        Switch to an OS/2 session.
        IPFC /INF HELPDEMO
(Use the VIEW Command to browse the database).

To create a text document with table of contents, glossary and index :
        HELPC /TXT HELPDEMO

To create a word processor document with table of contents,
glossary and index :

      HELPC /RTF HELPDEMO
      import helpdemo.rtf to your word-processor, and update fields.
      (e.g. Microsoft WinWord : File|Open to open the document
                        Answer Yes to convert from Rich Text Format
                        Press F9 to update embedded format fields
                        somewhere in the first page (after the
                        Table Of Contents title), and in the last
                        page (after the index title).)

Related Topics :
Overview
File List

The Help Engine

To start the help engine program enter HELPENG on the command line, and choose the help database you want to view from the list box. Use the Tab key to cycle between hot-link words, and press Enter to go to that link.

If you know the name of the SET file of the database you want to activate, you can enter it on the command line, and the help engine will display it.

If you have a mouse, you can use it to point at the hi-lighted links, and activate them by pressing the left mouse button.

Press the ALT-T key combination to choose from a list of TOPICS in the database, Press ALT-B to go back to the previous displayed TOPIC, Press ALT-X to exit the help engine.

The exit command can be given by pressing the right mouse button.

Use F10 to open the menu bar, and use the arrow keys to navigate the options. When you find the option you are interested in, Use the Enter key to execute it, or the Esc key to cancel. The menu bar can be navigated using the mouse as well.

To create a new help database please refer to the HELPC documentation file.

If you want to use the Native Mem Format Databases, Please use the supplied HE.EXE program.

Related Topics :
Trouble Shooting
Overview

Trouble Shooting

If you receive some kind of an error message while using the help development kit please refer to this section.

tableOps: xxxx error message :
This message is generated by the Paradox Engine if an internal error occurs, if you receive a message that is not understood (e.g. Table is busy), please try one of the following :
 1. Try running the Help Compiler or the Help Engine with another input and
     notice if the problem exists, if it does, turn to the next entry,
     otherwise try to see what the difference between the inputs is.
 2. Check your free memory, try to move some TSR's from memory, or use the
     protected mode compiler/engine available with the registered version.
 3. Check if you have loaded dos SHARE program, it is advised to use SHARE
     with programs that use the Paradox Engine. If you receive a table busy
     error, try to see if you have another program that uses the same table
     in another session (Windows/DESQview etc..).
 4. PANIC - Please contact the distibuter/author for additional help.

Microsoft's HC31 help compiler will generate a error in RTF file error :

 1. Re-compile using HELPC 4.0+. This should solve the problem, if the problem
     still exists - please try steps 2 and 3, and notify the author.
 2. Use a binary disk editor to look at the character at the position x - 1
     in the RTF file (where x is the error position reported by HC31).
 3. Try to replace that charqacter from a special PC oem character to a
     "standard" ascii character (such as *). If the problem still occurs,
     please contact ISoft D&M/The author, otherwise - you probably came
     accross a character that HC31 can not handle. In that case - try using
     HC300b (the one that generates winHelp 3.0 format), or delete these
     characters from your source HDF file.

Borland's HL help linker will generate a FE02: Out Of Memory error :
 1. Re-compile using HELPC V3.0+, to include ;INDEX directives after the
     screen's definitions.

Borland's HL help linker will complain about undefined context numbers :
 1. You probably have a contents.h file in the current directory, that hl
     tries to use to bind the help topics to Borland's IDE on-line help,
     try renaming this file, and re-compile with HL.

HelpC will not create the database when compiling to Native Mem (/MT+) :
 1. Try compiling with report generation on (/R+) to Native Paradox
     (/PX+) format. If there are errors, consult the HELPC.LST file, fix
     them, and re-try.
 2. If the database is BIG, you might not have enough memory - Remember that
     Native Mem tables are generated in memory. You will have to use the
     HelpCX protected mode compiler (Available with the registered version).

HelpEng/HE will complain on an old-database when trying to open a help
database with it :
 1. Re-Compile the database using helpC.

HelpEng/HE will complain on un-supported database format, when trying to

open a database :
 1. HelpEng can not display Native Mem databases, and HE can not display
    Native PX databases. Use the other help engine supplied.

HE will not be able to open a Native Mem database :
 1. Use the protected mode HEX help engine. (Available for registered users).

Where is HL - Borland's Help Linker :
 1. HL is not included with Borland C++ or Borland Pascal, Borland sells an
    additional package called Borland Open Architecture (and Borlad OA for
    Pascal) that include among other things - the THELP help linker.

HelpEng/HE will complain on an old-database when tring to open a help database with it :
 1. Re-Compile the database using helpC.

HelpEng/HE will complain on un-supported database format, when trying to open a database :
 1. HelpEng can not display Native Mem databases, and HE can not display
    Native PX databases. Use the other help engine supplied.

HE will not be able to open a Native Mem database :
 1. Use the protected mode HEX help engine. (Available for registered users).

We have encountered some users that ran into problems, because they did not use the dos SHARE
program. We advise that you will always load the SHARE program in your AUTOEXEC.BAT file.

Related Topics :
Overview
Help Engine
Help Compiler

Help Compiler

The HELPC help compiler is a dos command line program that receives a help definition file (.HDF) and produce a help database.

   The helpC command line is :

 HELPC source.fil [/NT:topic-table-name]
                  [/NL:link-table-name]
                  [/ND:data-file-name]
                  [/NK:keywords-table-name]
                  [/D:conditional-define]
                  [/ED=P|C|+]
                  [/EL=x]
                  [/R-]   [/ST-] [/CS+] [/EP+] [/WD-] [/RM-]
                  [/W+]   [/W30] [/W31] [TH+] [/QH+] [/TV+] [/PH+] [/XD+]
                  [/PX+] [/MT+] [/OS2] [/TXT] [/RTF]
The output of the compiler is a set data file called source.SET, that includes the names of the other help database components.

Please Notice - if the /W+ /W30 or /W31 switch is defined, the compiler will generate a .HPJ and .RTF files that can be used with the Microsoft HC (v3.00b) or HC31 help compiler, and output a .HLP windows help file!.

By default the help compiler produces a help cross reference report on the topics/links used in the database, if the /R- switch is set, no cross-ref report will be generated, and no database integrity indicators will be given.

IMPORTANT - Scan the output cross reference report for links defined, that are not defined as topics!

Notice - An additional switch is available that instructs the help compiler to indicate it's parse and code generation progress. Use the /EP+ switch to see that the helpC compiler is active. Please understand that the compile process can take a long time, on complex help databases. This switch does nothing more that ensure to you that the help compiler is alive.

The /ST switch defines if the topic name defined in the .TOPIC command can include embeded spaces. By default this switch is set to /ST- (No embedded spaces).

The /CS switch turns Topics and Links case sensativity to ON. Use this switch with care - Only if you must have a lower/mixed case index in THELP or POPHELP use this switch. Notice that help databases that compile fine with case sensativity set to off, might fail to do so in case sensativity on. By deafult this switch is set to off (-).

The /XD switch generates code to the MDVXHELP help compiler, available free in the file DVXHLP10.ZIP on the Internet. When this switch is given, a set of cover and section files is generated, as well as a response file that should be used to compile the help database.

The /OS2 switch generates IPF source code. You must compile the generated code with IPFC, from within an OS/2 session (window or full screen).

The /WD switch controls Warning Display. By default Warning Display is On, if you do not want to see the Warnings issued by the compiler when a topic name or link are found to be incompatible with the target source, and helpC translate them to be syntax compatible with the target.

Use the /RTF switch to create a word-processor ready RTF file, import this file (and update the fields e.g.- F9 in WinWord) to create a professional formatted documentation from the same source code!.

Use the /TXT switch to create a text file document from your .HDF source. With this feature you can create on-line help and printed documentation from the same source!.

The /RM- switch uses disk cross-reference report tables. Usage of this option will allow the creation of cross-reference report for huge databases. This options will slow the creation of reports, use it only for databases so big that the report can not be generated using memory tables.

The /ED= switch creates the external references (if ant are defined) in one of the following formats : /ED=P (Default) creates the external definitions in a pascal unit, /ED=C creates the external definitions in a C header file, and /ED=+ creates a C++ header file.

The /EL= switch sets the Error Level used. This switch can be overriden by the .ERRORLEVEL statement. The acceptable error levels are :

0	No Check	- (Default) - No action is taken.
1	Trap	- The link is replaced to a special topic named ERROR.
2	Strip	- The link is replaced by the Text part only.
3	Delete	- The text and link parts are deleted.
9	Abort	- helpC will abort with an error message, indicating the line number.

Please refer to the Exception Handling section below about helpC error processing.

Related Topics :
Overview
Command Line
Help Engine
Technical Documentation

Command Line

The compiler switches are defined on the command line. These switches can be placed before and after the source file name. The compiler will look for a file named HELPC.CFG in the current directory, and process it before any command line switches are processed.

Switches on the command line will over-ride switches set before them, on the command line or in one of the response options file.

Response option files can be specified by prefixing their name with a @ sign.

Examples :

If our HELPC.CFG file, in the current directory is :

/W31
/R-

And we compile using HELPC HELPDEMO

The helpdemo.hdf file will be compiled, with the target set to winhelp 3.1
and no cross-reference report.

Compile using HELPC HELPDEMO /r+

Will create winhelp 3.1 sources, but a report will be generated.

Compile using HELPC /w- HELPDEMO

Will generate native format help database, no report is generated.

If we have a file called MYSWITCH.CFG with the following lines :

/TH+
/ST-
/CS+

And we compile using HELPC @MYSWITCH.CFG HELPDEMO

A THELP source, using Long Topic Names, with case sensitivity will be
generated, no cross reference report is generated. (HELPC.CFG contains /r-)

Related Topics :
Overview
Help Compiler

Technical Documentation

The Help Engine database is a combination of text that is read by the user, and a Topic and Links database that allows the user to view the topics he is interested in, using hyper-text methods.

This document describes the HLPDK language syntax, HELPC compiler operation and the native databases structure.

Related Topics :
Help Compiler
Help Engine
Help Definition Format
Native Format Structure

Native Database Structure

Each help database is built of 5 components :

1. Help Data File   - this file contains the raw text data.
2. Topic Index   - a database table that contains the topic names, and
                         the references to the topics in the Data File.
3. Links Index   - a database table that contains the list of links
                         defined for each topic, and positional data.
4. Keywords Index   - a database table that contains the list of keywords,
                         and the topics that have these keywords.
5. Set Data File   - a database configuration file that contains the
                         names of the 3 components defined above, and some
                         additional information.

Version 1.x is the bare-bones version that includes only the basic operations needed from a hyper-text database, this version is implemented using the Paradox Engine V3.0 to manage the Topic and Link index tables.

Version 2.x has a new field in the topic table - the title field, help databases that were compiled with V1.x will not be executed by version 2.x, they will have to be re-compiled!.

Version 3.x fields have different length - the databases have to be re-compiled.

Version 5.0+ has a new entry in the help set file, database have to be re-compiled, or the value 128 should be added in the last line of the help set file, and the version number should be changed to 5.0.

Version 6.0+ has added the keyword support - a new database table is added, and a new set format is defined. Help databases will have to be re-compiled for Natice targets.

In Version 7.0 the title field has been expanded once more - to 70 characters. Help databases will have to be re-compiled.

In Version 8.0 the tType field was added to the Topics database. Old databases will have to be re-compiled.

The format of the help database is released to the public domain, if you implement a database viewer on this database structure, I would be happy to receive a copy ..

Related Topics :
Help Compiler
Help Engine
Help Definition Format
Topic Index Format
Links Index Format
Keywords Index Format
Set File Format

Help Definition Format

The help definition file format is as follows :

```
| remarks ...
| remarks ...
|
| .ENTRY topic-name
| [.TITLE optional title for topic]
| [.TITLEATTR paragraph attributes]
| [.KEYWORDS optional keywords for topic]
| [.GROUPS group1 [group2 ..]]
| topic text and link reference ..
| topic text and link reference ..
| topic text and link reference ..
| topic text and link reference ..
| .END-ENTRY
|
| remarks ...
|
|
| .ENTRY topic2-name
| [.KEYWORDS keyword1 keyword2 ..]
| [.GROUPS group3 [group2 ..]]
| [.PARAGRAPH [paragraph attributes]]
| topic 2 text and link reference ..
| topic 2 text and link reference ..
| topic 2 text and link reference ..
| [.END-PARAGRAPH]
| additional topic2 text ...
| .END-ENTRY
|
| .POPUP popup-name
| [.TITLE optional title for popup]
|     definition for some reference
|     ...
| .END-POPUP
```

As can be seen - lines between the .ENTRY and .END-ENTRY keywords are the lines processed by the help compiler, to create the help database, every other line is considered a remark, and is not parsed or evaluated by the help compiler.

The topic text and link reference lines are simple text lines, with special link marks.

A sample will provide the details :

This is a text help line that includes one ~link~link-topic-name~ in it.

This line will be displayed in the help engine program as :

This is a text help line that includes one link in it.

The word "link" will be in a different color then the rest of the line,
to indicate that this is a link. If the user hi-lights the link, and
presses Enter (or points the mouse cursor and clicks), the help engine will
look for a topic called "link-topic-name", and display it.

   Browse the HELPDEMO.HDF file for a demo of a help database definition file.

Notes :

1. The keywords .TOPIC and .END-TOPIC can be used instead of .ENTRY and
   .   .END-ENTRY respectively.

   An equivalent pair of keywords are .TOP and .ENDTOP .

   You can use .EXT-TOPIC instead of .TOPIC/.ENTRY etc.. - topics created
   with that statement are the same as regular topics, only - an external
   reference is created for them to be linked to a pascal/C/C++ program.

2. Keywords are NOT case sensitive. (.topic = .TOPIC = .eNTry)

3. The .TITLE keyword is optional. If no title is defined, the topic name
   is used as a title. The .TITLEATTR keyword is used to describe the
   title text attributes. The syntax is the same as the .PARAGRAPH text
   attributes. Please refer to the Text Attributes section below for further
   discussion of the optional paragraph attributes.

4. If you want the ~ sign to appear in your help database, use ~~ in your
   help source file. (e.g. "This is a ~~" will be "This is a ~" in the final
   help database.).

5. The Optional .PARAGRAPH and .END-PARAGRAPH commands in the help topic text,
   specify the start and the end of a paragraph. This feature is currently
   supported only for winhelp, RTF and IPF sources. This feature is important if you
   want to create long paragraphs that will wrap correctly when a user resizes
   the help window. Please refer to the Text Attributes section below for further
   discussion of the optional paragraph attributes.

6. When a .PARAGRAPH is issued, a .END-PARAGRAPH MUST be specified at the
   end of the paragraph. These commands must appear as the first words in
   a line. The text after the .PARAGRAPH command is NOT part of the text
   of the help topic.

7. The optional .KEYWORDS command can appear after the .TITLE command
   (If there is one), or the .TOPIC header (otherwise), Use this command
   to help the user search for sets of topics that are connected to a
   specific keyword. The syntax of this command is :
   .KEYWORDS keyword-1 keyword-2 ...

8. .POPUP and .END-POPUP (or .ENDPOP) are used to define popup topics that
   are displayed on the same screen as the original topic that includes the
   link to them. Popup entries allow the user to consult a glossary, and
   not lose the topic he was reading. Popups are supported in the winhelp,
   OS/2 IPF, Native PX and Native Mem formats. In the other formats Popups are
   used as standard topics.

9. In the Text target .POPUP entries will be automatically included in a Glossary file that you can add as an appendix to your documentation.

10. The .GROUPS statement associate a topic with several groups, you can create topics with menus to groups of other topics using the statement ".INSERT GROUP group-name" in your topic text. Please refer to the Groups section of this document for additional discussion of this function.

Additional commands include :

| .TEXTCOLOR, .LINKCOLOR and .LINKHIGH - VGA colors of text and links in normal and high (selected) states in the native help format.

The syntax of these commands is :

| .TEXTCOLOR w

or

| .LINKCOLOR x

or

| .LINKHIGH y

Where w, x and y are the color of the text/link, which is a combination of the color of the foreground with the color of the background multiplied by 16.

The colors that can be used for both fore and background are :

```
Black     = 0
Blue      = 1
Green     = 2
Cyan      = 3
Red       = 4
Magenta   = 5
Brown     = 6
LightGray = 7
```

The colors that can be used only for the foreground are :

```
DarkGray     = 8
LightBlue    = 9
LightGreen   = 10
LightCyan    = 11
LightRed     = 12
LightMagenta = 13
Yellow       = 14
White        = 15
```

For example : to set the normal color of a link to light green on blue, use

| .LINKCOLOR 26

where 26 = 10 + 16 * 1. (10 = light green, 1 = blue).

Creating Multiple Source Projects :

Use the .INCLUDE commands in the source of your .HDF file, to include an
additional source file. .INCLUDE is not supported in the middle of a topic.

For Example :

 .. some stuff

| .include 2ndfile.hdf

 .. additional stuff

will process 2NDFILE.HDF in the middle of our source file.

Please notice - .INCLUDE commands can be nested in your source file,
it is however not recommended to nest them too deep, helpC does not
close the input file when a .INCLUDEd file is processed, in order to
minimize help generation time - there is however, a limit on the number
of open files that can be used by a program - this is a limit set by dos.

Use the .INSERT CODE statement to insert target code that will not be
translated by helpC. helpC will send un-translated code to the output target
until a .END-CODE statement is found.

For example - creating a small table in winhelp format :
| .. normal code
| .ifdef windows
| .insert code
| \pard{\tab column1\tab column2\tab column3}\par
| \pard{\par col1data\tab col2data ..}\par
| .end-code
| .endif

GUI Font based targets :

The .FIXEDFONT command is used to define a non proportional font as the
default font of the help database. Currently the only targets affected
by this command are windows winHelp v3.x. If you find that your help
databases are not aligned after they are converted to winHelp format, use
this command and re-compile. This command has no effect on targets that are
different from the targets mentioned before.

The .PCFONT command is used to define a font that has the standard pc
character set (Terminal - in windows). If you convert a text based
database that uses the PC's special characters, use this command to
set the default font. This command has no effect on targets that are
different from the targets mentioned before.

Text targets :

The .PAGELENGTH and .PAGEWIDTH statements set the printed page length in
rows, and width in columns. The syntax is .PAGELENGTH xx or .PAGEWIDTH yy .

Use the .SETFONT command to change or add fonts to the font palette, The

syntax is : .SETFONT font-number 'Font-Name-In-The-Target-Language'. If
for example we would like to change font number 3 in the windows target
to be a font X of family Y we would write :
| .SETFONT 3 'Y X'

Use the .TITLEDEFAULTS statement to set the default text attributes of the
topics titles in the help database. These defaults can be over-ridden with
the .TITLEATTR statement in the topic definition. Please refer to the Text
Attributes section below for discussion of paragraph text attributes.

Use the .ATTRDEFAULTS statement to set the default text attributes of
paragraph text in the database. Please refer to the Text Attributes section
below.

Use the .ERRORLEVEL command to set the error level. The syntax is .ERRORLEVEL x
where x is one of the following :

    0   No Check   - (Default) - No action is taken.
    1   Trap   - The link is replaced to a special topic named ERROR.
    2   Strip   - The link is replaced by the Text part only.
    3   Delete   - The text and link parts are deleted.
    9   Abort   - helpC will abort with an error message, indicating the
                        line number.
Please refer to the Exception Handling section below about helpC error processing.

Use the .ALIAS command to substitute a popup/topic name with another name. The syntax is .ALIAS
newname origname. Whenever a newname link is found, the alias of origname is used. This can be very
useful when you want to give multiple names to a topic, or when HLPDK source is the target of an
incremental compiler.

Related Topics :
Technical Documentation
Conditional Defines
Groups
Text Attributes

Conditional Defines

The helpC compiler supports conditional define directives that allow you to tailor your source according to define symbols that are known to the compiler.

You can use the .ifdef, .ifndef, .else and .endif directives to test the target of your compile and do different things according to this target.

If for example you have a Windows bitmap you want to include in your generated RTF file, when you compile to winhelp format, but, you would like to use a statement in other targets that do not support graphics, you could do the following :

```
| .ifdef windows
|    some text {\bmc mybmp.bmp} some more text
| .else
|    some text AND STATEMENT HERE some more text
| .endif
```

When the compiler will generate the code, the RTF command \bmc will be inserted in the output of the compiler for windows targets (/W30, /W31), and a simple text statement will be inserted otherwise.

The conditional directives supported are :

```
| .ifdef   SYMBOL - Compile code only if SYMBOL is defined.
| .ifndef SYMBOL - Compile code only if SYMBOL is NOT defined.
| .endif          - Any .ifdef/.ifndef should be terminated by a .endif .
| .else           - Reverse the previous .ifdef/.ifndef
```

The following SYMBOLS are automatically defined by the compiler :

```
Symbol           When Defined
------           ------------
DEF              /PX+
W31              /W31
W30              /W30
THS              /TH+
QHS              /QH+
TVH              /TV+
PHS              /PH+
DVX              /XD+
MEM              /MT+
OS2              /OS2
TXT              /TXT
WINDOWS          /W30, /W31
THELP            /TH+
QUICKHELP        /QH+
TVHC             /TV+
NATIVEPARADOX    /PX+
POPHELP          /PH+
DESQVIEW/X       /XD+
NATIVEMEMORY     /MT+
VERx.y           When Compiling with helpC Vx.y
NATIVE           /PX+, /MT+
DOCUMENT         /TXT, /RTF
```

You can define your own conditional defines on the command line using the /D: switch of helpC.

e.g.

Source :

```
| ... some text ..
| .ifdef mydef
|    Special Text for mydef situation
| .endif
| ... more text
```

and you compile using HELPC /D:MYDEF the special text will be included.

Related Topics :
Technical Documentation
Help Definition

Text Attributes

Topic's text can receive visual attributes such as a font, size, bold, italic etc.. by using the paragraph attributes on the .PARAGRAPH (.par) line. Text attributes are currently supported in the winhelp and rtf targets.

The attributes that can be applied to text are :

Font   - The font the text will be displayed in.
Bold   - The text is displayed in bold characters.
Italic   - The text is displayed in italic characters.
Underline   - The text is displayed in underline characters.
Size   - The size of the font.
Box   - The paragraph can be contained in a box.
Align   - The paragraph can be alligned.
Color   - The text foreground color.

The syntax is :

| .PARAGRAPH [attr1 attr1-parameters [attr2 attr2-parameters [..]]]

The allowed parameters to attributes are :

1. FONT font-number-in-font-palette : The font number in the font palette, there
is a standard font palette for winhelp and rtf targets that include the
following fonts :

 0 : Times Roman   (roman Tms Rmn)
 1 : Symbol   (decor Symbol)
 2 : Helvetica   (swiss Helv)
 3 : LinePrinter   (modern LinePrinter)
 4 : Terminal   (roman Terminal)
 5 : Times New Roman   (roman Times New Roman)
 6 : Arial   (swiss Arial)
 7 : CG Times (WN)   (roman CG Times (WN))
 8 : Courier   (modern Courier)
 9 : Modern   (modern Modern)
10 : Script   (script Script)
11 : Univers (WN)   (swiss Univers (WN))
12 : Wingdings   (nil Wingdings)
13 : MS Sans Serif   (swiss MS Sans Serif)

2. BOLD ON|OFF - Set bold mode to on/off.

3. ITALIC ON|OFF - Set italic mode to on/off.

4. UNDERLINE ON|OFF - Set underline mode on/off.

5. SIZE font-size - Set the size of the font.

6. BOX NONE|NORMAL|THICK|SHADOW|DOUBLE - Print paragraph in a box which has one
of the described borders : NONE is no border at all.

7. ALIGN LEFT|RIGHT|CENTER|JUSTIFY - Print the text in the paragraph aligned

to one of the sides, the center or justified in the paragraph.

8. COLOR color-number-in-color-palette - The numbers are the same numbers used
for dos native systems and described in the help definition format section
above, regarding the .LINKCOLOR, .TEXTCOLOR and .LINKHIGH statements.

Example :

| .PAR BOX SHADOW FONT 13 BOLD
| ... some text here ..
| .ENDPAR

Will display the paragraph in a shadow bordered box, using font 13 in the
palette (MS Sans Serif assuming that the original palette is used) with bold
characters.

Related Topics :
Technical Documentation
Help Definition

Groups Definition and Usage

Topics can be associated to groups using the .GROUPS statement in the topic heading. Menus to these groups can later be created by using the command .INSERT GROUP group-name in the topic text.

An example will clarify the use of this function :

Let us assume that we have a car's collectors database, where the description of each car is a topic. We would like to present a hierarchy to the user that looks like that :

```
Overview Screen
   |
   +--------- Cars by Country
   |              |
   |              +------------ U.S.A ... List of cars manufactured in U.S.A
   |              +------------ U.K.
   |              +------------ Italy
   |
   +--------- Cars by Maker
   |              |
   |              +------------ Alfa-Romeo ... List of Alfa-Romeo models
   |              +------------ GM
   |              +------------ Mazda
   |
   +--------- Cars by type
                  |
                  +------------ 4x4 & Jeeps
                  +------------ Sport Cars
                  +------------ Family Sedans
```

We will create group references in the topics using the .GROUPS statement, The Alfa Romeo Spider 2000 topic will look like :

```
| .topic alfa_romeo_spider_2000
| .title Alfa Romeo Spider 2000
| .keywords alfa spider convertible sport classic
| .groups italian alfa sport
|
| The Alfa Romeo ...
|
|
| .end-topic
```

The Italian Cars Directory topic will be :

```
| .topic italy_directory
| .title Italian Cars Directory
| .keywords directory
|
| Cars Manufactured in Italy :
|
| .insert group italian
```

|
| .end-topic

If our database includes 3 cars that have a reference to the Italian group
this will be the equivalent of writing :

| .topic italy_directory
| .title Italian Cars Directory
| .keywords directory
|
| Cars Manufactured in Italy :
|
| ~Alfa Romeo Spider 2000~alfa_romeo_spider_2000~
| ~Ferrari 308 GTS~Ferrari_308_gts~
| ~Fiat Uno 70S~fiat_uno_70s~
|
| .end-topic

Related Topics :
Technical Documentation
.HDF Syntax
Conditional Defines

Exception Handling

helpC provides mechanism to handle link exceptions. An exception is a link to an identifier that is not defined as a topic or popup. In order to help you find these errors helpC contains the cross reference report that can be produced using the /R+ switch. You can browse the HELPC.LST file generated by helpC if a cross reference report is generated, and search for the string ERROR that describes undefined links.

helpC provides additional exception handling facilities. When an error is found during compile helpC can perform on of the following actions :

0: No Check - Default - helpC does not do anything. The link is passed as is to
the target. The target help engine/compiler will handle the
exception as it is implemented.

1: Trap   - helpC will replace the link key with the key ERROR. It is the
responsibility of the help database author to provide such a
topic/popup.

2: Strip   - helpC will pass the text of the link to the target with no link
support. The text will appear as normal text.

3: Delete   - Both the text and the link will be deleted and will not appear in
the target.

9: Abort   - helpC will abort, with an error message, and the line the link
is defined. This mode is recommended when you are debugging
your help database.

You can set the error level using the following techniques :

- Use the /EL=x switch on the helpC command line.

- Use the .ERRORLEVEL x .HDF statement.

Important Points :

1. When errorLevel = 0 (default) no topic database is created during pass 1, if you set .errorlevel somewhere in your source, after some topics/popups have been defined, these topics will not be in the symbol table created by helpC and will be considered an exception!.

Related Topics :
Technical Documentation
.HDF Syntax
Help Compiler

Topic Index Format

The topic index table format is as follows :

```
TOPIC       - 40 characters string key - this is the topic "name".
FILENAME    - 38 characters string field - includes the name of the
              help raw data file. (Notice - V1.0 ignores the value of this
              field, it is here for future expansion).
START        - longint (long) field that contains the offset from the data file
              start, to the start of the topics text data, in bytes.
LEN          - longint (long) field that contains the length (in bytes) of the
              text data of the topic.
TITLE       - 70 characters string key - the title of the topic.
TTYPE        - 1 character - topic type. Currently supported types are :
              'T' = Topic.
              'P' = Popup.
```

Related Topics :
Technical Documentation
Links Index Format
Keywords Index Format
Set File Format

Links Index Format

The links index table format is as follows :

TOPIC   - 40 characters string key - the name of the topic that
            has the link.
SEQ   - integer (int) key - contains the sequential number of
            the link in the topic. (This way you can assign the links
            order in the topic).
LINK   - 40 characters string key - the name of the link topic to
            display if the link is chosen.
LINE   - integer (int) field - the line (from the start of the topic)
            this link is displayed on.
COL   - integer (int) field - the column this link is displayed on.
TXT   - 70 characters string field - the text to display on the line.

Related Topics :
Technical Documentation
Topic Index Format
Keywords Index Format
Set File Format

Set File Format

   The set file is a text file that contains one entry in each line, using
 the following format :

help-database-engine
version-of-generating-compiler
topic-index-table name
links-index-table name
keywords-index-table name
help-raw-data-file name
number-of-topics
length-of-file
link-normal-color
link-high-color
text-normal-color

The help-database-engine values available today are DEF=Paradox and MEM=memory Tables.

Related Topics:
Technical Documentation
Topic Index Format
Links Index Format
Keywords Index Format

Keywords Index Format

The keywords index table format is as follows :

KEYWORD   - 40 characters string key - the keyword.
TOPIC   - 40 characters string key - the topic name that this keyword is
               assigned to.

Notice - the database will include a row for each keyword, that has the
TOPIC field blank. This is used to display the first list of keywords, that
if chosen, will display the 2nd list of topics that have the chosen keyword
in their .KEYWORDS definition.

Related Topics:
Technical Documentation
Topic Index Format
Links Index Format
Set File Format

Related Products

The following products/packages provide additional HLPDK functionality :

1. Name   : IL2HDK - Interrupt List (Ralf Brown) 2 HLPDK convert program.
   Version   : 2.2
   Availability : IL2HDK22.ZIP.
   Status   : Free.
   Author   : Ron Loewy.

2. Name   : HPCHDK - HelpPC 2 HLPDK convert program.
   Version   : 1.0
   Availability : HPCHDK10.ZIP.
   Status   : Free.
   Author   : Yaniv Golan (s2942275@tech02.technion.ac.il)

3. Name   : SCANHELP - Turbo Pascal Scanner to help.
   Version   : 3.01
   Availability : ? (Contact Author)
   Status   : Shareware.
   Author   : Duncan J. Murdoch (dmurdoch@mast.QueensU.CA)

4. Name   : NG2HDK - Norton Guides 2 HLPDK convert program.
   Version   : 1.0
   Availability : NG2HDK10.ZIP
   Status   : Free.
   Author   : Ron Loewy.

5. Name   : NEWSDB - News Clips Hypertext system.
   Version   : Beta.
   Availability : Not in GA yet. (I hope it will be around in Oct.).
   Status   : To Be Determined.
   Author   : Ron Loewy.

6. Name   : POPHDK - POPHELP 2 HLPDK convert program.
   Version   : 1.0
   Availability : POPHDK10.ZIP.
   Status   : Free.
   Author   : Yaniv Golan (s2942275@tech02.technion.ac.il)

7. Name   : JARG2HDK - Jargon File to HLPDK convert program.
   Version   : 1.0
   Availability : ? (Contact Author)
   Status   : ?
   Author   : Slren Pingel Dalsgaard (pingel@daimi.aau.dk)