Inf2Qh — Convert IBM OS/2 *.INF files to Microsoft Quickhelp files

Use of this program requires HelpMake.exe (OS/2 version 1.05 was used) and QH.EXE (version 1.8 was used).

[**QuickStart** – skip ahead to the section called “Running Inf2Qh” on p4]

1.1 What is QuickHelp?

Long time OS/2 programmers (from the 1.x days), like myself, may have become addicted to a program called QuickHelp (qh.exe). This little program is run as a keyboard monitor (“detach qh”) and is available from all full-screen OS/2 sessions when you hit ALT-Q.

For several years, all Microsoft operating system and language products used the QuickHelp format for their on-line help. Most of them shipped with HELPMAKE (the QuickHelp compiler). Some of the products that come to mind are:

MS OS/2 Toolkit 1.2
MS C 6.0a
MS PDS Basic
MS QuickC

Even OS/2 Warp 3.0 has at least one file in this format:

\os2\mdos\qbasic.hlp

Microsoft did a FANTASTIC job with the OS/2 API QuickHelp files. Unfortunately, they are for OS/2 1.x. They were so well done I still use them today, even when I am programming for OS/2 2.0.

1.2 About Inf2Qh

This program is a generic *.INF to *.QH converter. After the conversion is complete, you must compile the *.QH file into an *.HLP file with the HelpMake program.

Obviously, since QH is a text-mode program, things like fonts and bitmaps are not converted; but I think you'll be impressed with the level of sameness achieved with the original *.INF file.

As mentioned, this is a generic *.INF to *.QH converter —— if you don't like the way the information is arranged in the original *.INF file, this program is not going to help you —— with a few exceptions (see Formatting Exceptions, below)

1.3 Implementation Problems

The INF "Table of Contents", with its collapsible trees, was not reproducible. The way I handled this was to create list topics (the .list command in QH syntax) with names like "+Notices". When you select this topic, it clears the screen and shows the sub-topics for that topic. It was the best I could do. To get back up to the previous heading-level, press 'T' in QH.

If you specify the /t option (include Title), the commands to add your help file to the QH "Categories" menu will be created; note that you can only have a limited number of categories (about 23) in this menu, and I had no trouble at all using them all up. To alleviate this somewhat, you can use the "File | All Tables of Contents" command to get access to the files not in the Categories menu.

Many INF topic titles are too long for the QH '.list' command, which only looks at the first 50 characters on the line. The IOCTLS section in \toolkit\book\cp3.inf is a good example of this problem. The only way around this is to get to the topic using a hyper-link, or by selecting a topic on either side of it and using the view Next and Back functions to get to it.

Another problem with .lists's is that if they encounter a double space on the line, they treat it as a delimiter and search for a topic based on everything before the double space. For example, when you hit ENTER from a list on the line: "XCOPY – Copy Subdirectories", QH will try to find a topic named "XCOPY". Since there is no topic by this name, you'll get an error (see the /c command, in Formatting Exceptions, for a possible work around).

When compiling the converted file, you may receive a few messages like this:

```
     System Exceptions
  error H2001: duplicate context string
```

There was no way to avoid this. Explicit hyper-links to the duplicated topics will work as intended; but implicit searches will find which ever one is closer (use the 'E' (continue search) command, to cycle through the other topics with the same name).

Inf2Qh performs paragraph wrapping, so that the QH right margin (76) is honoured. However, wrapping is turned off while decoding an "example", and a few other INF formats that require that nothing be changed. For this reason, you may see a few "Line too long" errors from HELPMAKE.

IPF "footnotes" are converted into QH ".popup", unless they exceed 21 lines, in which case they are placed in a new topic.

Explicit references to external help files are not supported (for example, when CP1.INF has a hyperlink to something in CP2.INF, I do not create an explicit link — I simply change the text to BOLD).

The converter will never cause anything to be placed in the “References” menu.

1.4 Formatting Exceptions

There are some obvious *.INF files people are going to want to convert, that I have built in a few options for:

1.4.1 Option /c — Command Reference Mode.

This option was created for \OS2\BOOK\CMDREF.INF, but can apply to others.

In this mode, all topics with a “ –” are forced to have “ –” (two spaces), and an alias topic name is set up for everything preceeding the “ –”. This alleviates the problem, described above, for topics with names like “XCOPY – Copy Subdirectories”.

It also allows you to search for the commands just by their name (ex: COPY, BACKUP, etc), which is, of course, what you really want to do with the command reference, right?

1.4.2 Option /a — AutoCollect mode.

This option was created for any *.INF file that opens more than one window for a help topic.

When the IPF “auto dependent” type of windows are encountered (these are the “Auto Open” variety; for example, when you select DosOpen), the converter will not store the windows as separate topics. Instead, it puts them all into one topic, with sub-headings to separate them.

1.4.3 Option /p — Programmer’s Reference mode.

This option was created for the Warp programmer’s \toolkit\book\*.inf files (cp*.inf, gpi*.inf, pm*.inf, others).

This option automatically enables /a – Autocollect mode.

One of the common complaints I see from OS/2 programmers about VIEWing the online OS/2 API *.INF references is that a topic is split into too many panels. For example, looking up “DosOpen” opens up three windows — none of which are the really important ones. What people tell me is that want it all on ONE panel, so they can just scroll around themselves.

The /a (auto collect) option, explained above, will take you part of the way. The /p option goes one step further — if one of the “Auto Open” panels has “Topics –” in its title, the converter throws away that panel, but automatically sucks in all of the topics that were referenced by it. For example, without /p or /a, DosBeep would have EIGHT (!) panels:

1) DosBeep
2) Syntax – DosBeep
3) Parameter – freq
4) Parameter – dur
5) Return Value – ulrc
6) Related Functions – DosBeep
7) Example Code – DosBeep
8) Topics – DosBeep

With /p, this will be turned into one topic under the name DosBeep, except for panels they were originally defined by IPF as footnotes (the “Parameter –” panels) — they become popup windows.

Since this option removes so many topics (for example: “Return Value – ulrc”), it also eliminates many “duplicate topic” errors.

Note: The /p option causes a big organizational improvement for the OS/2 programmer, but it is still not as good as the MS OS/2 1.x QH references. Doing a job that good is not for this “generic converter”; the topics need to be parsed, diced, reformatted, rearranged, renamed, and have better .list support. I may release another program to do this in the future; but not any time soon.

1.5 Running Inf2Qh

If you run Inf2Qh.exe without any parameters, you will get the usage box:

```
ÛßßßßßßßßßßßßßßßßßßßßßßßßßßßßßßßßßßßßßßßßßßßßßßßßßßßßßßßßßßÛ
Û Inf2QH    (Convert *.INF to QuickHelp)          95/04/19 Û
Û                                             Version 1.99 Û
Û                                     by Peter Fitzsimmons Û
Û                                                          Û
Û Usage:    inf2qh <input.inf> [title] [options]           Û
Û Purpose:  Convert IPFC files to QuickHelp input files.   Û
Û Options: /v – verbose                                    Û
Û          /d – debug                                      Û
Û          /a – “AutoCollect Mode”            (see docs)   Û
Û          /p – “Programmer’s Reference Mode” (see docs)   Û
Û          /c – “OS/2 Command Reference Mode” (see docs)   Û
Û          /t – Include title for “Categories” menu        Û
Û                                                          Û
Û            Copyright (c) 1995,  A:WARE Inc.              Û
ÛÜÜÜÜÜÜÜÜÜÜÜÜÜÜÜÜÜÜÜÜÜÜÜÜÜÜÜÜÜÜÜÜÜÜÜÜÜÜÜÜÜÜÜÜÜÜÜÜÜÜÜÜÜÜÜÜÜÛ
```

The output file has the same name as the input file, but with the extension *.QH. It will be placed in the current directory.

[title] is optional. It overrides the title contained in the *.INF file. NOTE: Some *.INF files do not have a title; in this case, Inf2Qh will use the file name of the *.INF file as the title (if no title was specified when starting Inf2Qh). For more information on the [options], see “Formatting Exceptions”, earlier in this document.

1.5.1 Example:

```
Inf2Qh D:\TOOLKIT\BOOK\CP1.INF /p /t
Inf2Qh D:\TOOLKIT\BOOK\CP2.INF /p “CP Ref vol2”
Inf2Qh D:\TOOLKIT\BOOK\CP3.INF /p “CP Ref vol3”
Inf2Qh C:\OS2\BOOK\CMDREF.INF /c /t
```

(I have discovered that CP1.INF has a title built-in, and the other two do not)

This will create for files in the current dir: CP1.QH, CP2.QH, CP3.QH, CMDREF.QH. These files must now be compiled with the MS HelpMake program:

```
HELPMAKE /W200 /T /E /Ocp1.hlp  cp1.qh
HELPMAKE /W200 /T /E /Ocp2.hlp  cp2.qh
HELPMAKE /W200 /T /E /Ocp3.hlp  cp3.qh
HELPMAKE /W200 /T /E /Ocmdref.hlp  cmdref.qh
```

NOTE: This will take a while — 10mins perhaps. To speed it up, try /E0 instead of /E; this will turn off compression.

1.6 The $Price

I think this program has a small audience (albeit an appreciative one), so I'm not charging anything for using this program at this time.

I may, in the future (see next heading), use the same code base to create a product worth selling.

If you feel I have in some way significantly improved your life, I'd enjoy a thank-you note as remuneration (see the bottom of this file for a way to contact me).

1.7 The Future

I spent more than a week on this program (probably more like two) — and it turns out the easy part was decoding the *.INF format. The hard part was looking at the style that *.INF files were written, and how to get QuickHelp to do the same thing. This program is a lot more complicated then it probably appears. If I had known what I was getting into, rather than write a converter, I may have just written a native QH-like *.INF reader.

So there is not much of a future for this program. What I am interested in doing instead (these are not promises, just ideas) are:

1) An *.inf file printer. A program that paginates an *.INF file, adds a Table of Contents and Index (both with page numbers), and prints it out in PostScript format.

2) A native pop-up QH-Like *.inf viewer.

3) A specialized version of Inf2Qh for the Programmer's Reference files (if I do #2, I won't do this).

1.8 Acknowledgements

Carl Hauser – first person that I am aware of to hack at the *.INF format.
Colin Thomson – added to Carl's information.

1.9 Contact us for all your OS/2 development needs

Device Drivers, File Systems, Applications, we do them all. Contact:

Peter Fitzsimmons,
President,
A:WARE Inc.
6056 Cayeswood Court
Mississauga Ontario
Canada L5V-1B1.

Voice: 905 858 3222
Data : 905 858 8488 (BBS)

Fidonet: 1:259/414
Internet: sol3@olc.gvc.com

Contents