Using Single Source

When you use RoboHELP Classic with RoboHELP Office, you can generate other Help formats as well as WinHelp and WinHelp 2000. These formats include Microsoft HTML Help, WebHelp, and JavaHelp.

To generate a Help format from a RoboHELP Classic project:

1. From the RoboHELP Explorer, open the **Single Source** folder.
2. Double-click on the Help format you want and follow the instructions on your screen. The Single Source Wizard prompts you for information, then generates all the files necessary for the selected format.

Right-click options

From the RoboHELP Explorer **Single Source** folder, you can right-click on Microsoft HTML Help, WebHelp, and JavaHelp to access the following menu items:

- **Edit:** Use this option to edit a project after you've generated it. You can define the editor for each type of Help format from the RoboHELP Explorer **Tools** menu, **Options**, **Single Source** tab. If you haven't defined an editor, RoboHELP launches the application defined in your operating system based on the file extension. It is recommended that you make the changes in RoboHELP and regenerate the Help format rather than modify the generated source.
- **View:** Use this option to view a project after you've generated it. You can define the viewer for each type of Help format from the RoboHELP Explorer **Tools** menu, **Options**, **Single Source** tab. If you don't have a viewer defined, RoboHELP launches the application defined in your operating system based on the file extension.
- **Generate:** Use this option to start the Single Source Wizard, which generates the selected Help format. The wizard allows you to make selections that determine the folder, name, conversion options, and style/format options for the conversion.
- **Quick Generate:** If you have already set up your conversion options (using the Properties menu), you can select Quick Generate to create the Help format. Selecting Quick Generate skips the Single Source Wizard and generates the Help format immediately.
- **Properties:** Use this option to see or change the settings for the selected Help format.

RoboHELP Explorer icon

This icon means the menus and commands listed here are found in the RoboHELP Explorer.

ubrukt

Microsoft HTML Help overview

Microsoft® HTML Help is an online Help format based on Hypertext Markup Language format (HTML). It is another Help format, in addition to WinHelp, that runs on the Microsoft Windows platform. The Microsoft HTML Help format can be used on existing 32-bit Windows platforms and will also be the Help standard for Windows 2000 and future Microsoft operating systems..

Authors can use Microsoft HTML Help to develop online Help systems for software applications, books and training guides, and other online documentation.

HTML Help uses a combination of HTML, traditional WinHelp features, and unique Help features designed specifically for this Help format. Microsoft HTML Help enhances navigation with a table of contents, index, full-text search, related topics links, KLinks (keyword links), and ALinks (See Also cross-references).

This Help format uses the HTML Help compiler to compress all files in the project into a single file (.CHM) for distribution.

The main components of Microsoft HTML Help include:

HTML Help ActiveX® control: Supports navigation features such as table of contents, index, link controls (related topics, keywords, ALinks), and HTML Help controls (WinHelp topic links, splash screens, close window controls, etc.).

Layout engine: Microsoft Internet Explorer 4.0 or later supplies all the required components for HTML Help support.

HTML Help viewer: This window is designed specifically for displaying compiled HTML Help (.CHM files). It uses components of the Internet Explorer browser while displaying content in its own proprietary window interface.

Compressed HTML: The HTML Help compiler generates compressed HTML – a collection of all files in the project in a single .CHM file that occupies less disk space and is faster to load.

Key features of Microsoft HTML Help

Microsoft HTML Help has all the major Help features you need, including popup and drop-down text, full-text searches, context-sensitive Help, browse sequences, related topics, and graphical navigation and organization support (including multi-level indexes and tables of contents).

The basic features of a Microsoft HTML Help system include:

- HTML Help topics
- Hyperlinks and bookmarks
- Images
- Multimedia, Dynamic HTML, and special effects
- Table of contents (TOC)
- Index and keywords
- Style sheets
- HTML Help objects and ActiveX controls
- Scripts
- HTML Help window and Internet browser
- Full-text search
- Information types

You can create Microsoft HTML Help files incorporating these features using RoboHELP. For more detailed information about Microsoft HTML Help, go to the Microsoft HTML Help site at:

w **http://msdn.microsoft.com/workshop/author/htmlhelp/**

Tip:

If you’re looking for a cross-platform Help solution that is also browser-independent, consider WebHelp.

End-user requirements for Microsoft HTML Help

Make sure your end users have the following system requirements when you ship Microsoft HTML Help:

- 32-bit operating system (Windows 95, 98, NT 4.0, or 2000)
- 16 MB RAM or more (24 MB or more recommended)
- Internet Explorer 4 (for full HTML functionality)

Context-sensitive Help for Microsoft HTML Help

Context-sensitive Help provides information about what users see inside applications – details about fields and controls at dialogs, explanations of messages, and descriptions of windows and screen objects. It's called "context-sensitive" because each topic is called by the application and because it provides information relevant to the task the user is trying to accomplish. In Microsoft HTML Help, there are two kinds of context-sensitive Help:

Window-Level Help (also known as dialog Help, F1 Help, and Help button Help)

What's This? Help (also known as field-level Help)

Note: If you are creating a standalone project that is not part of an application, you do not need to develop context-sensitive Help.

Window-level Help in Microsoft HTML Help

Window-level Help provides an explanation about items at an open dialog or window. The displayed information explains how to work with the dialog or window that is active.

- To access window-level Help, users click a **Help button** or press **F1**. In some applications, a Help option is also available from the right-click menu.
- Help about using the window or dialog opens in a **window**.

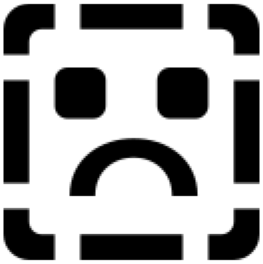
How is window-level Help created?

In RoboHELP HTML, you create window-level topics as regular HTML Help topics. These topics can include formatting, links, related topic buttons, images, and any kind of HTML-specific features that are suitable for your content. You do not include these topics in the table of contents or index, although you can link to them from topics such as step-by-step procedures.

For window-level Help topics to function as context-sensitive Help in Microsoft HTML Help, you create aliases and then compile the project. The way aliases are used in Microsoft HTML Help is very different from the way they’re used in WinHelp. In WinHelp, aliases are used to connect more than one map number to the same topic, or to connect an existing topic to a map ID. To find out more about aliases in Microsoft HTML Help, refer to the RoboHELP HTML online Help.

What's This? Help in Microsoft HTML Help

What's This? Help in Microsoft HTML Help provides help about specific fields and controls in a dialog box. Users click the question mark in the upper-right corner of a dialog and then click a particular field or control. Help about the specific component appears in a **text-only popup window**.



This kind of Help looks very much like tool tips that appear when users pause the mouse over toolbar buttons and other components. Examples of What's This? Help can be found throughout many Microsoft Windows applications.

Note: In some applications, developers program Help so users can right-click the field or control to view What's This? Help.

How is What's This? Help created?

In RoboHELP HTML, you create What's This? Help by creating text-only topics (context-sensitive topics in text-only format that are created and authored in a dialog in RoboHELP HTML).

In Microsoft HTML Help, What's This? Help is text only – formatted text, links, and graphics are not supported. The content that you author for these topics is intended to explain how to use each field or control at a particular dialog. Multiple topics are saved in text-only files (.TXT).

What's This? Help requires the creation of more topics than window-level Help, but the topic content is brief. Since these topics do not support formatting, it takes less time to create them. You also do not need to design a custom window, because the popup window that is used to display the information is automatically available in the system.

Target platforms comparison

Platforms are the type of computer or operating system you’re using, and the target platform is the specific platform you are creating Help for. The following information explains which platforms you can use when you’re using WinHelp and HTML Help.

WinHelp

- Windows 3.1, 95, 98, NT 4.0, or 2000

Microsoft HTML Help

- Windows 3.1, 95, 98, NT 4.0, or 2000

Distributing Microsoft HTML Help

Whether you are creating Microsoft HTML Help for applications or as standalone products, you need to distribute your Help system file(s) as well as files that provide support for Microsoft HTML Help functionality.

Your Help files include:

- **.CHM files.** If you are distributing a single Help system, you have one .CHM file to distribute. If your project includes secondary .CHM files that support merged indexes, external TOC files and links to external HTML Help systems, include all the secondary .CHM files as well. (The secondary .CHM files are not compiled into the master .CHM file.)
- **.HLP, .CNT files.** If your project includes links to WinHelp topics, distribute the .HLP and .CNT files with the HTML Help system. (These files are not compiled in the .CHM file.)

  If your Help system is used with a software program, provide your developer with copies of the above Help files. Your developer can install these files in the same folder where the program's executable (.EXE file) is located or in another folder (if preferred). If your Help system is used as a standalone product, you can install it in any folder that suits your requirements.

Tip:

- An easy way for users to run standalone Microsoft HTML Help is to double-click the .CHM file in the Windows Explorer or from a shortcut icon on the Windows desktop.

The support files for Microsoft HTML Help functionality include:

Platform	Microsoft HTML Help files	Support for browse sequences, glossaries, WebSearch
Windows 95	Internet Explorer 4.x or later HHUPD.EXE	HHActiveX.DLL
Windows NT 4	Internet Explorer 4.x or later HHUPD.EXE	HHActiveX.DLL
Windows 98	HHUPD.EXE	HHActiveX.DLL
Windows 2000	Built-in components	HHActiveX.DLL

All the above support files are supplied by Microsoft and redistributed by eHelp when you install your Help authoring product. (HHActiveX.DLL is provided by eHelp.) The licensing agreement you have with eHelp allows you to redistribute Microsoft HTML Help and Internet Explorer files as long as they are included with your HTML Help product.

Tips:

- If your Help system also includes third-party ActiveX controls, you need to install and register the file(s) that support these components. For more information, see Installing and registering ActiveX controls.
- HHUPD.EXE is a program that automates the process of installing a group of Microsoft HTML Help system files that are required for HTML Help functionality. The installation CD for your eHelp product includes this file. For information, see Installing Microsoft HTML Help system files.
- Internet Explorer (IE) is the browser that provides functionality for Microsoft HTML Help systems. The installation CD for your eHelp product includes IE 5. For more information, see Installing Internet Explorer.
- HHActiveX.DLL is an optional file that provides support for online glossaries (created in the Glossary Designer), browse sequences, and WebSearch. This file needs to be installed and registered on end-users' systems (it is not included with HHUPD.EXE). For more information, see Installing HHActiveX.DLL.

Installing HHActiveX.DLL

If your Microsoft HTML Help project includes an online glossary, browse sequences or WebSearch, a special file needs to be installed and registered on end-users' systems to support these features. The name of this file is HHActiveX.DLL. On your authoring system, it is located in the program folder for your eHelp authoring product.

- **Application Help.** Provide your developer with a copy of HHActiveX.DLL when you deliver your .CHM files. For information about how to install and register .DLLs, your developer can check the documentation included with the application's install program.
- **Standalone Help.** Copy HHActiveX.DLL on the disk or CD where your .CHM files are located. This file can be manually installed and registered.

To manually install HHActiveX.DLL:

1 Copy **HHActiveX.DLL** to Windows\System.

2 Click the Windows **Start** button.

3 Select **Run**.

4 In **Open**, enter the following:

```
regsvr32 hhactivex.dll
```

5 Click **OK**.

Tips:

- You only need to distribute HHActiveX.DLL if you are distributing a Microsoft HTML Help system that includes an online glossary, browse sequences or WebSearch.
- For information about the files to ship with your HTML Help product, see Distributing Microsoft HTML Help and Installing Microsoft HTML Help system files.
- If your project includes custom or third-party ActiveX controls, you need to install and register the ActiveX control.

Installing Microsoft HTML Help system files

To provide support for Microsoft HTML Help functionality on end-users' systems, Microsoft HTML system files need to be installed and registered on their computers. Your eHelp product installation disk includes the file HHUPD.EXE, which provides a shortcut for installing and registering all the required Microsoft HTML Help system files. HHUPD.EXE is located in a folder titled "Redist" on the install CD.

- **Application Help.** Provide your developer with a copy of HHUPD.EXE when you deliver your .CHM files. Your developer will run HHUPD.EXE to install the Microsoft HTML Help system files. For information about how to run external applications (HHUPD.EXE), your developer can check the documentation included with the install program.
- **Standalone Help.** Copy this file on the disk or CD where your .CHM files are located. Users can manually install the files by running HHUPD.EXE from the disk or CD.

Note: Internet Explorer 4.x or later should be installed before installing HHUPD.EXE. IE 5 is available on the install CD for your eHelp authoring product. Your licensing agreement with eHelp enables you to install and distribute IE 5 with your HTML Help products. For more information, see Installing Internet Explorer.

To prepare HHUPD.EXE for distribution:

1. Insert the installation CD for your eHelp authoring product.
2. By default, when you insert the CD, InstallShield will prepare your system for product installation.
3. Exit from setup.
4. From the Windows Explorer, navigate to the drive where your CD ROM drive is located (for example, e:/).
5. Open the folder titled **Redist**.
6. Right-click the file HHUPD.EXE and select **Copy**.
7. Save a copy of this file on your system. For standalone Help, save a copy of this file on your disk or CD. For application Help, provide your developer with a copy.

To manually install HHUPD.EXE:

1. On the end-user's system, insert the disk or CD that contains the .CHM files and HHUPD.EXE.
2. Double-click HHUPD.EXE. It automatically installs and registers the required system files on the end-user's system.
3. Copy the .CHM file(s) on the end-user's system.

Tips:

- If you are distributing a Microsoft HTML Help project that includes an online glossary, browse sequences, or WebSearch, you will also need to install and register a special file that supports these features (it is not included with HHUPD.EXE). For more information, see Installing HHActiveX.DLL.
- For more information about files to ship with Microsoft HTML Help, see Distributing Microsoft HTML Help.

Installing and registering ActiveX controls

If your Microsoft HTML Help project includes custom or third-party ActiveX controls, the ActiveX files (.OCX, .DLL) need to be installed and registered on end-users' systems.

- **Application Help.** Provide your developer with a copy of HHActiveX.DLL when you deliver your .CHM files. For information about how to install and register .DLL and .OCX files, your developer can check the documentation included with the application's install program.
- **Standalone Help.** Copy the .DLL or .OCX file on the disk or CD where your .CHM files are located. This file can be manually installed and registered.

To manually install an ActiveX control:

1. Copy the **.OCX** (or **.DLL**) file to Windows\System.
2. Click the Windows **Start** button.
3. Select **Run**.
4. In **Open**, enter the following:

   ```
   regsvr32 activex_name.ocx
   ```

   where activex_name is the name of the .OCX or .DLL file
5. Click **OK**.

Tips:

- For information about the files to distribute with your HTML Help product, see Distributing Microsoft HTML Help and Installing Microsoft HTML Help system files.
- You can copy the ActiveX control files into a custom folder rather than into Windows\System. If you do this, include the path for the alternate folder in front of the *.OCX or *.DLL file name so Windows can locate the file.
- If your project includes an online glossary, browse sequences, or WebSearch, you need to distribute HHActiveX.DLL with your Help product. For more information, see Installing HHActiveX.DLL.

Advantages of Internet Explorer 4.0 or later

Although you can use Internet Explorer 3.02 with Microsoft HTML Help, Internet Explorer 4.x and 5.0 provide several advantages, including full or enhanced support for the following:

- Popup hyperlinks
- Text-only popups
- Dynamic HTML
- Avenues
- Topic keywords
- ALink controls
- Online glossaries
- Drop-down hotspots and expanding hotspots
- Style sheets
- Multimedia within a compiled HTML Help file (.CHM)
- HTML Help ActiveX control and other ActiveX controls
- Full-text search features
- Favorites tab

Note: When you install RoboHELP, you have the option of also installing Internet Explorer 5.

Installing Internet Explorer

To view and run Microsoft HTML Help, end-users need Internet Explorer 4.x or later. This browser is built into the Windows 98 and 2000 operating systems. For Windows 95 and NT 4, it needs to be installed.

Internet Explorer 5 (IE) is available on the installation CD of your eHelp authoring product. Your licensing agreement with eHelp enables you to install and distribute IE 5 with your HTML Help products.

To prepare IE 5 for distribution and installation:

1. Insert the installation CD for your eHelp authoring product.
2. By default, when you insert the CD, Install Shield will prepare your system for product installation.
3. Exit from the setup program.
4. From the Windows Explorer, navigate to the drive where your CD ROM drive is located (for example, e:/).
5. Right-click the folder titled IE and select **Copy**.
6. Save a copy of this folder on your system. For standalone Help, save a copy on your disk or CD. For application Help, provide your developer with a copy.

To manually install IE:

1. On the end-user's system, insert the disk or CD that contains the .CHM files and the IE folder.
2. Open the IE folder.
3. Double-click the file ie5setup.exe.
4. Follow the on-screen instructions.

Notes:

- Your developer will run ie5setup.exe to install the Internet Explorer files as part of the main install. For information about how to run external applications, your developer can check the documentation included with the install program.
- IE 5 should be installed BEFORE installing Microsoft HTML Help system files. This is especially important for your developer to know. For more information, see Installing Microsoft HTML Help system files.

User interface comparison

User interfaces are portions of a program with which users interact, such as menus. The following information explains how they work in WinHelp and HTML Help.

WinHelp	Microsoft HTML Help
• Compiled WinHelp .HLP file (Contents .CNT file for WinHelp 4)	• Compiled HTML Help .CHM file contains all project files
• HLP file is launched by WINHELP.EXE or WINHELP32.EXE	• CHM file is launched by HH.EXE and the Html Help API
• Built-in WinHelp viewer provides display functionality	• HTML Help viewer provides display functionality (built-in for Windows 98, also installable via IE browser)
• Contents, Index and Find tabs displayed in separate window	• Contents, Index Search, Favorites and Glossary tabs displayed at left side of tri-pane window
• Help topics displayed in windows (main, secondary and popups)	• Topics displayed on right side of tri-pane window, in secondary windows, or in popup windows
• Authors can use non-scrolling regions to separate topic headings from content (headings do not scroll with content)	• Authors can use frames to create additional panes within windows

Source file comparison

Source files are important files associated with your Help project. The following information explains how they work in WinHelp and HTML Help.

WinHelp	**Microsoft HTML Help**
• Rich Text Format (RTF)	• Hypertext Markup Language format (HTML)
• WinHelp project file (HPJ)	• HTML Help project file (HHP)
• Compiled Help file (HLP)	• Compiled Help file (CHM)
• Table of contents (CNT)	• Table of contents file (HHC)
• No separate index file	• Index file (HHK)
• Help files (RTF)	• Help files (HTM)
• Multiple Help topics in a single .RTF file	• One Help topic in a single .HTM file
• Multiple sub-topics in a single Help topic (mid-topic IDs)	• Multiple sub-topics in a single HTM file (bookmarks)

Compilation comparison

As you build your Help project, you create Help content with the goal of compiling. Compiling means combining all the source files and components in the Help project and processing them into one compiled Help file. The following information explains how they work in WinHelp and HTML Help.

WinHelp

- WinHelp compiler: HCRTF.EXE
- Compiles project (.HPJ), topics (.RTF), images (.BMP, .WMF, .MRB, .SHG), multimedia (.WAV, .MOV), and other source files into a single file (.HLP) for distribution. WinHelp 4 projects require a separate contents file (.CNT) as part of the deliverables

Microsoft HTML Help

- HTML Help compiler: HHA.DLL
- Compiles project (.HHP), contents (.HHC), index (.HHK), topics (.HTM), images (.GIF, .JPEG), multimedia (.WAV, .MOV, etc.) and other source files into a single file (.CHM) for distribution

Context-sensitive Help comparison

Context-sensitive Help provides information about what users see inside applications – details about fields and controls at dialogs, explanations of messages, and descriptions of windows and screen objects. The following information explains how it works in WinHelp and HTML Help.

WinHelp	**Microsoft HTML Help**
• Field-level/What's This? Help (text formatting)	• Field-level/What's This? Help (text-only)
• Dialog/window-level Help (text formatting)	• Dialog/window-level Help (text-formatting)
• Map files	• Map files
• Aliases (connect topic IDs to Help topics)	• Aliases (match topic IDs to window-level Help topics)
• WinHelp API is used by developers to provide context-sensitive functionality with applications	• HTML Help API is used by developers to provide context-sensitive functionality with applications

Keywords and index comparison

Online indexes are similar to those you’d find in the back of a book – alphabetical listings of terms designed to quickly direct users to topics that best answer their questions. A WinHelp index consists of a list of keywords and phrases. The following information explains how they work in WinHelp and HTML Help.

WinHelp

- Topic keywords saved in footnotes
- No separate index file
- Two-level or single-level keywords
- Merged indexes from other WinHelp systems
- K-Link macro (based on K-Keywords)

Microsoft HTML Help

- Topic keywords saved in HTML files
- Index keywords saved in .HHK file
- Multi-level or single-level keywords
- Merged indexes from other HTML Help systems
- Keyword link controls topics (created from index file keywords and topic keywords)
- Keywords that jump to other keywords in the index (cross-references)
- Keywords that open email messages, Web sites, FTP sites, newsgroups, multimedia files, external HTML topics

Table of contents comparison

Help system tables of contents (TOC) provide users with a hierarchical overview of the content. Users can browse through a TOC to see what topics are covered and select a topic to get the information they want. The following information explains how they work in WinHelp and HTML Help.

WinHelp	Microsoft HTML Help
• Hierarchy of books and pages (arranged by authors) displayed in separate Help topics dialog	• Hierarchy of books and pages (arranged by authors) displayed in left-hand pane of HTML Help viewer
• Separate .CNT file (for WinHelp 4) shipped with .HLP file	• TOC items saved in .HHC file in HTML Help project, compiled into HTML Help .CHM file
• Merged tables of contents from other WinHelp systems (Master CNT file)	• Merged tables of contents from other HTML Help systems (external TOC files)
• Pages can link to WinHelp topics, run macros, open external WinHelp topics and go to Web sites, intranets and display HTML pages	• Books and pages can link to HTML Help topics, Web sites, intranets, email addresses, FTP sites, newsgroups, play sound and video, and open external HTML topics (in other .CHM files)
• Pages can open target topics in main window and secondary window	• Books and pages can open target topics in main window, secondary windows and frames
	• Availability of TOC items can be specified by using information types
	• Customizable icons for books and pages

Hypertext links comparison

Hypertext links (also called hotspots) are electronic links that allow users to move from topic to topic in a Help system. The following information explains how they work in WinHelp and HTML Help.

WinHelp	**Microsoft HTML Help**
• WinHelp jumps link text and images to topics	• HTML hyperlinks link text, images and multimedia to topics
• Jumps can be in the form of text, images and authorable buttons	• Hyperlinks can be in the form of text, images, and video clips
• Destination topics can be displayed in main, secondary and popup windows	• Destination topics can be displayed in HTML Help viewer, popup windows and frames
• Destinations include WinHelp topics (in .HLP file), external WinHelp topics (in other .HLP files), HTML topics (in .CHM files), Web and intranet sites, and HTML pages	• Destinations include HTML topics (in .CHM file), remote topics (in external .CHM files), Web and intranet sites, email addresses, FTP sites, newsgroups and multimedia (sound and video clips)
• Supports jumps to subtopics in the same WinHelp topic (mid-topic jumps)	• Supports links to subtopics in the same HTML topic (bookmarks)
• Macro hotspots run special scripts that accomplish WinHelp-specific links such as shortcuts that run applications	• Link controls link to related topics, ALinks and keywords, HTML Help controls link to WinHelp topics, and run applications via shortcuts
• Images with multiple hotspots (SHED clickable images)	• Images with multiple hotspots (image maps)

Windows comparison

Windows are the "containers" that display topics in the compiled Help system. The following information explains how they work in WinHelp and HTML Help.

WinHelp	**Microsoft HTML Help**
WinHelp projects can include main, secondary and popup windows	HTML Help projects can include main, secondary and popup windows (both links to HTML topics or text-only topics)
Main window is WinHelp viewer (built into Windows operating system)	Main window is HTML Help viewer (part of Windows 98 operating system, installable via Internet Explorer browser with non-Windows 98 systems)
Main and secondary windows can be customized to include title bar (caption), size and position settings, background color (scrolling and non-scrolling regions), and extra navigation buttons	Main and secondary windows can be customized for title bar (caption), size and position settings, tri-pane design, Contents, Index, Favorites, Search, and Glossary tabs, and extra navigation buttons including avenues
Features such as print, copy, annotate, bookmark definitions, font size and colors are automatically included in main window	Features such as print, hide, show, refresh, search highlight on/off are automatically included in HTML Help viewer
	Other display features such as fonts, colors, and language are based on default browser settings (Internet Explorer)
WinHelp popups are created by specifying an action type or by creating a new topic and entering text directly into text box Popup window sizes to fit content and closes with a single mouse click WinHelp popups display WinHelp topics and support RTF formatting, images, jumps, and all other topic content Popup windows use the same color as the window from which they are launched (default colors can be specified)	Popup links to HTML topics are based on JavaScript and Dynamic HTML HTML topics are displayed in popup windows that size to fit content and close with a single mouse click Hyperlink popups support formatting, images, links, and all other HTML topic content
	Text popups (based on JavaScript) are created by entering text directly into popup window Popup window sizes to fit content and closes with a single mouse click Supported formatting includes text color, size, font, background color, vertical and horizontal margins

Related topics comparison

Related topics are navigational links that work somewhat like the "See Also" references – they cross-reference other topics containing related subjects. Help authors provide See Also references so users can easily locate more information within this same subject or topic grouping. The following information explains how they work in WinHelp and HTML Help.

WinHelp	**Microsoft HTML Help**
• K-Keywords are assigned to topics	• Topics are created in project and are available for using as related topic links
• Related Topic hotspots are created with KLink macro	• Related topic link controls (based on ActiveX)
• K-Keywords are linked to selected topics	• Created as an HTML Help link control
• Users click Related Topics hotspots to see a list of topics	• Selected topics are assigned to the control, based on topic titles
	• Users click Related Topics buttons (authorable button text or custom image) to see a list of related topics

See Also comparison

See Also references are navigational links in Help systems that work somewhat like the "See Also" references in books – they cross-reference other topics containing related topics. The following information explains how they work in WinHelp and HTML Help.

WinHelp	**Microsoft HTML Help**
• A-Keywords are assigned to topics	• ALink names are assigned to topics
• See Also hotspots are created with ALink macro	• ALink controls are created (based on ActiveX)
• See Also keywords (A-Keywords) are linked to selected topics	• Topics are assigned to control, based on their ALink names
• Users click See Also hotspots to see a list of topics to go to	• Users click ALink button (authorable button text or custom image) to see a list of topics to go to

Browse sequences comparison

Browse sequences allow users to move forward and backward through a series of topics in an order defined by the Help author. The following information explains how they work in WinHelp and HTML Help.

WinHelp	**Microsoft HTML Help**
• Browse sequences are created in Browse Sequence Editor	• Browse sequences are created in Browse Sequence Editor
• Topics are added to browse sequences in the editing tool	• Browse sequences are assigned to topic properties and written to a project file (.BRS)
• Users navigate through browse sequences with special navigation buttons that are included in the main Help window	• Users navigate through browse sequences with special navigation buttons that are added to the HTML Help viewer

Build tags comparison

Build tags allow you to design and create several different Help files from a single set of source files. The following information explains how they work in WinHelp and HTML Help.

WinHelp	Microsoft HTML Help
• Build tags can be assigned to specific topics or created for an entire document (tag is assigned to all topics in a given .RTF file)	• Build tags can be assigned to specific topics
• Build expressions are specified for the project, and the WinHelp compiler determines topics to include in compiled .HLP file (blank expression includes all topics)	• Build expressions are specified for project and HTML Help compiler determines topics to include in compiled .CHM file (blank expression includes all topics)

Topic formatting comparison

You can change the formatting of topics, including the styles, text colors, and tables. The following information explains how formatting works in WinHelp and HTML Help.

WinHelp	Microsoft HTML Help
• Microsoft Word templates (.DOT)	• Cascading Style Sheets (.CSS)
• Individual blocks of text (characters and paragraphs) can be manually formatted	• Individual blocks of text can be manually formatted (inline styles)
• In Microsoft Word, styles can be created for individual documents and applied to characters and paragraphs	• Embedded styles can be created in single HTML files
• Document styles are available for all WinHelp topics in the same file	• Embedded styles can be applied to any text within the single HTML file
	• Embedded styles are only available for a single HTML Help topic (not available with other topics in the project)
• Word templates can be created and linked to one or multiple documents	• External style sheets can be created and linked to one or multiple HTML files
• All documents that are linked to template can use the same styles	• All topics that are linked to style sheet can use the same styles
• When a style is changed in the template, all topics that are linked to the template are automatically updated	• When a style is changed in the style sheet, all topics that are linked to the style sheet are automatically updated
• When a style is added to template, it is available to all topics that are linked to the template	• When a style is added to style sheet, it is available to all topics that are linked to style sheet

Tables comparison

While tables are easy to create in Microsoft Word, there are some limitations to creating tables in WinHelp. The following information explains how they work in WinHelp and HTML Help.

WinHelp	**Microsoft HTML Help**
• Tables are supported, but formatting is limited	• Tables are supported with HTML formatting
• Width and heights are supported	• Width and height, borders, shading are supported
• No borders, no shading	

Images comparison

RoboHELP supports a variety of image file formats. The following information explains how they work in WinHelp and HTML Help.

WinHelp	**Microsoft HTML Help**
• Supported formats include bitmaps (.BMP), multi-resolution bitmaps (.MRB), and Windows metafiles (.WMF)	• Supported formats include graphics interchange format (.GIF), joint photograph expert group (.JPG, .JPEG)
	• Splash screens and icons support bitmaps (.BMP), multi-resolution bitmaps (.ICO) and Windows metafiles
• SHED graphics (.SHG) provide support for clickable hotspot graphics	• Both .GIF and .JPEG formats support image maps
• Transparent graphics (.BMP format)	• Transparent images (.GIF format)
• Not available	• Animation (.GIF format)

Multimedia comparison

Help multimedia is using more one medium in a Help system – audio, video, or animation. The following information explains how it works in WinHelp and HTML Help.

WinHelp	Microsoft HTML Help
• Supported formats include .WAV and .AVI as "baggage" with special supported software	• Supported formats include .AVI, .MOV, .AU, .MID, .RMI, and .WAV (any video animation and audio formats used by ActiveX, JavaScript, VBScript and those supported by standard HTML)
• Multimedia functionality is provided by macros	• Multimedia functionality is provided with HTML tags

Internet/intranet access comparison

Accessing the Internet or an intranet from within a Help system is convenient for end users. The following information explains how they work in WinHelp and HTML Help.

WinHelp	**Microsoft HTML Help**
• Supported via macros	• Automatically provided by HTML code (HREF tag)

Extensions comparison

Extensions are programs or program modules that add functionality to or extend the effectiveness of RoboHELP. The following information explains how they work in WinHelp and HTML Help.

WinHelp	Microsoft HTML Help
• WinHelp macros and DLLs	• HTML Help controls, link controls, ActiveX controls, scripting

Miscellaneous comparisons

The following information explains how several miscellaneous features work in WinHelp and HTML Help.

WinHelp	Microsoft HTML Help
• Non-scrolling regions	• Non-scrolling regions can be emulated via frames
• Macros (startup macros, topic entry macros)	• Scripts (more powerful than macros)
• Word field codes (automatic numbering for steps)	• Field codes are not supported, HTML numbered lists for steps
• Not available	• Dynamic HTML for special effects (such as expanding hotspots, drop-down hotspots, fade-in text and objects)
• Not available	• Splash screens (image displays when topic loads, BMP files)
• Not available	• Information types

HTML Help topics

HTML topics are the building blocks of an HTML Help project. Each topic is an individual file in HTML format that can be associated with information types. HTML Help topics contain the content of your Help file and can be comprised of text, images, hyperlinks, bookmarks, multimedia, HTML Help objects, ActiveX controls, and more.

Hyperlinks and bookmarks

Hyperlinks provide a way for users to navigate throughout your HTML Help system. In addition to other topics, hyperlinks can point to Web sites, URLs, e-mail addresses, FTP sites, and newsgroups. Hyperlinks can be in the form of text or images.

Bookmarks are like anchors that you can link hyperlink to. They provide a way for users to navigate within an HTML file or topic. When the reader clicks a bookmark, HTML launches them to the bookmark's actual location in the topic.

Images

HTML Help supports .JPG, .JPEG, and .GIF image formats, which are used in HTML pages because they are the fastest to load into browsers. GIFs support transparent images – images for which all the white pixels in the image are replaced with the background color of the currently active window. GIFs also support animation created by displaying several image layers in sequence inside a single file. GIF images provide the best quality for grayscale photographs, screen shots, cartoons, small icons, buttons, bars, lines, and bullets. They can also be interlaced, where the HTML page slowly builds an image of the GIF in gradual steps.

JPEG images are compressed high-color or true-color images. If you are using images with 256 colors or more (such as pictures), you will want your images to be in compressed JPEG format, which is handled very well by HTML browsers and the HTML Help window.

Use the WYSIWYG editor to add images to your Help topics, and to display different background colors and images.

Multimedia, Dynamic HTML, and special effects

You can make your HTML Help system come to life by adding movies, sounds, animated .GIFs, marquees, and Dynamic HTML. Microsoft HTML supports .AVI, .MOV, .AU, .MID, .RMI, and .WAV formats for creating sounds and movies. Animated .GIFs contain a sequence of several images in a single file that, when viewed in the Help window or browser, create the appearance of animation. Marquees are scrolling text messages that act like dynamic banners. Dynamic HTML adds exciting animation and visual effects to text and objects in your topics. For example, you can use it to have text slowly fade into view or slide into the window from any direction.

Table of contents (TOC)

The table of contents in an HTML Help system works just like the Contents tab in a Windows 95 WinHelp system, which means that it can display topics in an expandable and collapsible hierarchy.   You create the TOC by inserting and naming books, and by dragging and dropping topics from the Topic List into the TOC Composer. Books and pages can have information types associated with them so users have a way to filter information in the HTML Help window or browser.

Index and keywords

The index in an HTML Help system works like the Index tab in a Windows 95 WinHelp system. Index entries are based on keywords created from topic titles and special keyword phrases that you enter into the project. You can drag and drop topics into the Index Designer and create a list of keywords and associate topics with them. The resulting index is hierarchical – like the kind you find at the back of a book – and can include multiple levels.

Style sheets

Similar to templates used in Microsoft Word and other programs, a style sheet is a template that controls the formatting of topics. You can alter the appearance of a topic by changing the formatting assigned to it from a style sheet. Style sheets provide a way to define a full array of layout specifications for HTML topics such as font style and size, line spacing, paragraph indents, and more.

HTML Help objects and ActiveX controls

HTML Help objects are special navigation controls for HTML Help such as Splash screens, Related Topics, and Shortcut buttons. ActiveX controls are reusable HTML objects that add special functions to your topics such as calendar controls, charts, and banners.

Scripts

Scripts are another type of HTML object that add special functions to your topics. Scripts are to HTML what macros are to WinHelp, although scripts are much more powerful. A script tells the Help window or browser what to do after a user does something with an HTML object (such as clicking a button). HTML Help supports scripts that are created in VBScript and JavaScript.

HTML Help window and Internet browser

In HTML Help, the familiar WinHelp window has been transformed into an HTML Help window - a combination of the standard Help window with the added functionality of an Internet browser. It is designed as a tri-pane window with table of contents and index tabs at the left, topic content at the right, and navigation controls across the top, although you can design it to suit your needs and add your own custom windows. You can also display HTML Help systems in Microsoft's Internet Explorer browser (version 4.x is recommended for full functionality).

Full-text search

HTML Help supports full-text searches. You can set the HTML Help window to display a Search tab along with the table of contents and index tabs on the left side. The Search tab only lists words that appear within Help topics.

Information types

Information types determine if a topic appears in an HTML Help table of contents. In HTML Help, you define information types when you define the properties for books and pages. This allows you to customize the number of levels that appear in your table of contents and to exclude certain topics from the TOC.

Alias

In Microsoft HTML Help, the relationship between a window-level topic, a topic ID, and its corresponding map number.

Microsoft HTML Help system files

Microsoft HTML Help system files make it possible for your compiled Microsoft HTML Help files (.CHMs) to work properly. These files are automatically installed and registered on your computer when you install your eHelp authoring product. End-users also require these files, but they are not included in the compiled .CHM file that you copy on their systems.

Your eHelp product installation CD includes a file that automatically installs these files on end-users' systems. The name of this file is HHUPD.EXE. It is located in the **Redist** folder on the CD.

When you run HHUPD.EXE, the following files are installed and registered on end-users' systems:

File:	**Description:**
HH.EXE	Displays .CHM files in the HTML Help viewer.
HHCTRL.OCX	HTML Help ActiveX control that provides functionality for displaying topics, the table of contents, index, full-text search and all other HTML Help components.
ITIRCL.DLL	Supports compressed .CHM files.
ITSS.DLL	Supports full-text search.

Tips:

- See Installing Microsoft HTML Help system files for information about installing Microsoft HTML Help system files.
- If your Microsoft HTML Help system includes an online glossary, browse sequences or WebSearch, you need to install and register an ActiveX control that provides functionality for these components (it is not included with HHUPD.EXE). For details, see Installing HHActiveX.DLL.
- If your project includes custom or third-party ActiveX controls, you need to install and register the ActiveX control. For details, see Installing and registering ActiveX controls.

WinHelp overview

WinHelp stands for Windows Help and is a Help format used for Help systems that run on a Microsoft Windows operating system.

The WinHelp format uses a WinHelp compiler to process the final Help file and views that Help file using the WinHelp engine viewer. The WinHelp engine viewer is distributed with every copy of Microsoft Windows and installed in the \WINDOWS directory.

There are two types of WinHelp – WinHelp 3 and WinHelp 4. The main difference between them is the Windows operating system they are targeted to run on:

- **WinHelp 3.** WinHelp 3 is the original Windows Help. It was primarily developed for Windows 3.x and Windows 3 applications, which use 16-bit technology. WinHelp 3 features and functionality are fully supported on all Windows operating systems – Windows 95 and newer.
- **WinHelp 4.** WinHelp 4 was introduced with Windows 95. It uses an expanded 32-bit technology and includes newer features and added functionality not available through WinHelp 3. WinHelp 4 Help systems only run on Windows 95, 98, NT 4.0, and 2000 operating systems. WinHelp 4 expands the features in WinHelp 3 and offers several additional features – such as the Help Topics browser user interface (containing the Contents, Index, and Search tabs), a hierarchical table of contents that uses the books and pages metaphor, See Also links to easily display related topics, and more.

Creating WinHelp 3 and WinHelp 4 from the same source

With RoboHELP Classic, you can easily create Help that runs on multiple Windows platforms – Windows 3.x, 95, 98, NT 4.0, or 2000 – from the same Help project source files.

Keep in mind that there are differences between the Help files created for Windows 3.x, and those created for Windows 95, 98, NT 4.0, and 2000. The WinHelp 3 compiler used to create the Windows 3.x file doesn't support many features and functions of the WinHelp 4 compiler. For more information about the differences, see Comparing WinHelp 3 to WinHelp 4.

To create WinHelp 3 and WinHelp 4 Help from the same Help project:

1 Create your Help project.

2 From the **File** menu, select **Generate**, then select either **WinHelp 3** (Win 3.1) or **WinHelp 4** (Windows 95, 98 2000 and NT). The **Results dialog** displays the results of the compile on screen.

- To look at your resulting Help file, click **Run**.
- If there are errors, click **Error Wizard** to review and locate errors.

Tip:

You can use the Compatibility Wizard to give your Windows 3 Help system the look and feel of Windows 95/NT Help systems. You can add a three-tab window that includes a fully functional Contents tab, an Index tab, and a Find tab. After compiling, you can access the three-tab window by clicking a Help Topics button that the Compatibility Wizard inserts on the WinHelp 3 button bar. To have a fully functional Contents tab, it's important to create a table of contents in the TOC tab of RoboHELP Explorer.

Comparing WinHelp 3 to WinHelp 4

WinHelp 3 and WinHelp 4 are different WinHelp formats created with different compilers that operate in different Windows environments.

- **WinHelp 3.** WinHelp 3 supports the 16-bit Windows 3.x environment and all Windows-based Help features and functions supported by 16-bit systems. WinHelp 3 is compiled using one of two WinHelp 3 compilers, which allow you to create Help for Windows 3.x. (Windows 3.x Help systems can be run on Windows 95, 98, NT 4.0, and 2000 operating systems.)
- **WinHelp 4.** WinHelp 4 supports the Windows 95, 98, NT 4.0, and 2000 environments and all Windows-based Help features and functions supported by Windows 3.x (16-bit systems) and Windows 95, 98, NT 4.0, and 2000 (32-bit systems). WinHelp 4 is compiled using the WinHelp 4 compiler (HCRTF.EXE), which allows you to create Help for Windows 95, 98, NT 4.0, and 2000 operating systems.

Here's more on the differences between the 16-bit nature of Windows 3.x and the 32-bit nature of Windows 95, 98, NT 4.0, and 2000:

- If you are generating a WinHelp 3 Help file and the compiler is unable to continue, you will receive an error message. For example, long file names and directory names are not supported by the WinHelp 3 compiler and will cause the compiler to stop.
- If you added features not supported by WinHelp 3 and changed the Primary Target to WinHelp 3 or Only allow WinHelp 3 supported features, consider removing those features before compiling your Help system for WinHelp 3.
- If you selected the WinHelp 3 Compatibility option on the Primary Target even though you will be using the WinHelp 4 compiler, RoboHELP makes any unsupported WinHelp 4 compiler features unavailable to you during Help development. In this way, RoboHELP tracks the differences between the two compilers for you so you don't have to.
- If you generate your WinHelp 3 Help file using a Windows 95 or greater operating system, the Help Topics button does not display in the compiled project. You **will** be able to see the Help Topics button using Windows 3.1, Windows NT 4.0, or Windows 2000.
- To add the Compatibility option to a WinHelp 3 project, see Using the Compatibility Wizard.
- See WinHelp 4 features not supported in WinHelp 3.

WinHelp 4 features not supported in WinHelp 3

The following WinHelp 4 features are not supported in WinHelp 3:

- What's This? Help Composer topics
- Help window Print button, eHelp button, Smart See Also button, Contents tab, Index tab, and Search tab
- Help window Font option and Use System Colors option
- Secondary Help window button bar
- Second-level keywords for Indexing
- A-Keywords for linking topics
- Authorable buttons and mini-buttons
- Transparent images
- Multimedia files (If you have RoboHELP Office, you can create video files for WinHelp 3 by using the Software Video Camera or insert sound and video into WinHelp 3 using RoboHELP Office's enhanced RHMMPLAY.DLL.)
- Long file names and folder names (including spaces or other characters)
- Index Macro right-click menu option
- See Also option
- Watermarks for topics
- Compression settings (variable)
- ActiveEdit (for Help system development)
- Compile notes
- Temp directory option in Project Settings, Advanced tab
- Master table of contents
- WinHelp 2000 Explorer View

Strategies for sharing source files

Trying to keep track of all the differences between the two compilers can be difficult – especially if you need to create a WinHelp 3 system as well as a WinHelp 4 system. So how do you do it?

- **Allow RoboHELP to track the differences for you.** When you select a primary target for your Help system, you choose the primary compiler – WinHelp 3 or WinHelp 4. By choosing **WinHelp 4** with the **Only allow WinHelp 3 supported features** option, RoboHELP provides you with a warning that the feature isn't supported by WinHelp 3, so you have the option of including it or not.
- **Use Single Source.** RoboHELP Classic's Single Source allows you to generate WinHelp 3 and WinHelp 4 Help systems from the same source files with just one click.

Generating WinHelp 4 Help

You can use RoboHELP Classic's Single Source to generate Help for WinHelp 4 (Windows 95, 98, NT 4.0, and 2000), even if the Help file is primarily for WinHelp 3 (Windows 3.x). WinHelp 4 Help systems run on Windows 95, 98, NT 4.0, and 2000 operating systems.

To generate WinHelp 4 Help:

1 From the **File menu**, select **Generate**, then select **WinHelp 4** (Windows 95, 98, NT 4.0, and 2000).

2 RoboHELP Explorer indicates that generating RTF files for WinHelp 4 make take some time. Click **Yes** to continue. RoboHELP displays the WinHelp 4 results.

   - If there are errors, you can click **Error Wizard** to review and locate the errors.
   - If you want to look at the resulting Help file, click **Run**.

Tips:

- If the primary target is already set to WinHelp 4, you can use Quick Generate to quickly generate your Help file. Open the **Single Source** folder, right-click on **WinHelp 4 (Windows 95, 98, NT 4.0, and 2000),** and select **Quick Generate**.
- If you want to view the WinHelp 4 Help file after the conversion, open the **Single Source** folder, right-click on **WinHelp 4 (Win 95 & NT),** and select **View**.

Generating WinHelp 3 Help

You can use RoboHELP Classic's Single Source to generate Help for WinHelp 3 (Windows 3.x), even if your Help file is primarily for a WinHelp 4 target (Windows 95, 98, NT 4.0, or 2000). WinHelp 3 Help systems run on Windows 3.1, Windows 95, Windows NT 4.0, and Windows 2000 operating systems.

Be aware that when you generate WinHelp 3, you are using the WinHelp 3 engine and compilers. That means any WinHelp 4 capabilities in your current Help project will not appear in the resulting WinHelp 3 Help file (HLP). If the primary target for your Help project is set as WinHelp 4 with WinHelp 3 Compatibility, RoboHELP automatically "turned off" any incompatible compiler and viewer options while you developed the Help file. That means all features in your current Help project will translate into the WinHelp 3 HLP file.

To generate WinHelp 3 Help:

1 From the **File** menu, select **Generate**, then select **WinHelp 3 (Win 3.1)**.

2 RoboHELP Explorer indicates that generating RTF files for WinHelp 3 make take some time. Click Yes to continue. RoboHELP displays the WinHelp 3 results.

- If there are errors, click **Error Wizard** to review and locate the errors.
- If you want to look at the resulting Help file, click **Run**.

Tip:

You can use the Compatibility Wizard* to give your WinHelp 3 Help system the look and feel of WinHelp 4 systems. You can add a three-tab window that includes a fully functional Contents tab, an Index tab, and a Find tab. After compiling, you can access the three-tab window by clicking a Help Topics button that the Compatibility Wizard inserts on the WinHelp 3 button bar. To have a fully functional Contents tab, it's important to create a table of contents in the TOC tab of RoboHELP Explorer.

* Available only in RoboHELP Office.

About the WinHelp Compatibility Wizard

What is WinHelp Compatibility Wizard?

The WinHelp Compatibility Wizard adds Windows 95 functionality to Windows 3.1 Help files. This means that you can take a Windows 3.1 Help file, and with a few brief commands, make it look just like a Windows 95 Help file, but it will still run perfectly under Windows 3.1. See Using the Compatibility Wizard.

The Compatibility Wizard adds the Contents, Index, and Find+ tabs to new and existing Windows 3.1 Help projects. It registers all the necessary commands in the Help project file. No additional work is required; you just compile the Help project as you would normally.

The Compatibility Wizard also adds a Help Topics button to the button bar in a compiled Help system. This button takes users to the new tabs.

Best of all you do not need the source to the Windows 3.1 Help file. You can run the Compatibility Wizard on a Help (HLP) file.

Tip:

The Compatibility Wizard (rather than the HyperViewer) is best to use when you want to control the process of creating Windows 95 functionality. The Compatibility Wizard is fully configurable! You create the CNT file that configures the Contents tab; so you can design the layout of the Contents tab!

Benefits of Using the Compatibility Wizard

The WinHelp Compatibility Wizard brings many Windows 95 features to Windows 3.1 Help systems. The most important features include the Contents, Index, and Find+ tabs.

Contents tab

- Users can access the Contents tab by clicking the Help Topics button in the menu bar. It reveals an easy-to-read, structured view of a Help system (using books and pages). It allows users to explore a Help system visually and access a desired topic instantly.
- The Contents tab allows users to print all topics in a book at once, rather than having to select and print each topic separately.
- The Contents tab is fully configurable. You decide which books and pages (topics) to include and in what order.

Index tab

- The Index tab works like a traditional book index for a Help system. It lists keywords (words or phrases) alphabetically.
- The Index tab replaces the Search button of Windows 3.1 Help files.
- The Index tab is particularly useful when a user needs to search for Help on a specific topic and the exact wording of the topic is not known.

Find+ tab

- The Find+ tab provides access to a full-text search. This search is conducted by the WinHelp full-text search engine, which is far superior to the standard WinHelp search engine.
- The WinHelp Find+ tab locates every occurrence of a selected word or text string within the Help file.
- Find+ options let the user define the way in which the Find+ tab works.

Using the Compatibility Wizard

Use this procedure to add WinHelp 4 functionality to your WinHelp 3 Help system. (You must have a WinHelp 3 project open to use the Compatibility Wizard.)

1 From the **File** menu, select **Project Settings**, then select the **Extensions** tab.

2 Select the **WinHelp Compatibility Wizard** option.

3 Click the 📝 icon if you want to change the text that appears on the button face of the main Help window (the Help Topics button). The default button text is “Help Topics”, but you can change it to suit your design needs.

4 Click **OK**.

5 Save, compile, and run your project.

Important:

If you generate your WinHelp 3 Help file using a Windows 95 operating system, the Help Topics button will not display in the compiled project. You **will** be able to see the Help Topics button using Windows 3.1 Windows NT 4.0, or Windows 2000.

{ewl RoboEx32.dll, WinHelp2000, }

Generating JavaHelp

You can create JavaHelp from your RoboHELP projects. RoboHELP provides an automated feature that generates all the source files and saves them in a specified subfolder (the default name is "JavaHelp"). You distribute these source files, plus your project-related files (*.HTM, *.GIF, etc.) to your users.

To generate JavaHelp:

1. Open the project containing the files you want to use to generate JavaHelp.
2. From the **File** menu, select **Generate - JavaHelp**. The Single Source wizard opens. You'll be prompted through a series of screens that offer different options for specifying how to set up the JavaHelp output.
3. A default location is given for the output folder. If you want to choose a different location for the output folder, click 📂 and navigate to a new drive/folder. A default output file name is specified for your JavaHelp output file name. If you want to choose a different output file name, you can type one in (you must use the .hs extension). Click **Next** to proceed through the Wizard screens.
4. Click **Finish** when you get to the end of the Wizard prompts.

   Note: Help authors need a copy of the Java Development Kit (version 1.2 or later) and the JavaHelp 1.0 or 1.1 components in order to properly generate and view JavaHelp files. The JavaHelp Development Kit and the JavaHelp components can be downloaded from the Sun Microsystems Web site, www.sun.com.

Viewing JavaHelp

The JavaHelp user interface is similar to the Microsoft HTML Help viewer and includes the following features:

- Table of contents
- Index
- Full-text search
- Customizable display window
- Compressed project file (.JAR file)
- Context-sensitive Help

Since JavaHelp is an HTML-based format, you get the full power of the Web in your Help system. Using JavaHelp's combined power of Help features and HTML functionality, you can uncover new application areas to suit your needs.

After you've generated JavaHelp output, you can view the results in the JavaHelp Help viewer.

To view JavaHelp output:

1 Open the project you want to view.

2 Open the **Single Source** folder and right click on **JavaHelp**.

3 Select **View**.

4 The JavaHelp output opens in the JavaHelp viewer.

- The left pane contains the Contents, Index and Search tabs. For Netscape Navigator, it also displays a Help tab.
- The right pane displays the first topic (default topic).
- Navigate to different topics by clicking on them in the Contents. You can also use the arrows on the toolbar.

Editing JavaHelp

You can use RoboHELP to edit your Help project source files and then generate JavaHelp again.

To edit JavaHelp:

1. Open RoboHELP Classic and the Help project you used to create JavaHelp output.
2. Make your changes.
3. Generate JavaHelp again.

Selecting properties for JavaHelp

All Single Source projects have properties associated with them to define their characteristics and behavior. You can change the following options in the JavaHelp Properties dialog:

- **Target.** Select the name and location for the Help project and to select the source documents to include when importing the project. Access the Define Build Expression dialog to add build tags.
- **Folders.** Set up locations for saving JavaHelp output files. The settings you make in this dialog specify how your output source is organized (folders and subfolders).
- **Features.** Override the default features for JavaHelp popups, What's This? Help, linking external topics, images, and bullets/numbering. We have carefully selected the defaults for you, so you only need to change these settings if you want to customize them.
- **JavaHelp.** Specify which features you want to appear on the JavaHelp navigation (left-hand) pane, specify more conversion selections, and access advanced options in this dialog.

Use this procedure to change any of the default settings for Help format properties.

To select properties for JavaHelp:

1 From the RoboHELP Explorer Project tab, open the **Single Source** folder.

2 Right-click on **JavaHelp** and select **Properties**.

3 Click the tab you want and make changes.

4 Click **OK** when you're done, and generate JavaHelp.

JavaHelp overview

JavaHelp, developed by Sun Microsystems, is a new compressed Help format designed to work with applications written in the Java programming language. JavaHelp and Java applications run on a variety of platforms (Windows, UNIX, Mac, Linux, etc.). JavaHelp is a delivery system, not a Help authoring tool. Sun does not intend to provide a Help authoring tool for creating JavaHelp, but you can use RoboHELP Office to accomplish the task.

RoboHELP provides rich support for the emerging JavaHelp format and automatically creates all the Java-based Help features such as tables of contents, full-text search, dynamic index, navigation controls, and popup windows, along with HTML-based features such as HTML content, hyperlinks, images, etc. You can now create JavaHelp systems with the same point-and-click and drag-and-drop ease as you create WinHelp, HTML Help, and WebHelp — all from the same source project. You can also leverage existing WinHelp and HTML projects to create JavaHelp systems. eHelp's solution, based on Sun's JavaHelp API, also supports file compression for smooth distribution of the JavaHelp system.

What do you need?

Help authors need a copy of the Java Development Kit (version 1.2 or later) and the JavaHelp 1.0 or 1.1 components in order to properly generate and view JavaHelp files. The JDK can be downloaded for free from the Sun Microsystems Web site www.sun.com.

JavaHelp is the preferred format to use with applications written in the Java programming language. If you need to create cross-platform application Help, standalone Help, intranet content, pages for Web sites, or online documentation, we recommend using WebHelp instead of JavaHelp. WebHelp is currently in its third release and is a proven, reliable tool for the uses listed above.

What do your end users need?

Your end users need the Java Runtime Environment (JRE) 1.2.1 and the JavaHelp 1.0 or 1.1 components in order to view your JavaHelp system. For more information see Distributing JavaHelp.

What comes with JavaHelp 1.0 or 1.1?

- A set of file formats that can be used to create Help content files.
- An API that allows Java developers to integrate JavaHelp with their applications. The APIs are standard extensions to the Java Developer Kit (JDK).
- The JavaHelp viewer, an HTML-based Help system viewer. It includes the components necessary to view a JavaHelp system.
- Documentation for Help authors and developers.
- Several example Help systems.

Who should use JavaHelp?

JavaHelp provides a flexible solution for authors who design Help for applications written in the Java programming language. JavaHelp itself was developed in Java code. One of the most important features of Java is its portability across computer platforms. When you develop a Help project in JavaHelp it is cross-platform as well.

How does JavaHelp work?

Behind the scenes, JavaHelp is a compressed file (.JAR file) or a set of files used to run a Help system for a Java application. You generate JavaHelp after you finish authoring Help topics, creating a table of contents, and building the index in RoboHELP. All the files you need to distribute are copied into a single folder called "JavaHelp," or into a .JAR file if you choose to compile.

About the Java programming language

Java is a programming language from Sun Microsystems developed in the early 1990's and released in 1995. Java's main features include:

- **Software portability.** Java programs run on all computer systems that have a Java Virtual Machine (JVM) installed. The Java logo is "write once, run anywhere." After a Java program is compiled, it can be run on a Macintosh, a Windows personal computer, or any other computer with a JVM.
- **Internet capabilities.** Java is a popular language for Web and Internet programming. Java offers developers code libraries for graphical user interfaces (GUIs), Internet access, Internet environment, networking and Java applets (application programs that run in web browsers). Before Java, the Internet was a read-only interface — Internet surfers browsed to URLs and the Web delivered pages. Now that Java applets allow programs to run in web browsers, the Internet is a more interactive environment.
- **Ease of Maintenance.** Java programs are easy to maintain because they can be upgraded from a central server. New applications and enhancements can be loaded from the network and incorporated without recompiling. This allows new features to be incorporated as they are needed, instead of requiring developers to create major new software versions.
- **Security.** Java provides a framework for developers to put security-related features in their applications, including cryptography, encryption, and authentication. Java is designed for safe downloading of executable content, so applets can run without damaging a system.

Features available in JavaHelp

JavaHelp is designed as a comprehensive Help system containing the features that authors depend on and the navigation aids users expect.

- **JavaHelp Help viewer.** The HTML-based viewer is comprised of a toolbar, content pane (uses HTML 3.2 for displaying topics), and navigation pane.
- **Table of contents.** Includes an unlimited number of hierarchical levels and collapsible/expandable topics.
- **Index.** Powerful index that lists keywords and all of the topics a given keyword is associated with.
- **Full-text search capability.** Search engine examines full text of the content and supports multi-word queries. Search results are sorted based on the number of "hits" found in each topic.
- **Flexible display.** JavaHelp can be viewed in its own primary window or embedded in an application. Navigators, content windows and other components can be individually embedded in an application.
- **Customization.** The JavaHelp API provides developers and Help authors with the flexibility to add, combine, replace, customize and extend components and the user interface. Elements of JavaHelp programs that are frequently customized include navigation panes, toolbars and the Help viewer.
- **Context sensitive.** Context sensitive Help and ID mapping is provided in the JavaHelp API through the Swing toolkit of the Java Foundation Classes (JFC).
- **Popups.** Two kinds of popups are supported in JavaHelp. Text-only popups open in a small window within the topic. Popup links are links that open in their own special popup window.
- **Merging options.** The JavaHelp API supports merging of tables of contents, indexes, and search databases.
- **Compression.** JavaHelp projects can be compressed into a single file in the Java Archive (JAR) format. The compression process dramatically reduces the size of your JavaHelp project so it requires less space.
- **Bulleted and numbered lists.** You can include both bulleted lists and numbered lists in JavaHelp projects.
- **Internationalization.** JavaHelp coordinates with the Java Development Kit 1.2 to provide full I18N support. Help content, indexes, and tables of contents are supported and function in a locale-specific way.

Unsupported features in JavaHelp

Listed below are some of the known limitations of JavaHelp. This information is for reference only and may not apply to every JavaHelp project.

For the latest information about JavaHelp, visit the Sun Microsystems Web site at http://java.sun.com/products/javahelp.

JavaHelp unsupported features

- Capitalization
  - All caps
  - Small caps
- Custom colors (Some custom colors do not display accurately in the JavaHelp viewer.)
- Fonts
  - Fonts/Font sets (*1) (Fonts are automatically set to JavaHelp default fonts.)
  - Font size
- Forms (Only very simple forms display properly in the JavaHelp viewer.)
- Framesets
- HTML files generated by Word 2000
- Images
  - Images (Do not display consistently in compressed JavaHelp.)
  - Background images (*1)
  - Image maps
- Links
  - to email addresses
  - to FTP sites
  - to external files
- Lists (Only simple bulleted or numbered lists display properly in the JavaHelp viewer.)
- Multimedia
  - Sound files (such as .WAV or .MIDI)
  - Video files (such as .AVI)
- Paragraph alignment tags (i.e., <P align="center">, <P align="left">, <P align="right">)
- Plain text (non-HTML) files (*1)
- Popup links (The popup window does not have a bottom border if the entire popup is contained within the content frame. This limitation does not apply to text-only popups.)
- Special effects
  - ActiveX controls
  - Dynamic HTML
  - JavaScript
  - Splash screens
  - Visual Basic scripts
- Style sheets (*2)
- Table attributes
  - Values set in the initial tag (TABLE) of a table override cell width specifications.
  - The table width attribute (TABLE WIDTH) functions when specified in absolute pixels, but not in percentages.
  - When used within the TABLE tag, the background color attribute (BGCOLOR) does not function (*1).
  - When used within the TABLE tag, the break tag (BR) causes the table cell to be expanded by # of BR tags used1.
  - Table alignment cannot be set precisely.
- Text animations and effects

(*1) This feature is not supported in JavaHelp 1.0, but is supported in JavaHelp 1.1.

(*2) Partial cascading style sheet support is provided with JavaHelp 1.1 (external style sheets work correctly). The default style sheet used with JDK 1.2.2 is included in the javahelp\doc\css folder when you install JavaHelp. You can use level 1 cascading style sheets with the JDK 1.2.2 or higher, or Swing 1.1.1 with JDK 1.1.

Files generated for JavaHelp

If you choose to compress your JavaHelp project, a .JAR file is created. The .JAR file is similar to the .CHM file created when you compile Microsoft HTML Help. The .JAR contains all of the source files in your JavaHelp project compressed into one file.

If you do not compress your JavaHelp project, all of the source files needed to run your JavaHelp project (.HTM, .GIF, .JPEG, etc.) are contained within the JavaHelp project folder. This folder is created when you generate JavaHelp. It is updated each time you choose to generate JavaHelp again.

Note: Source files are not viewable from the Project tab in the left-hand pane. Use the Windows Explorer to view the JavaHelp folder and corresponding files.-

End-user requirements for JavaHelp

Along with the .JAR file (or the set of source files in your JavaHelp project folder) your users need the JavaHelp 1.0 or 1.1 components and the Java Runtime Environment (JRE) 1.2.1.

If your project is part of a Java application, your developer can integrate your project (.JAR file or source files), the JavaHelp 1.0 or 1.1 components, and the JRE 1.2.1 into the application itself. Or, you can require that your users download the JavaHelp 1.0 or 1.1 components and JRE 1.2.1 directly from Sun Microsystems (www.sun.com).

Tips for creating JavaHelp

Creating a JavaHelp project in a methodical way can help you keep the process well organized. Here are some tips for developing a JavaHelp project:

- **Don't spend time on elements not supported by JavaHelp.** If you know you are a creating a Help project that you intend to transfer into JavaHelp, read through the JavaHelp limitations carefully. That way you won't spend time creating elements and formatting that are not supported in JavaHelp.
- **Incorporate build tags.** Make full use of build tags, especially if you are creating a master project that will be transferred into several other formats (WebHelp, JavaHelp, printed documentation, etc.).
- **Carefully coordinate creation of context-sensitive JavaHelp.** Meet with your developer to plan how you are going to create context-sensitive Help for your JavaHelp project. Identify who is doing what and when.

Advantages of JavaHelp

- **Platform independent.** JavaHelp is written entirely in the Java programming language giving it the ability to run on any platform that supports the Java Runtime Environment (JRE) and has the JavaHelp viewer installed.
- **Customizable and extensible.** Developers can use existing JFC components to customize and extend JavaHelp. This flexibility provides developers with the framework to create custom user interfaces and functionality within JavaHelp projects.
- **Familiar user interface.** JavaHelp Help systems contain a user interface that is similar to the Microsoft HTML Help viewer. The following popular features are supported in JavaHelp:
  - Table of contents
  - Index
  - Full-text search
  - Customizable display window
  - Compressed project file (.JAR file)
  - Context-sensitive Help

Other functionality issues

- **New software.** JavaHelp 1.0 and 1.1 are comprised of new technology that is being changed and updated. As with any new software, early adopters serve as "testers" and usually endure some problems and bugs as they work with the product.
- **Limited features.** Many features commonly included in Help authoring tools such as the ability to create special effects, use a wide variety of custom colors and design detailed tables are not supported in JavaHelp 1.0 and 1.1. They may be available in the future.
- **Customization takes time.** JavaHelp provides a basic, familiar user interface that works well for most JavaHelp projects. In many cases, though, authors, developers, and managers will want a more professional Help system that supports advanced features. This customization work requires the time of developers, who often face heavy time constraints.
- **Printing.** JavaHelp 1.0 does not support printing. In JavaHelp 1.1 or later, a print button is included on the JavaHelp viewer toolbar so you can print topics. (For the best printing results in your application's Help, JDK 1.2.2.-002 must be used in the application. JDK 1.2.2-002 is available in the JavaHelp 1.1 Beta JRE folder.)
- **HotJava browser.** HotJava is a web browser developed by Sun Microsystems. JavaHelp uses the HotJava browser to render HTML code. HotJava is an acceptable browser, but is not as powerful as Internet Explorer or Netscape Navigator.

Distributing JavaHelp

JavaHelp can run on any platform including Windows, MacOS, UNIX, Linux and Sun Solaris. When you generate JavaHelp, all the source files are saved in a subfolder in your project folder. By default, the name of the subfolder is JavaHelp, although you can change the name if necessary.

For your JavaHelp project to work properly and efficiently on any platform, include the following:

- If the JavaHelp project is compressed, include the .JAR file.
- If the JavaHelp project is not compressed, include all of the individual source files in your JavaHelp output folder (such as .HTM, .GIF, .JPEG, .WAV, etc.). For more information about specific files, see Files generated for JavaHelp.
- The JavaHelp 1.0 or 1.1 components.
- Java Runtime Environment (JRE). You can include the JRE with your Help system or users can download this from the Sun Microsystems Web site (www.Sun.com).

What is context-sensitive JavaHelp?

Context-sensitive Help provides information about what users see inside applications — details about fields and controls at dialogs, explanations of messages, and descriptions of windows and screen objects. It is called "context-sensitive" because each topic is called by the application and because it provides information relevant to the task the user is trying to accomplish.

With JavaHelp, authors can create full-featured Help systems that include context-sensitive Help. Creating context-sensitive JavaHelp can be done with the assistance of a developer who can do custom programming.

As with any type of context-sensitive Help, developing context-sensitive JavaHelp is a cooperative effort between Help authors and developers.

Author's role. Creates topics and writes content. Each context-sensitive Help topic is a separate file that describes how to use an application at the dialog or window-level.

Application developer's role. Writes the Java code that enables context-sensitive Help to work properly. Programs the Help topics in the application so the correct ones are displayed when users request help (calls correct item in .JR file).

Creating context-sensitive JavaHelp

Creating context-sensitive JavaHelp is a collaborative effort between authors and developers. JavaHelp supports context-sensitive Help, but all of the code to make it work properly must be written and customized by your developer.

Your developer manually adds the string to the application code so the Help can be invoked. The Java application "calls" the context sensitive Help using the string.

The JavaHelp equivalent of the .hh file for context-sensitive Help is the .JHM file. It is a good idea to print a copy of the .JHM file for your application developer to use while working on context-sensitive Help.

The example below shows one way to invoke a JavaHelp Help set:

```
class ContextHelp
 {
  private HelpSet hs;
   private HelpBroker hb;
    public boolean ShowHelp(String strTopic, String strHelpSet)
     {
      if (hb == null)
       {
        ClassLoader loader = getClass().getClassLoader();
         URL url;
          try
           {
            url = HelpSet.findHelpSet(loader, strHelpSet);
            hs = new HelpSet(loader, url);
           }
         catch (Exception e)
       {
       return false;
       }
       hb = hs.createHelpBroker();
     }
   hb.setCurrentID(strTopic);
  hb.setDisplayed(true);
 return true;
 }
}
```

Testing context-sensitive JavaHelp

Testing context-sensitive JavaHelp, just like creating it, is a collaborative effort between authors and developers.

To test context-sensitive JavaHelp:

1. **Compress the JavaHelp project.** If you are creating context-sensitive Help for a Java application, compile the project and create a .JAR file. This file contains all of the files in your project including the context-sensitive topics. Your developer needs to place the .JAR file in your application’s extensions folder (or place it in the class path that is passed to the Java Virtual Machine.
2. **Test the context-sensitive topics.** In order to test, you need to have access to the application. It's important to test the topics in the application to make sure that all of them are displayed at the correct dialogs and windows in the application.
3. **Fix errors.** Different types of errors can occur so it is important to work closely with your developer to solve any problems.

See Also in JavaHelp

If your original project contains See Also references, you can retain them when you generate JavaHelp files. A JavaHelp control translates the original See Also reference control into a new one designed specifically for JavaHelp.

The file containing the JavaHelp control is named BSSCJHRC.JAR. Let your developer know that they need to place the BSSCJHRC.JAR file in your application’s extensions folder, or place it in the class path that is passed to the Java Virtual Machine (the Java.exe).

To test the JavaHelp controls:

1 Write your topics in RoboHELP and incorporate See Also references.

2 Generate JavaHelp source files.

3 View your JavaHelp project.

Note: If two, small red question marks appear in place of your See Also references, the BSSCJHRC.JAR is not in the proper location. Ask your developer to place the BSSCJHRC.JAR in the correct location.

Generating Microsoft HTML Help

You can create and customize Microsoft HTML Help (.HHP file, .CHM file, and .HTM files) using your Help project and RoboHELP Classic's Single Source. The Microsoft HTML Help project is created based on how you instruct the program to convert the .HPJ file.

To create Microsoft HTML Help:

1 Start RoboHELP Classic and open the WinHelp project you want to convert.

2 From the RoboHELP Explorer, select **Generate** from the **File** menu, then select **Microsoft HTML Help**.

3 The Single Source wizard opens. You'll be prompted through a series of screens that offer options for setting up the HTML Help project.

4 Select the output folder and file name where the HTML Help project files will be stored. (The default path is the project folder, and the default file name is the project file name with a .CHM extension.) Click 📂 to navigate to the drive/folder where you want the output file to be stored, or type in the name of the output file.

5 Select the source documents (.DOC files) you want to generate as HTML files. Click **Next** to proceed through the Wizard screens.

6 Click **Finish** when you get to the end of the Wizard prompts.

Files generated for Microsoft HTML Help

RoboHELP's One-Click Single Source generates the following files for Microsoft HTML Help:

- *.CHM – compiled HTML Help file (equivalent of the *.HLP in WinHelp)
- *.HTM – the topics in your Help system
- *.HHC – the contents file
- *.HHK – the index file
- *.GIF or *.JPG – image files (created from the image files in the WinHelp source)
- *.WAV, *.AVI, *.WAV – multimedia files
- ERRORLOG.ERR – the conversion log, including any errors or features that didn't convert

You'll find these files in the \HtmlHelp folder under the RoboHELP project folder (unless you specified another location in the Single Source Wizard).

Viewing Microsoft HTML Help

Microsoft HTML Help uses a shared layout engine to display content in Microsoft's tri-pane HTML Help viewer, which works in much the same way as the Internet Explorer browser and includes the following features:

- A toolbar on the top pane that provides buttons for showing and hiding the left pane, browsing and printing topics, and more. If you create a custom window, you can specify the toolbar buttons to display in the viewer.
- A navigation pane on the left side that displays tabs for the Contents, Index, and Search features.
- A topic pane on the right side that displays topic content, with scroll bars for further viewing. In the HTML Help viewer, the selected book or page in the left pane is synchronized with the active topic in the right pane.

After you've created Microsoft HTML Help using RoboHELP Classic's Single Source wizard, you can view the HTML Help file at any time right from RoboHELP Classic.

To view Microsoft HTML Help:

1. Start RoboHELP Classic and open the WinHelp project you want to convert.
2. Open the **Single Source** folder and right-click on **Microsoft HTML Help**.
3. Select **View**.
   RoboHELP launches the application specified as the Microsoft HTML Help viewer. If you don't specify a viewer, RoboHELP launches the CHM (compiled HTML Help file) and displays the tri-pane HTML window.

   Consult your HTML viewer application for details on how to open HTML Help projects.

Tips:

- To select your preferred HTML Help viewer, select a viewer on the Single Source tab in RoboHELP Options. See Selecting a Viewer for Microsoft HTML Help.
- You can also display Microsoft HTML Help in Microsoft's Internet Explorer (this is the browser that supports ActiveX controls). For full functionality, run your HTML Help systems in Internet Explorer 4.x or later.

Using build tags

Use the Define Build Tag Expression dialog to add build tag expressions to projects. Build tag expressions are supported with Microsoft HTML Help, WebHelp, and JavaHelp.

The purpose of a build expression is to specify the topics you want to include and/or exclude from a project. To use build tag expressions, your project needs to include build tags and they need to be assigned to topics. You can assign build tags to topics in RoboHELP Classic.

To define build tags for HTML Help:

1. Start RoboHELP Classic and open the WinHelp project you want to convert.
2. From the RoboHELP Explorer menu, select **File** – **Generate** – **Microsoft HTML Help**. The Single Source wizard opens.
3. In **Build Expression**, click **Define**. The **Define Build Tag Expression** dialog opens.
4. Select a build tag from **Build Tags** or click **Add Tag** to add a new one. The new build tag displays in the **Build Tag Expression** box.
5. Select a Boolean operator to use with the build tag expression:
   - AND includes or excludes topics that use specific build tags
   - OR includes or excludes topics that use specific build tags.
   - NOT excludes topics that use a specific build tag.
6. Click **OK** to accept the changes or **Clear** to remove a build tag expression.

Microsoft HTML Help Advanced Settings

You can select options to customize your tri-pane window, TOC, and buttons when you generate HTML Help.

To select advanced settings:

1 Start RoboHELP Classic and open the WinHelp project you want to convert.

2 Select **File – Generate – Microsoft HTML Help**. The Microsoft HTML Help Wizard opens. Click **Next** until you see the **Microsoft HTML Help – Options** dialog.

3 Click Advanced. Select the options you want in each tab (see below).

Tri-Pane options

You can customize the buttons you display on your browser or HTML viewer.

- Select display options for the navigation pane.
- Select the font for the TOC tab. Currently, font type and size can be specified.

TOC options

- Customize the way the table of contents looks and behaves. Select styles and the appearance of TOC edges.

Custom Buttons

Specify up to two custom buttons, if you want to add a button that links to a URL address.

- Select **Show Custom Button 1** and enter the text for the button face. Then enter the URL address (e.g., http://www.blue-sky.com).
- Do the same for **Custom Button 2** if you want a second custom button.

Tip:

- Custom Button 1 looks like this on the button bar:
- Custom Button 2 looks like this on the button bar:

Editing Microsoft HTML Help

You can use RoboHELP to edit your Help project source files and then generate Microsoft HTML Help again.

To edit Microsoft HTML Help:

1 Open RoboHELP Classic and the Help project you used to create Microsoft HTML Help output.
2 Make your changes.
3 Generate Microsoft HTML Help again.

Selecting an editor for Microsoft HTML Help

The Single Source tab under RoboHELP Options allows you to specify an HTML editor when editing your Microsoft HTML Help system. You can select another HTML editor, but most HTML editors only allow you to edit individual HTML pages rather than the entire Microsoft HTML project.

To select an editor for Microsoft HTML Help:

1 From the RoboHELP Explorer **Tools** menu, select **Options**.

2 Select **Single Source**.

3 Select **Microsoft HTML Help**.

4 In **Editor**, enter the path or navigate to the appropriate program and click **Open**.

Note: You can only edit individual HTML files, not compiled HTML Help files (.CHM files).

Selecting a Viewer for Microsoft HTML Help

RoboHELP Classic allows you to select the viewer for HTML Help output. If a viewer field is blank for a Single Source selection such as HTML Help, RoboHELP uses the default from your operating system.

Compiled Microsoft HTML Help is automatically displayed in the HTML Help viewer (a special window exclusively designed for viewing HTML Help, based on Internet Explorer components). You can also view Microsoft HTML Help in Internet Explorer.

The first time you generate output, RoboHELP Classic prompts you to view the files. After that, you can view the output any time by right clicking the output format under **Single Source** (in the **Project tab**) and selecting **View**.

To specify a viewer for Microsoft HTML Help:

1 From the RoboHELP Explorer menu, select **Tools – Options**.

2 Select the **Single Source** tab.

3 Select **Microsoft HTML Help**.

4 In **Viewer**, enter the path of your preferred viewer or navigate to the appropriate program and click **Open**.

Selecting properties for Microsoft HTML Help

All Single Source projects have properties associated with them to define their characteristics and behavior. You can change the following options in the Microsoft HTML Help Properties dialog:

- **Target.** Select the name and location for the Help project and to select the source documents to include when importing the project. Access the Define Build Expression dialog to add build tags.
- **Folders.** Set up locations for saving HTML topic files and images. The settings you make in this dialog specify how your output source is organized (folders and subfolders).
- **Formatting.** Specify how you want to format the HTML topic files. You have the option of working with or without style sheets.
- **Features.** Override the default features for popups, What's This? Help, linking to external topics, images, and bullets/numbering. We have carefully selected the defaults for you, so you only need to change these settings if you want to customize them.
- **Microsoft HTML Help.** Specify which features you want to appear on the HTML Help navigation (left-hand) pane, specify more conversion selections, and access advanced options in this dialog.

Use this procedure to change any of the default settings for Help format properties.

To select properties for Microsoft HTML Help:

1 From the RoboHELP Explorer Project tab, open the **Single Source** folder.

2 Right-click on **Microsoft HTML Help** and select **Properties**.

3 Click the tab you want and make changes.

4 Click **OK** when you're done, and generate Microsoft HTML Help.

Select Folders and File Names

This dialog prompts you to specify the folder or folders where your HTML and GIF files are to be placed.

Select folder for HTML files

Type the full path of the location where you want to store your HTML files or click 📂 to select the desired destination folder.

Select folder for GIF files

GIFs must be stored in the same directory or one subfolder below your HTML files. For example, if the HTML files are stored in \TUTORIAL, then the GIFs must be stored in either \TUTORIAL or in a subfolder of \TUTORIAL (for example, \TUTORIAL\GRAPHICS).

Short filenames (8.3 format)

The Single Source Wizard names the HTML files it creates by combining the name of the selected Help file with the names of the Help Topics. For a Help file, LITERATURE.HLP, the topic "Hemingway" becomes "LiteratureHemingway.html".

Enabling short filenames creates DOS 8.3 filenames. The HTML files begin with the first three characters of the Help file’s name followed by "00001," "00002," etc. So, if "Hemingway" were the 12th topic in LITERATURE.HLP, it would be converted to "Lit00012.htm".

Prefix

By default, when you enable the short filenames option, the Single Source Wizard begins the names of the files it generates with the first three characters of the Help file’s name. You can enter a different three-character prefix here.

Select style sheet and numbered list options

Select the style sheet to use

HTML style sheets are similar to Microsoft Word templates in that they automatically format your topic text. Select one that suits your design needs. You can update the styles in the style sheet after your Help project is converted.

<No Stylesheet>

If you do not want to attach an external style sheet to your HTML Help topics, select **<No Stylesheet>**, which applies the default HTML styles to all of the topics (gray background, black serif font). You can always attach style sheets to topics after the project is converted, so it's not critical that you select one at this point.

<Embedded Styles>

If you want your HTML Help topics to look exactly like your WinHelp topics, select **<Embedded>**. This embeds the formatting within the HTML Help topics instead of applying external style sheets to them.

Use explicit body style (use only for IE 3.x)

If your HTML Help is intended for Internet Explorer 3.x viewing, select **Use explicit body style** and a style sheet. (This feature writes style definitions in the HTML <BODY> tag so IE 3.x can handle formatting.) A good style sheet to use for IE 3.x viewing is "Win 95 sec window style."

Convert Bulleted Lists

For most bulleted lists, you'll want to select this option. The resulting bulleted list is in HTML Bulleted list style. If you have fancy bullets, you may want to clear this option. Clearing this option displays your bulleted lists as text without the list indent.

Convert Numbered Lists

For most numbered lists, you'll want to select this option. The resulting numbered list is in HTML Numbered list style. If you have really complex lists, you may want to clear this option. Clearing this option displays your numbered lists as text without the list indent.

HTML conversion options

Browse buttons (Prev / Next)

If your original Help file contains a browse sequence, selecting this option adds Previous and Next buttons to your HTML pages. Using these buttons, your users can emulate the browse sequence from the original Help file. (Remember that these buttons follow the same browse sequence established in the original Help file. They are not related to the browser's Back and Next buttons in any way.)

Browse button options

If you select the Browse buttons option described above, choose the top or bottom of the page as the location for the buttons. (Remember that these buttons follow the same browse sequence established in the original Help file. They are not related to the browser's Back and Next buttons in any way.)

Use hotspot text instead of buttons

If you want your pages to load faster, select text only format. When you enable this option, the Browse buttons appear as hypertext jumps rather than as graphical buttons.

Change white in graphics to transparent

Select this option to make the white areas in graphics transparent. If you want the colors of your graphics to remain as they appear in your Help file, clear this option.

Single Source Wizard: Welcome

Use this dialog to select the name and location for the Help project and to select the source files to include when generating the output.

Select output folder and output file name: Selects the path and appropriate file name to use with the Help project.

Use short folder/file names (WebHelp conversions only): Assigns file names that use no more than eight characters (plus three-character extension). Use this feature if you are deploying an intranet or Web site on a Web server that does not support long file names. (Ask your webmaster if the server supports long files names, if you’re not sure.)

Choose source document(s): Selects the .DOC files to import. By default, all documents in the project are selected.

Select All: Adds all files to the selection.

Clear All: Clears all files from the selection.

Build Expression: Click **Define** to select or add build tag expressions to Help projects.

Define Build Tag Expression dialog

Use the Define Build Tag Expression dialog to add build tag expressions to projects. Build tag expressions are supported with HTML Help, WebHelp, and JavaHelp.

The purpose of a build expression is to specify the topics you want to include and/or exclude from a project. To use build tag expressions, your project needs to include build tags and they need to be assigned to topics.

- **Build tag expression:** Defines the expression to use with the build tag.
- **Add Tag:** Adds selected tag to Build Tag Expression box.
- **AND:** Includes or excludes topics that use specific build tags.
- **OR:** Includes or excludes topics that use specific build tags.
- **NOT:** Excludes topics that use a specific build tag.
- **Clear:** Removes build tag expression.
- **Build Tags:** Displays a list of all build tags that are defined for the project.

Example

Single Source Wizard: Folders

Use the Folders dialog to set up locations for saving HTML topic files and images. The settings you make at this dialog specify how your output source is organized (folders and subfolders).

Folder for HTML Files

- **Do not create subfolder:** Saves all HTML topic files in the root directory you've chosen for your output.
- **Create subfolder:** Saves all HTML topic files in a subfolder, based on name entered in text box.
- **Create subfolders based on source document names:** (Default) Creates and saves HTML topic files in subfolders, based on the names of the .DOC files.
- **Create subfolders based on TOC structure:** Creates and saves HTML topic files in subfolders, based on the titles of books in the WinHelp table of contents. Books within books are created as subfolders within folders and topics are arranged accordingly. All topics not used in the WinHelp table of contents are saved in the root directory you've chosen for your output.

Folders for Images

- **Do not create subfolder:** Saves all image files in the root directory.
- **Create subfolder:** (Default) Saves all image files in a subfolder, based on name entered in text box.

File Options

- **HTML File Extension:** Saves all HTML topic files using the extension selected from the drop-down. Choices include HTM (default) and HTML.
- **Always use lowercase:** Saves all file names in lowercase letters. This will help to eliminate problems if your Help files are being accessed on a UNIX server (UNIX is a case-sensitive platform).

Single Source Wizard: Formatting

Use the Formatting dialog to specify how you want to format the HTML topic files. You have the option of working with or without style sheets. If you don’t want to use style sheets, the HTML topics will be formatted using inline styles (which is like applying manual character and paragraph formatting in Word). If you want to use style sheets, you can create new style sheets based on the formatting in the WinHelp topics or you can select a style sheet you already use with other HTML topics.

Inline Styles (no linked style sheets): (Default for WebHelp) Converts all document formatting into HTML inline styles which are similar to character and paragraph formatting in Microsoft Word. This option does not involve creating or applying styles or style sheets to the topics. All formatting is handled on a paragraph-by-paragraph and/or character-by-character basis. You can apply styles to the topics after you import the project. If you want new topics you create to look like the ones you generate, you’ll need to manually apply character and paragraph attributes to the text.

Linked style sheets (one for each source document): (Default for Microsoft HTML Help) Creates one style sheet file (.CSS) for each .DOC file that you included as a source document. If the formats used in each .DOC file are unique, separate style sheets will incorporate these unique styles so you can continue to use them in the HTML Help project. For example, if your WinHelp project has two .DOC files, california.doc and arizona.doc, two style sheets are created, california.css and arizona.css. All topics in california.doc are linked to california.css in the HTML Help precut and all topics in arizona.doc are linked to arizona.css.

Linked style sheet based on one source document: Creates a single external style sheet (.CSS file) based on one of the .DOC files used in the conversion. For example, if the WinHelp project has two .DOC files (california.doc and arizona.doc) and you select california.doc, one style sheet is created (the style sheet will have the same name as the project, e.g., states.css). All HTML topics are linked to this style sheet and the formatting in all of the topics is based on the styles in states.css.

Linked style sheet based on existing style sheet: Links an existing style sheet to all HTML topics. For this feature to be effective, the styles in the .DOC files need to have a one-on-one relationship with the styles in the style sheet*. This means that styles in both the documents and the style sheet should use the exact same names. The styles should also be applied to the appropriate paragraphs and characters that need to use them. If not, the style sheet will provide the styles that you can apply to the text, but the text will be formatted manually (since style names did not match the ones in the existing style sheet).

Minimum Font Size: (WebHelp only) This option resolves problems caused by some browsers and platforms that do not display small fonts well. You can experiment to see which fonts are the most readable on your end users’ systems and set the font accordingly. (The default minimum font setting is 9 point.)

* The two most common Microsoft Word styles, **Normal** and **Heading 1**, are automatically mapped to the **<P>** and **<H1>** HTML styles, respectively.

Single Source Wizard: HTML Help Features

This wizard dialog allows you to override the default features. We have carefully selected the defaults for you. You only need to change these settings if you want to customize the settings.

Popups: Converts WinHelp popups into Smart Popups or normal HTML hyperlinks.

- **Smart Popups*:** Creates links that display destination topics in popup windows that size to content. These popup windows support all HTML formatting.
- **Regular hyperlinks:** Creates links that display destination topics in the same window where topic containing link is displayed.

What’s This Help: Specifies how to convert context-sensitive WinHelp topics.

- **Text-only Topic files*:** Creates and saves topics in text-only format (.TXT). Use this option for What’s This? Help topics.
- **Individual HTML files:** Creates and saves topics as regular HTML topic files (to use as window-level context-sensitive Help).
- **Remove Topics:** Removes topics from the single source conversion.

External Topics: Specifies how to convert external WinHelp topic links.

- **Link to external HTML Help topic*:** Converts external WinHelp topic links to external HTML Help topic links (links to HTML files in external .CHM files). Use this option if you plan on merging multiple WinHelp projects into one HTML Help system.
- **Retain:** Keeps external WinHelp topic links. Use this option if you plan on linking HTML topics to WinHelp topics. The HTML topics will include WinHelp Topic controls (special ActiveX controls that open WinHelp .HLP files).
- **Remove:** Removes links to external WinHelp topics.

Images: Conversion option for .BMP images.

- **GIF*:** A compressed bitmap format.
- **JPEG:** A compressed true-color or high-color bitmap format.

Bullets / Numbering: Specifies how to convert WinHelp bulleted and numbered lists.

- **HTML bullets and numbering*:** Converts WinHelp bulleted and numbered lists into HTML bulleted and numbered lists. These lists are auto-numbered or bulleted.
- **Formatted Text:** Turns WinHelp bulleted and numbered lists into text paragraphs. Numbers are hard-coded as text. A symbol character is used to represent the bullet.

* = recommended: All options that include an * indicate the recommended setting.

Single Source Wizard: JavaHelp Features

This wizard dialog allows you to override the default features. We have carefully selected the defaults for you. You only need to change these settings if you want to customize the settings.

Popups: Converts WinHelp popups into Smart Popups or normal HTML hyperlinks.

- **JavaHelp Popups*:** Creates JavaHelp popups that open in its own popup window.
- **Regular Hyperlinks:** Creates links that display destination topics in the same window where topic containing link is displayed.

What's This Help: Specifies how to convert context-sensitive WinHelp topics.

- **Individual HTML files:** Creates and saves topics as regular HTML topic files (to use as window-level context-sensitive Help).
- **Remove topics:** Removes topics from the single source conversion.

External Topics: Specifies how to convert external WinHelp topic links.

- **Link to external JavaHelp topic*:** Converts external WinHelp topic links to external HTML topic links (links to HTML files in external folders). Use this option if you plan on merging multiple WinHelp projects into one JavaHelp system.
- **Retain:** Keeps external WinHelp topic links. Use this option if you plan on linking HTML topics to WinHelp topics. The HTML topics will include WinHelp Topic controls (special ActiveX controls that open WinHelp .HLP files).
- **Remove:** Removes links to external WinHelp topics.

Images: Conversion option for .BMP images.

- **GIF*:** A compressed bitmap format.
- **JPEG:** A compressed true-color or high-color bitmap format.

Bullets / Numbering: Specifies how to convert WinHelp bulleted and numbered lists.

- **HTML bullets and numbering*:** Converts WinHelp bulleted and numbered lists into HTML bulleted and numbered lists. These lists are auto-numbered or bulleted.
- **Formatted Text:** Turns WinHelp bulleted and numbered lists into text paragraphs. Numbers are hard-coded as text. A symbol character is used to represent the bullet.

Minimum Font Size: This option resolves problems caused by some browsers and platforms that do not display small fonts well. You can experiment to see which fonts are the most readable on your end users' systems and set the font accordingly. (The default minimum font setting is 10 point.)

* = recommended: All options that include an * indicate the recommended setting.

Single Source Wizard: WebHelp Features

This wizard dialog allows you to override the default features. We have carefully selected the defaults for you. You only need to change these settings if you want to customize the settings.

Popups: Converts WinHelp popups into Smart Popups or normal HTML hyperlinks.

- **Smart Popups*:** Creates links that display destination topics in popup windows that size to content. These popup windows support all HTML formatting.
- **Regular hyperlinks:** Creates links that display destination topics in the same window where topic containing link is displayed.

What's This Help: Specifies how to convert context-sensitive WinHelp topics.

- **Individual HTML files*:** Creates and saves topics as regular HTML topic files (to use as window-level context-sensitive Help).
- **Remove topics:** Removes topics from the single source conversion.

External Topics: Specifies how to convert external WinHelp topic links.

- **Link to external WebHelp topic*:** Converts external WinHelp topic links to external HTML topic links (links to HTML files in external folders). Use this option if you plan on merging multiple WinHelp projects into one WebHelp system.
- **Retain:** Keeps external WinHelp topic links. Use this option if you plan on linking HTML topics to WinHelp topics. The HTML topics will include WinHelp Topic controls (special ActiveX controls that open WinHelp .HLP files).
- **Remove:** Removes links to external WinHelp topics.

Images: Conversion option for .BMP images.

- **GIF*:** A compressed bitmap format.
- **JPEG:** A compressed true-color or high-color bitmap format.

Bullets / Numbering: Specifies how to convert WinHelp bulleted and numbered lists.

- **HTML bullets and numbering*:** Converts WinHelp bulleted and numbered lists into HTML bulleted and numbered lists. These lists are auto-numbered or bulleted.
- **Formatted Text:** Turns WinHelp bulleted and numbered lists into text paragraphs. Numbers are hard-coded as text. A symbol character is used to represent the bullet.

* = recommended: All options that include an * indicate the recommended setting.

Single Source Wizard: HTML Help options

You can specify which features you want to appear on the HTML Help navigation (left-hand) pane and access advanced options in this dialog.

Navigation Pane

- To include TOC and Index tabs, select **TOC & Index**.
- To include a Favorites tab, allowing you to annotate (or bookmark) topics by adding them to the Favorites tab, select **Favorites**.
- In **Search**, select **Regular** if you want to provide simple searches based on all or part of a topic’s title. If you don’t want the Search tab, select **No Search**. An **Advanced** search provides a more thorough search based on topic properties.
- In **Tab Position**, select where the TOC, Index, and Search tabs should be displayed in the HTML Help viewer or browser (**Top**, **Left**, or **Bottom**).
- In **Default Tab**, select the tab that should be displayed when the browser or HTML viewer is opened.

Options

- Select **WebSearch** to enable the WebSearch button on your main button bar. WebSearch allows end users to perform Internet searches about the Help topics and application they’re using right from the compiled Help window. (Users need Internet access to use the WebSearch button.)
- **Create Browse Sequences** enables you to keep the browse sequences you’ve set in your Help project.
- **Ignore Secondary windows** is selected by default. HTML Help doesn’t support secondary windows, so they are automatically converted to popup windows.
- Select **Create Compiled CHM only** if you want to output a compiled HTML Help file and not the .HHP file and the individual HTML files. The default is to generate the .CHM along with the project and HTML files.
- Select where to add index keywords in **Add Keywords to**. If you add them to **Each Topic** (default), you can use the HTML files in other projects. If you add them to the **Index File (HHK)**, they are included in a separate file in the same directory as the HTML Help project you are creating.
- Click Advanced to customize options for your tri-pane window, TOC, and custom buttons.

Advanced: Tri-pane options

You can select additional options to customize your tri-pane window, TOC, and custom buttons when you generate HTML Help.

Buttons

You can customize the buttons you display on your browser or HTML viewer.

- **Hide/Show** allows you to hide or display the left tri-pane window.
- **Back** and **Forward** allow you to navigate backwards or forward through topics.
- **Stop**, **Refresh**, **Home**, and **Print** are standard browser buttons.
- **Options** displays the standard tri-pane options. In addition, the Options button provides **Internet Options** (for standard Internet settings found on most browsers) and **Search Highlight Off** (search terms that are found in topic files will not be highlighted).
- **Locate** synchronizes the left and right pane (when Auto Synchronize is not selected).
- If you selected the **Home** button to be displayed, specify the **Home URL** by typing the Internet address.

Navigation Pane

The navigation pane is the left-hand side of the HTML Help viewer (or the tri-pane window). You can customize how it’s displayed in this dialog.

- **Hide Nav Pane on Startup** hides the left tab components in the tri-pane window.
- **Auto Show/Hide Nav Pane** automatically hides the left-hand navigation pane when users switch focus from a compiled Help file to a software program.
- **Auto Synchronize TOC** specifies if the left and right panes should be automatically synchronized when the user navigates the content.
- **Width** defines the width of the navigation pane in pixels.

Font

- Select the font to appear on the Contents and Index tabs. Currently, font type and size can be specified. Select **Default** to set the font back to your system’s default setting.

Advanced: TOC options

You can use this dialog to customize the table of contents used with HTML Help.

Style:

- **Border:** Adds a border around the TOC pane.
- **Dialog Frame:** Adds a frame around TOC pane.
- **Lines from Root:** Displays lines connecting books and pages starting at root.
- **Plus/Minus Squares:** Displays plus and minus squares that open and close books when clicked (Lines from Root must be selected).
- **Always Show Selection:** Shows selected page even if TOC pane is not the currently selected item (does not have focus) (Auto Synchronize TOC in Tri-Pane options must be unselected).
- **Folders Instead of Books:** Displays folder icons instead of book icons in the TOC pane.
- **Single Click to Open Book:** Enables books to open with a single-click rather than a double-click.
- **Lines Between Items:** Adds lines between books and pages.
- **Raised Edge:** Formats the table of contents so it looks "raised" from the tri-pane (giving it a 3D effect).
- **Sunken Edge:** Formats the table of contents so it looks "sunken" into the tri-pane (giving it a 3D effect).

Advanced: Custom Button options

You can specify up to two custom buttons if you want to add a button to your HTML Help project that links to a URL address.

- Select **Show Custom Button 1** and enter the text for the button face. Then enter the URL address (e.g., http://www.blue-sky.com).
- Do the same for **Custom Button 2** if you want a second custom button.

Tip:

- Custom Button 1 looks like this on the button bar:
- Custom Button 2 looks like this on the button bar:

Single Source Wizard: WebHelp options

You can select options to set up your navigation pane tabs and other options in this dialog.

Navigation Pane

The navigation pane is the left-hand side of the tri-pane window.

- Select the tabs you want displayed in your WebHelp project. You can display the **TOC** (Contents), the **Index**, and the **Search** tabs. Clear the tabs you don’t want to appear.
- In **Preferred Format**, select **Dynamic HTML** or **Java Applet**, depending on the user interface and functionality you prefer.

In Topic Navigation Bar

The topic navigation bar allows you to place browse sequence navigation and the WebSearch button in every topic.

- Select **Browse Sequences** to keep the browse sequences you’ve set up in your Help project. (These buttons follow the same browse sequence established in the original WinHelp file. They are not related to the browser’s Back and Next buttons.)
- Select **WebSearch** to enable WebSearch functionality. WebSearch allows end users to perform Internet searches about the Help topics and application they’re using right from the compiled Help window. (Users need Internet access to use the WebSearch button.)
- Select the **Location** of your browse sequence arrows and WebSearch button. You can choose to display these options on the **Top Left**, **Top Right**, **Bottom Left**, or **Bottom Right** of each topic.
- Select the **Style** of the browse sequence and WebSearch. **Text** creates hotspot links for the browse arrows and hotspot text for WebSearch (“WebSearch”), and **Button** creates buttons for these options with text labels.

Prefix HTML files with Help project name

- Select this option if you want each HTML file generated to begin with the Help project name.

Single Source Wizard: JavaHelp options

You can specify which features you want to appear on the JavaHelp navigation pane and whether your file names are prefixed with the JavaHelp project name.

Navigation Pane

The navigation pane is the left-hand side of the tri-pane window.

- Select the panes you want displayed in your JavaHelp project. You can display the **TOC** (Contents), the **Index**, and the **Full-Text Search** panes. Clear the panes you don’t want to appear.

In Topic Navigation Bar

The topic navigation bar allows you to place browse sequence navigation and the WebSearch button in every topic.

- Select **Browse Sequences** to keep the browse sequences you’ve set up in your Help project. (These buttons follow the same browse sequence established in the original WinHelp file. They are not related to the browser’s Back and Next buttons.)
- Select **WebSearch** to enable WebSearch functionality. WebSearch allows end users to perform Internet searches about the Help topics and application they’re using right from the compiled Help window. (Users need Internet access to use the WebSearch button.)
- Select the **Location** of your browse sequence arrows and the WebSearch button. You can choose to display these options on the **Top Left**, **Top Right**, **Bottom Left**, or **Bottom Left** of each topic.
- Select the **Style** of the browse sequence and WebSearch. **Text** creates hotspot links for the browse arrows and hotspot text for WebSearch (“WebSearch”), and **Button** creates buttons for these options with text labels.

Output

You can select the type of JavaHelp output you want to generate.

- If you want to create a .JAR file and .HS file, select **Compressed JavaHelp**.
- If you want to create a .JAR file, .HS file, and individual source files, select **Compressed JavaHelp with source files**.
- If you want to create an .HS file and individual source files, select **Uncompressed JavaHelp with source files**.

Prefix HTML files with Help project name

- Select this option if you want each HTML file generated to begin with the Help project name. Use this option if you plan on merging multiple WinHelp projects into one JavaHelp system.

Single Source Wizard Result

This window lets you know that the Single Source Wizard successfully generated the output files. You can see the new Help file by clicking **View Result**. Click **Publishing** to copy your output files to a new location. If you do want to view or publish files, click **Close** to remove this window.

JavaHelp - Requirement

This dialog is displayed when you choose to generate JavaHelp. If you do not have JavaHelp 1.0 or 1.1 installed on your hard drive, you will not be able to include a full text search in your JavaHelp project.

- **Generate without Full Text Search:** Generate JavaHelp source files without creating full text search capability.
- **Locate JavaHelp Install folder:** Displays your project hierarchy so you can locate the JavaHelp install folder. Click **Network** if the install folder is located on your corporate network.

JavaHelp - JDK Requirement

This dialog is displayed when you choose to generate JavaHelp. If you do not have the Java Development Kit (JDK) 1.2 or later installed on your hard drive, you will not be able to generate compressed JavaHelp (a .JAR file).

- **Generate Uncompressed JavaHelp:** Generate JavaHelp source files without creating a compressed .JAR file.
- **Locate JDK Install folder:** Displays your project hierarchy so you can locate the JDK install folder. Click **Network** if the install folder is located on your corporate network.

JavaHelp - Unsupported features

This dialog is displayed when you choose to generate JavaHelp. Some RoboHELP features are not supported by JavaHelp. Some of these limitations are listed in the text box. For more information, refer to Unsupported features in JavaHelp.

- **Do not show this message again:** Select this checkbox if you do not want to see this dialog box each time you generate JavaHelp.

Single Source Properties: Target

All Single Source projects have properties associated with them to define their characteristics and behavior. Use the **Target tab** to select the name and location for the Help project and to select the source files to include when generating the output.

Select output folder and output file name for WebHelp: Displays the default output folder and default output file name. If you want to select a different folder or file name, type it directly in the text box. You can also click the browse button to navigate to a different location on your hard drive.

Use short folder/file name (8.3 format): (This option available for WebHelp only.) Shortens names of folders and files to 8 characters. Use this feature if you are deploying an intranet or Web site on a Web server that does not support long file names. (Ask your webmaster if the server supports long files names, if you’re not sure.)

Choose source document(s): Selects the .DOC files to import. By default, all documents in the project are selected

Clear All: Clears all files from the selection.

Select All: Adds all files to the selection.

Build Expression: Click **Define** to select or add build tag expressions to Help projects.

Single Source Properties: Folders

All Single Source projects have properties associated with them to define their characteristics and behavior. Use the **Folders tab** to set up locations for saving HTML topic files and images. The settings you make at this dialog specify how your output source is organized (folders and subfolders).

Folder for HTML Files

- **Do not create subfolder:** Saves all HTML topic files in the root directory you’ve chosen for your output.
- **Create subfolder:** Saves all HTML topic files in a subfolder, based on name entered in text box.
- **Create subfolders based on source document names:** (Default) Creates and saves HTML topic files in subfolders, based on the names of the .DOC files.
- **Create subfolders based on TOC structure:** Creates and saves HTML topic files in subfolders, based on the titles of books in the WinHelp table of contents. Books within books are created as subfolders within folders and topics are arranged accordingly. All topics not used in the WinHelp table of contents are saved in the root directory you’ve chosen for your output.

Folders for Images

- **Do not create subfolder:** Saves all image files in the root directory.
- **Create subfolder:** (Default) Saves all image files in a subfolder, based on name entered in text box.

File Options

- **HTML File Extension:** Saves all HTML topic files using the extension selected from the drop-down. Choices include HTM (default) and HTML.
- **Always use lowercase:** Saves all file names in lowercase letters. This will help to eliminate problems if your Help files are being accessed on a UNIX server (UNIX is a case-sensitive platform).

Single Source Properties: Formatting

All Single Source projects have properties associated with them to define their characteristics and behavior. Use the **Formatting tab** to specify how you want to format the HTML topic files. You have the option of working with or without style sheets. If you don’t want to use style sheets, the HTML topics will be formatted using inline styles (which is like applying manual character and paragraph formatting in Word). If you want to use style sheets, you can create new style sheets based on the formatting in the WinHelp topics or you can select a style sheet you already use with other HTML topics.

Inline Styles (no linked style sheets): (Default for WebHelp) Converts all document formatting into HTML inline styles which are similar to character and paragraph formatting in Microsoft Word. This option does not involve creating or applying styles or style sheets to the topics. All formatting is handled on a paragraph-by-paragraph and/or character-by-character basis. You can apply styles to the topics after you import the project. If you want new topics you create to look like the ones you generate, you’ll need to manually apply character and paragraph attributes to the text.

Linked style sheets (one for each source document): (Default for Microsoft HTML Help) Creates one style sheet file (.CSS) for each .DOC file that you included as a source document. If the formats used in each .DOC file are unique, separate style sheets will incorporate these unique styles so you can continue to use them in the HTML Help project. For example, if your WinHelp project has two .DOC files, california.doc and arizona.doc, two style sheets are created, california.css and arizona.css. All topics in california.doc are linked to california.css in the HTML Help precut and all topics in arizona.doc are linked to arizona.css.

Linked style sheet based on one source document: Creates a single external style sheet (.CSS file) based on one of the .DOC files used in the conversion. For example, if the WinHelp project has two .DOC files (california.doc and arizona.doc) and you select california.doc, one style sheet is created (the style sheet will have the same name as the project, e.g., states.css). All HTML topics are linked to this style sheet and the formatting in all of the topics is based on the styles in states.css.

Linked style sheet based on existing style sheet: Links an existing style sheet to all HTML topics. For this feature to be effective, the styles in the .DOC files need to have a one-on-one relationship with the styles in the style sheet*. This means that styles in both the documents and the style sheet should use the exact same names. The styles should also be applied to the appropriate paragraphs and characters that need to use them. If not, the style sheet will provide the styles that you can apply to the text, but the text will be formatted manually (since style names did not match the ones in the existing style sheet).

Minimum Font Size: (WebHelp only) This option resolves problems caused by some browsers and platforms that do not display small fonts well. You can experiment to see which fonts are the most readable on your end users’ systems and set the font accordingly. (The default minimum font setting is 9 point.)

* The two most common Microsoft Word styles, **Normal** and **Heading 1**, are automatically mapped to the **<P>** and **<H1>** HTML styles, respectively.

Single Source Properties: HTML Help Features

All Single Source projects have properties associated with them to define their characteristics and behavior. Use the **Features tab** to override the default features. We have carefully selected the defaults for you. You only need to change these settings if you want to customize them.

Popups: Converts WinHelp popups into Smart Popups or normal HTML hyperlinks.

- **Smart Popups*:** Creates links that display destination topics in popup windows that size to content. These popup windows support all HTML formatting.
- **Regular hyperlinks:** Creates links that display destination topics in the same window where topic containing link is displayed.

What's This Help: Specifies how to convert context-sensitive WinHelp topics.

- **Text-only Topic files*:** Creates and saves topics in text-only format (.TXT). Use this option for What's This? Help topics.
- **Individual HTML files:** Creates and saves topics as regular HTML topic files (to use as window-level context-sensitive Help).
- **Remove Topics:** Removes topics from the single source conversion.

External Topics: Specifies how to convert external WinHelp topic links.

- **Link to external HTML Help topic*:** Converts external WinHelp topic links to external HTML Help topic links (links to HTML files in external .CHM files). Use this option if you plan on merging multiple WinHelp projects into one HTML Help system.
- **Retain:** Keeps external WinHelp topic links. Use this option if you plan on linking HTML topics to WinHelp topics. The HTML topics will include WinHelp Topic controls (special ActiveX controls that open WinHelp .HLP files).
- **Remove:** Removes links to external WinHelp topics.

Images: Conversion option for .BMP images.

- **GIF*:** A compressed bitmap format.
- **JPEG:** A compressed true-color or high-color bitmap format.

Bullets / Numbering: Specifies how to convert WinHelp bulleted and numbered lists.

- **HTML bullets and numbering*:** Converts WinHelp bulleted and numbered lists into HTML bulleted and numbered lists. These lists are auto-numbered or bulleted.
- **Formatted Text:** Turns WinHelp bulleted and numbered lists into text paragraphs. Numbers are hard-coded as text. A symbol character is used to represent the bullet.

* = recommended: All options that include an * indicate the recommended setting.

Single Source Properties: WebHelp Features

All Single Source projects have properties associated with them to define their characteristics and behavior. Use the **Features tab** to override the default features. We have carefully selected the defaults for you. You only need to change these settings if you want to customize the settings.

Popups: Converts WinHelp popups into Smart Popups or normal HTML hyperlinks.

- **Smart Popups*:** Creates links that display destination topics in popup windows that size to content. These popup windows support all HTML formatting.
- **Regular hyperlinks:** Creates links that display destination topics in the same window where topic containing link is displayed.

What's This Help: Specifies how to convert context-sensitive WinHelp topics.

- **Individual HTML files*:** Creates and saves topics as regular HTML topic files (to use as window-level context-sensitive Help).
- **Remove topics:** Removes topics from the single source conversion.

External Topics: Specifies how to convert external WinHelp topic links.

- **Link to external WebHelp topic*:** Converts external WinHelp topic links to external HTML topic links (links to HTML files in external folders). Use this option if you plan on merging multiple WinHelp projects into one WebHelp system.
- **Retain:** Keeps external WinHelp topic links. Use this option if you plan on linking HTML topics to WinHelp topics. The HTML topics will include WinHelp Topic controls (special ActiveX controls that open WinHelp .HLP files).
- **Remove:** Removes links to external WinHelp topics.

Images: Conversion option for .BMP images.

- **GIF*:** A compressed bitmap format.
- **JPEG:** A compressed true-color or high-color bitmap format.

Bullets / Numbering: Specifies how to convert WinHelp bulleted and numbered lists.

- **HTML bullets and numbering*:** Converts WinHelp bulleted and numbered lists into HTML bulleted and numbered lists. These lists are auto-numbered or bulleted.
- **Formatted Text:** Turns WinHelp bulleted and numbered lists into text paragraphs. Numbers are hard-coded as text. A symbol character is used to represent the bullet.

Single Source Properties: JavaHelp Features

All Single Source projects have properties associated with them to define their characteristics and behavior. Use the **Features tab** to override the default features. We have carefully selected the defaults for you. You only need to change these settings if you want to customize the settings.

Popups: Converts WinHelp popups into Smart Popups or normal HTML hyperlinks.

- **JavaHelp Popups*:** Creates JavaHelp popups that open in its own popup window.
- **Regular Hyperlinks:** Creates links that display destination topics in the same window where topic containing link is displayed.

What's This Help: Specifies how to convert context-sensitive WinHelp topics.

- **Individual HTML files:** Creates and saves topics as regular HTML topic files (to use as window-level context-sensitive Help).
- **Remove topics:** Removes topics from the single source conversion.

External Topics: Specifies how to convert external WinHelp topic links.

- **Link to external JavaHelp topic*:** Converts external WinHelp topic links to external HTML topic links (links to HTML files in external folders). Use this option if you plan on merging multiple WinHelp projects into one JavaHelp system.
- **Retain:** Keeps external WinHelp topic links. Use this option if you plan on linking HTML topics to WinHelp topics. The HTML topics will include WinHelp Topic controls (special ActiveX controls that open WinHelp .HLP files).
- **Remove:** Removes links to external WinHelp topics.

Images: Conversion option for .BMP images.

- **GIF*:** A compressed bitmap format.
- **JPEG:** A compressed true-color or high-color bitmap format.

Bullets / Numbering: Specifies how to convert WinHelp bulleted and numbered lists.

- **HTML bullets and numbering*:** Converts WinHelp bulleted and numbered lists into HTML bulleted and numbered lists. These lists are auto-numbered or bulleted.
- **Formatted Text:** Turns WinHelp bulleted and numbered lists into text paragraphs. Numbers are hard-coded as text. A symbol character is used to represent the bullet.

* = recommended: All options that include an * indicate the recommended setting.

Single Source Properties: HTML Help Options

All Single Source projects have properties associated with them to define their characteristics and behavior. Use the **Microsoft HTML Help tab** to specify which features you want to appear on the HTML Help navigation (left-hand) pane and access advanced options in this dialog.

Navigation Pane

- To include TOC and Index tabs, select **TOC & Index**.
- To include a Favorites tab, allowing you to annotate (or bookmark) topics by adding them to the Favorites tab, select **Favorites**.
- To include a History tab, showing the topics you’ve navigated to, select **History**.
- In **Search**, select **Regular** if you want to provide simple searches based on all or part of a topic’s title. If you don’t want the Search tab, select **No Search**. An **Advanced** search provides a more thorough search based on topic properties.
- In **Tab Position**, select where the TOC, Index, and Search tabs should be displayed in the HTML Help viewer or browser (**Top**, **Left**, or **Bottom**).
- In **Default Tab**, select the tab that should be displayed when the browser or HTML viewer is opened.

Options

- Select **WebSearch** to enable the WebSearch button on your main button bar. WebSearch allows end users to perform Internet searches about the Help topics and application they’re using right from the compiled Help window. (Users need Internet access to use the WebSearch button.)
- **Create Browse Sequences** enables you to keep the browse sequences you’ve set in your Help project.
- **Ignore Secondary windows** is selected by default. HTML Help doesn’t support secondary windows, so they are automatically converted to popup windows.
- Select **Create Compiled CHM only** if you want to output a compiled HTML Help file and not the .HHP file and the individual HTML files. The default is to generate the .CHM along with the project and HTML files.
- Select where to add index keywords in **Add Keywords to**. If you add them to **Each Topic** (default), you can use the HTML files in other projects. If you add them to the **Index File (HHK)**, they are included in a separate file in the same directory as the HTML Help project you are creating.
- Click Advanced to customize options for your tri-pane window, TOC, and custom buttons.

Single Source Properties: WebHelp Options

All Single Source projects have properties associated with them to define their characteristics and behavior. Use the **WebHelp tab** to set up your navigation pane tabs and other options in this dialog.

Navigation Pane

The navigation pane is the left-hand side of the tri-pane window.

- Select the tabs you want displayed in your WebHelp project. You can display the **TOC** (Contents), the **Index**, and the **Search** tabs. Clear the tabs you don’t want to appear.
- In **Preferred Format**, select **Dynamic HTML** or **Java Applet**, depending on the user interface and functionality you prefer.

In Topic Navigation Bar

The topic navigation bar allows you to place browse sequence navigation and WebSearch in every topic.

- Select **Browse Sequences** to keep the browse sequences you’ve set up in your Help project. (These buttons follow the same browse sequence established in the original WinHelp file. They are not related to the browser’s Back and Next buttons.)
- Select **WebSearch** to enable WebSearch functionality. WebSearch allows end users to perform Internet searches about the Help topics and application they’re using right from the compiled Help window. (Users need Internet access to use the WebSearch button.)
- Select the **Location** of your browse sequence arrows and the WebSearch button. You can choose to display these options on the **Top Left**, **Top Right**, **Bottom Left**, or **Bottom Right** of each topic.
- Select the **Style** of the browse sequence and WebSearch. **Text** creates hotspot links for the browse arrows and hotspot text for WebSearch (“WebSearch”), and **Button** creates buttons for these options with text labels.

Prefix HTML files with Help project name

- Select this option if you want each HTML file generated to begin with the Help project name.

Single Source Properties: JavaHelp Options

All Single Source projects have properties associated with them to define their characteristics and behavior. Use the **JavaHelp tab** to specify which features you want to appear on the JavaHelp navigation pane and whether your file names are prefixed with the JavaHelp project name.

Navigation Pane

The navigation pane is the left-hand side of the tri-pane window.

- Select the panes you want displayed in your JavaHelp project. You can display the **TOC** (Contents), the **Index**, and the **Full-Text Search** panes. Clear the panes you don’t want to appear.

In Topic Navigation Bar

The topic navigation bar allows you to place browse sequence navigation and the WebSearch button in every topic.

- Select **Browse Sequences** to keep the browse sequences you’ve set up in your Help project. (These buttons follow the same browse sequence established in the original WinHelp file. They are not related to the browser’s Back and Next buttons.)
- Select **WebSearch** to enable WebSearch functionality. WebSearch allows end users to perform Internet searches about the Help topics and application they’re using right from the compiled Help window. (Users need Internet access to use the WebSearch button.).
- Select the **Location** of your browse sequence arrows and WebSearch. You can choose to display these options on the **Top Left**, **Top Right**, **Bottom Left**, or **Bottom Left** of each topic.
- Select the **Style** of the browse sequence and the WebSearch button. **Text** creates hotspot links for the browse arrows and hotspot text for WebSearch (“WebSearch”), and **Button** creates buttons for these options with text labels.

Output

You can select the type of JavaHelp output you want to generate.

- If you want to create a .JAR file and .HS file, select **Compressed JavaHelp**.
- If you want to create a .JAR file, .HS file, and individual source files, select **Compressed JavaHelp with source files**.
- If you want to create an .HS file and individual source files, select **Uncompressed JavaHelp with source files**.

Prefix HTML files with Help project name

- Select this option if you want each HTML file generated to begin with the Help project name. Use this option if you plan on merging multiple WinHelp projects into one JavaHelp system.

Build tags

Build tags enable you to design and create several different WinHelp projects from a single set of source files. They give you control over which topics are included in a file. You create the build tags, then mark the topics you want to use with them.

Example build tag

For this example, assume you have three different topics tagged with these build tags:

Topic 1 – tag1

Topic 2 – tag2

Topic 3 – tag3

To include:	Do this:
All topics assigned to a single build tag	Select the build tag name. (Tag1)
All topics assigned to multiple build tags	Use AND with the build tag names. (Tag2 and Tag3)
All topics assigned to any combo of named build tags	Use OR with the build tag names. (Tag 3 OR Tag4)
All topics except those assigned to a specific build tag	Use NOT with the build tag to exclude. (NOT Tag1)
All topics except those assigned to multiple build tags	Use NOT in combo with AND. (Tag1 AND NOT Tag2 AND NOT Tag3)
All topics except those assigned to any combination of named build tags	Use NOT in combo with AND and OR. (NOT Tag1 AND Tag3 OR Tag5 OR Tag7)

Smart Publishing Wizard: Destination

Use the Smart Publishing Wizard-Select Destination dialog to specify a destination for files and folders you want to publish to a location such as a corporate intranet, Internet site, local hard drive, or network.

- **Server Name/URL list.** Displays server names you have already manually added as locations for publishing files and folders. To sort the list, click on Server Name or URL.
- **New.** Displays the New Destination dialog, which enables you to specify a new server location for publishing files or folders.
- **Edit.** Displays the Edit Destination dialog, which enables you to edit the name and path of locations listed in the Server Name/URL list.
- **Delete.** Removes the selected location from the Server Name/URL list.
- **Options:**

  Check for deleted files. Select this option if you want the Smart Publishing Wizard to check for files that have been deleted from the destination location. (By default, the Smart Publishing does not perform this check because it slows down the publishing process.)

  Prompt before overwriting files. Select this option if you want the Smart Publishing Wizard to prompt you before overwriting any files.

  Republish all. Select this option if you want the Smart Publishing Wizard to republish ALL files to the destination target, whether they are newer than existing files or not. (The Smart Publishing Wizard determines which files to update by comparing dates to look for the most recent file and by comparing files to check if they are different.)

Smart Publishing Wizard dialog

Use the Smart Publishing Wizard-Select Source dialog to select files and folders to publish to a specific location such as a corporate intranet, Internet site, hard drive, or network.

Note:

You may need to work with your network system administrator to determine the best method of publishing your files and to obtain access permissions.

- **Source Location.** Specifies the location of the source files you want to publish. Type in a path or click Select file/Folder to browse to a location.
- **Publish Subfolders.** Select this option if you want to publish all subfolders within the file or folder selected in Source Location. Clear this option if you do not want to publish all subfolders.

Smart Publishing Wizard: Edit destination

Use the Edit Destination dialog to edit a location you have already added to the Server Name/URL list in the Smart Publishing Wizard. You may need to work with your network system administrator to determine the best method of publishing your files and to obtain the necessary access permissions.

Note:

You may need to work with your network system administrator to determine the best method of publishing your files and to obtain access permissions.

- **Descriptive name.** Specifies the name of the Server/URL you want to add to the list. Each name must be unique.
- **Connection Protocols.** Specifies the type of transfer protocol to use in order to reach the new destination.

  FTP. Select this option if you connect to the new destination through FTP.

  FrontPage Enabled. Select this option if you have Microsoft FrontPage installed on your system and would like to publish files using FrontPage as your interface.

  HTTP. Select this option if you connect to the new destination through HTTP.

  File System. Select this option if you want to publish your files to a local or network drive.

If you selected FTP as your connection protocol, you have these options:

- **Host Name.** Specifies the name of the server on which you want to publish your files. You can also use an IP address.
- **User ID.** Specifies the user ID that the Smart Publishing Wizard should use to connect to the chosen host. (Your network system administrator can provide you with a user ID.)
- **Password.** Specifies the password for the user ID. (Your network system administrator can provide you with a password.)
- **Save Password.** Saves the password so you do not need to enter it each time you publish your files. (Please note that saving the password enters the password into your registry. The password is not encrypted, so any user with access to the registry can view or modify it.)
- **Port.** Specifies the exact port to use when connecting to the host. (Your network system administrator can provide you with the correct port number to use.)
- **Anonymous User.** Select this option to publish your files as an anonymous user that does not require a password. If you need to supply a password or other information associated with being an "anonymous user" (e.g., an email address), leave this check box empty, enter "Anonymous" in the User ID field above, and add the other required information in the Password field above. (Your network system administrator can tell you if publishing files as an anonymous user is permitted and what process to use.)
- **Server Directory.** Specifies the destination directory on the server. You can leave the forward slash to accept the default.

If you selected FrontPage Enabled as your connection protocol, you have these options:

- **Host Name.** Specifies the name of the server on which you want to publish your files.
- **Web Name.** Specifies the name of an individual FrontPage Web on the server you selected in Host Name. (No initial slash is required for subwebs.)
- **Server Directory.** Specifies the destination directory on the server. You can leave the forward slash to accept the default.

If you selected HTTP as your connection protocol, you have these options:

- **Host name.** Specifies the name of the server on which you want to publish your files.
- **Server Directory.** Specifies the destination directory on the server. You can leave the forward slash to accept the default.

If you selected File System as your connection protocol, you have this option:

- **Destination Path.** Specifies the exact destination path to the location where you want to publish your files. Either type in the path or click the browse button and navigate to a location.

Smart Publishing Wizard: New destination

Use the New Destination dialog to add a location to the Smart Publishing Wizard. This enables you to select the destination from the Smart Publishing Wizard Server Name/URL list instead of having to type in the path every time you want to publish your files or folders.

Note:

You may need to work with your network system administrator to determine the best method of publishing your files and to obtain access permissions.

- **Descriptive name.** Specifies the name of the Server/URL you want to add to the list. Each name must be unique.
- **Connection Protocols.** Specifies the type of transfer protocol to use in order to reach the new destination.

  FTP. Select this option if you connect to the new destination through FTP.

  FrontPage Enabled. Select this option if you have Microsoft FrontPage installed on your system and would like to publish files using FrontPage as your interface.

  HTTP. Select this option if you connect to the new destination through HTTP.

  File System. Select this option if you want to publish your files to a local or network drive.

If you selected FTP as your connection protocol, you have these options:

- **Host Name.** Specifies the name of the server on which you want to publish your files. You can also use an IP address.
- **User ID.** Specifies the user ID that the Smart Publishing Wizard should use to connect to the chosen host. (Your network system administrator can provide you with a user ID.)
- **Password.** Specifies the password for the user ID. (Your network system administrator can provide you with a password.)
- **Save Password.** Saves the password so you do not need to enter it each time you publish your files. (Please note that saving the password enters the password into your registry. The password is not encrypted, so any user with access to the registry can view or modify it.)
- **Port.** Specifies the exact port to use when connecting to the host. (Your network system administrator can provide you with the correct port number to use.)
- **Anonymous User.** Select this option to publish your files as an anonymous user that does not require a password. If you need to supply a password or other information associated with being an "anonymous user" (e.g., an email address), leave this check box empty, enter "Anonymous" in the User ID field above, and add the other required information in the Password field above. (Your network system administrator can tell you if publishing files as an anonymous user is permitted and what process to use.)
- **Server Directory.** Specifies the destination directory on the server. You can leave the forward slash to accept the default.

If you selected FrontPage Enabled as your connection protocol, you have these options:

- **Host Name.** Specifies the name of the server on which you want to publish your files.
- **Web Name.** Specifies the name of an individual FrontPage Web on the server you selected in Host Name. (No initial slash is required for subwebs.)
- **Server Directory.** Specifies the destination directory on the server. You can leave the forward slash to accept the default.

If you selected HTTP as your connection protocol, you have these options:

- **Host name.** Specifies the name of the server on which you want to publish your files.
- **Server Directory.** Specifies the destination directory on the server. You can leave the forward slash to accept the default.

If you selected File System as your connection protocol, you have this option:

- **Destination Path.** Specifies the exact destination path to the location where you want to publish your files. Either type in the path or click the browse button and navigate to a location.

Smart Publishing Wizard: Publish

Use the Publish dialog to ensure that the files you specified are published to the correct location. The Wizard identifies the path to the files you are publishing and the location to which you want them published. If any information is incorrect, click Back and edit the source location or destination path.

Smart Publishing Wizard: Results

The Results dialog displays after you publish files or folders using the Smart Publishing Wizard.

- **Statistics:**

  Total Files. Indicates the number of files or folders you specified to publish.

  Files Published. Indicates the total number of files actually published.

  Elapsed Time. Displays the amount of time it took to publish the files or folders.
- **Files Published.** Displays the path and name of each file or folder published.
- **Print.** Prints the list of files published.

Generating WebHelp

RoboHELP Classic provides an automated feature that generates source files and saves them as HTML files in your project folder within a subfolder named “WebHelp”. You distribute these source files, plus your project-related files (*.HTM, *.GIF, etc.) to your users. You can use WebHelp output files for application Help, standalone projects, or online documents.

To generate WebHelp:

1 Open the project.

2 From the **File** menu, select **Generate - WebHelp**. The Single Source wizard opens. You'll be prompted through a series of screens that offer different options for specifying how to set up the WebHelp output.

3 Select the project folder and start page for the WebHelp output. Click 📂, navigate to the drive/folder where the file is located, and double-click to select it. **Note:** The start page uses the .HTM extension. Click **Next** to proceed through the Wizard screens.

4 Click **Finish** when you get to the end of the Wizard prompts.

Note: If the HTML files use spaces in the file names, the WebHelp copies that are saved in the output folder replace the spaces with underscores to make them compatible with Netscape Navigator.

Viewing WebHelp

After you've generated WebHelp output, you can view it. RoboHELP automatically uses your default browser to view WebHelp output.

To view WebHelp output:

1 Open the project you want to view.

2 Open the **Single Source** folder and right-click on **WebHelp**.

3 Select **View**.

The WebHelp output opens in your default browser in a two-pane window:

- The left pane contains the Contents, Index and Search tabs. For Netscape Navigator, it also displays a Help tab.
- The right pane displays the first topic (default topic).

About WebHelp projects

WebHelp is generated as an output format. You create your project in RoboHELP Classic, adding all of the features that are supported by WebHelp such as tables of contents, indexes and full-text search. (Check WebHelp browser limitations for features to avoid including in your project if you know it will be viewed on Netscape Navigator browsers.)

At the end of your project when you are ready to test or distribute the files, you generate the WebHelp files from a File menu option. All of the files you need to distribute are copied into a single folder within your project folder. A completed WebHelp project is cross-browser and platform independent — the project runs well on any combination of browser and platform.

Style sheets for WebHelp

When you create WebHelp from RoboHELP using styles, the styles will translate when you generate WebHelp. For example, you can choose to have your styles output into a linked style sheet or smart inline styles. The WebHelp Wizard guides you through the style choices you can select.

Editing WebHelp

You can use RoboHELP to edit your Help project source files and then generate WebHelp again.

To edit WebHelp:

1 Open RoboHELP Classic and the Help project you used to create WebHelp output.

2 Make your changes.

3 Generate WebHelp again.

Selecting properties for WebHelp

All Single Source projects have properties associated with them to define their characteristics and behavior. You can change the following options in the WebHelp Properties dialog:

- **Target.** Select the name and location for the Help project and to select the source documents to include when importing the project. Access the Define Build Expression dialog to add build tags.
- **Folders.** Set up locations for saving HTML topic files and images. The settings you make in this dialog specify how your output source is organized (folders and subfolders).
- **Formatting.** Specify how you want to format the HTML topic files. You have the option of working with or without style sheets. Specify the minimum font to be displayed.
- **Features.** Override the default features for popups, What's This? Help, linking to external topics, images, and bullets/numbering. We have carefully selected the defaults for you, so you only need to change these settings if you want to customize them.
- **WebHelp.** Specify which features you want to appear on the WebHelp navigation (left-hand) pane, specify more conversion selections, and access advanced options in this dialog.

Use this procedure to change any of the default settings for Help format properties.

To select properties for WebHelp:

1. From the RoboHELP Explorer Project tab, open the **Single Source** folder.
2. Right-click on **WebHelp** and select **Properties**.
3. Click the tab you want and make changes.
4. Click **OK** when you're done, and generate WebHelp.

WebHelp overview

eHelp Corporation's WebHelp technology makes it possible to deliver fully functional HTML Help and Web content across platforms and browsers. These easy-to-create, reliable Help formats are widely used by Help authors and Web content developers around the world.

RoboHELP Office provides WebHelp functionality as Help format. Once you're finish with a project, simply select WebHelp as the Help format and generate your files.

WebHelp is an uncompiled format that supports standard Help features including a table of contents, index and full-text search, as well as context-sensitive Help.

WebHelp is flexible and reliable. It provides a solution for authors who design WinHelp, HTML Help and intranet/Internet content that needs to run with a variety of browsers (e.g., Netscape Navigator) and on a variety of platforms (e.g., UNIX, Macintosh, Windows 3.1). Any combination of platform and browser can access the same information because WebHelp detects the required files to launch at run-time.

Who needs to use it?

WebHelp is designed for authors who must create a Help project, online document or a Web site that may be accessed from a variety of platforms and browsers. The output files created when you generate WebHelp are cross-platform and browser-independent.

How WebHelp works

Behind the scenes, WebHelp is a set of files used to run your online information. You generate the WebHelp files after authoring topics, a table of contents, and an index. All of the files you need to distribute are copied into a single folder that you name.

When users start your WebHelp project, the key components auto-detect the end-users' system (platform and browser) and launch the appropriate files for that particular system. This automatic process assures that your WebHelp project will look exactly the way you designed it, regardless of browser and platforms.

What's new in WebHelp

The latest version of WebHelp includes new features that give you a more sophisticated level of control over your projects. Also, added navigation features make your projects easier for end users to navigate.

WebHelp gives you cross-platform, browser-independent HTML-based Help. Virtually any browser can be used to access WebHelp. Your Help projects can be accessed from a variety of platforms including UNIX, Linux, Macintosh, and Windows 3.1, 95, 98, NT 4.0, and 2000. WebHelp is designed for application Help projects, standalone projects, online documentation, and intranet/Internet Web pages.

WebHelp has been thoroughly tested with Internet Explorer and Netscape Navigator on a wide variety of platforms, ensuring that WebHelp projects run seamlessly on all end-user configurations.

Here are some of the new features in WebHelp:

- **Customizable in-topic navigation bar.** The in-topic navigation bar in WebHelp is now completely customizable. Pick and choose from different elements such as a hide/show navigation pane button, browse sequences buttons, or a WebSearch button. You can further customize each element by assigning your own image or text.
- **Sync TOC.** Another customizable element you can add to the in-topic navigation bar is the Sync TOC (synchronize table of contents) button. As users navigate to various topics through links or jumps in the right-hand (topic) pane, the table of contents no longer reflects the current topic. Pressing the Sync TOC button fixes this problem by synchronizing the table of contents with the topic currently displayed in the topic pane.
- **Compressed table of contents, index, and full-text search.** Behind the scenes, WebHelp now compresses your table of contents, index, and full-text search files. Compression can reduce the size the size of these files a dramatic 65-95%. This in turn reduces the size of your WebHelp project and makes distribution easier. The smaller size also decreases the amount of time that is required to download these files to your users, which can be significant for large projects.
- **JavaScript.** RoboHELP uses JavaScript to manage compatibility issues related to the various browser and operating system differences that occur in your end users' environment. In previous versions of WebHelp, RoboHELP embedded this JavaScript into each topic page. This JavaScript addition increased the size of each individual file. To alleviate this problem and remove the extra code, JavaScript used for this purpose is now referenced in an external JavaScript file. Any custom scripts you add, and any added by RoboHTML for such things as DHTML, will continue to be embedded into the topic page itself.
- **Smart Publishing Wizard.** After you generate WebHelp files, you have the option to copy or "publish" the files. The Smart Publishing Wizard enables you to easily publish files or folders to different locations such as a corporate intranet, Internet site, or even a local or network drive. The publishing wizard is "smart" because it only publishes files that are newer and different than the last time the files were published.

Different uses for WebHelp

WebHelp is ideal for projects that need to run on a wide variety of platforms using different browsers including:

- Application Help
- Standalone Help
- E-commerce application Help
- Online books
- User guides
- Employee handbooks
- Policies and procedures manuals
- Operations manuals
- Corporate reports
- Sales catalogs
- Proposals
- Training guides
- Project plans
- Technical support manuals
- Human resources information
- Client/customer information
- Corporate intranet web pages
- Internet Web pages
- Extranet Web pages

How WebHelp differs from HTML Help

Microsoft HTML Help is a compiled Help format intended to run on Windows platforms. It relies on the Microsoft HTML Help engine to support Help-specific features such as keywords, tables of contents, indexes, HTML Help controls, link controls, and full-text search. It relies on components of Internet Explorer to display the compiled Help system from a special HTML Help viewer.

You create Microsoft HTML Help by compiling the project files into a single HTML Help file (.CHM). It is only supported using 32-bit Windows platforms with Internet Explorer installed.

WebHelp is based on HTML, but runs with a variety of browsers (e.g., Netscape Navigator) and on a variety of platforms (e.g., Macintosh, UNIX, Windows 3.1). Depending on the configuration, WebHelp employs Dynamic HTML, Java applet, or HTML Lists to provide functionality. (WebHelp displays HTML Lists for older browsers that lack full Java or Dynamic HTML support.)

Unlike Microsoft's HTML Help, WebHelp is not a compiled format. WebHelp supports all primary HTML Help features, including a table of contents, multi-level index, related topics, and full-text search.

The following information summarizes the key differences between HTML Help and WebHelp:

HTML Help	**WebHelp**
Relies on Microsoft HTML Help engine for support	Supported by Web browser
Compiled and compressed (.CHM file)	Uncompiled set of output files
Requires 32-bit Windows platforms	Runs on any platform including Windows, UNIX, Linux, Sun Solaris and Macintosh
Requires Internet Explorer	Browser-independent

WebHelp and browsers/platforms

While WebHelp has been tested on a wide variety of browsers (Internet Explorer 2.0 and later, Netscape Navigator 2.0 and later, Opera, HotJava, and WebTV), we recommend always using the latest version. The following information summarizes recommended platforms and browsers for WebHelp.

Recommended platforms and browsers for WebHelp

Platform:	**Browser:**
Windows 2000	Internet Explorer 4.0 or later (Windows 2000 ships with IE 5.x)
	Netscape Navigator 3.01 or later
Windows 98	Internet Explorer 4.0 or later (Windows 98 ships with IE 4.x)
	Netscape Navigator 3.01 or later
Windows 95	Internet Explorer 3.0 or later
	Netscape Navigator 3.01 or later
Windows NT 4	Internet Explorer 3.0 or later
	Netscape Navigator: 3.01 or later
Windows 3.1*	Internet Explorer 4.0 or later
	Netscape Navigator 3.04 or later
Macintosh	Internet Explorer 4.0 or later
	Netscape Navigator 4.0 or later
UNIX platforms	Internet Explorer 4 or later
	Netscape Navigator 4.0 or later

* Use short (8 characters) file names for all of your project files because Windows 3.1 does not support long file names.

Note: WebHelp displays HTML Lists for older browsers that lack robust Java or Dynamic HTML support.

Files Generated for WebHelp

When you use WebHelp, the source files are generated in a subfolder that you name. This folder is saved in your project folder. The WebHelp source files that are copied into this folder include:

File:	**Description:**
WEBHELP.CAB	WebHelp applet in Internet Explorer-compatible cabinet file
WEBHELP.ZIP	WebHelp applet in Netscape Navigator-compatible uncompressed ZIP file
WEBHELP3.JS	WebHelp JavaScript file that supports the Dynamic HTML contents, index, and full-text search
BSSCDHTM.JS	WebHelp JavaScript file that supports Dynamic HTML related topics popups, Smart popups, and other effects
BSSCDHT1.JS	Split JavaScript file to support Windows 3.1 file size limitation
BSSCDHT2.JS	Split JavaScript file to support Windows 3.1 file size limitation
TABFRAME.HTM	Topic for Dynamic HTML display of Contents, Index, and Search images
NAVFRAME.HTM	Navigation frame (contents, index, search)
NAVPANE1.HTM	Frameset for Internet Explorer Dynamic HTML navigation
NAVPANE2.HTM	Frameset for Java applet navigation
TOCDHTML.HTM	Dynamic HTML layout of TOC (books and pages)
TOCLIST.HTM	HTML List layout of TOC (books and pages)
IDXBODY.HTM	Index content JavaScript file (contains index keyword data)
IDXDHTML.HTM	Index content frameset
IDXFORM.HTM	Index content form loading file
IDXLIST.HTM	HTML List layout of Index
FTSBODY.HTM	Full-text search JavaScript file (contains search data)
FTSDHTML.HTM	Full-text search frameset
FTSFORM.HTM	Full-text search form loading file
FILENAME.HHC	Site map file that contains TOC
FILENAME.HHK	Site map file that contains Index
FILENAME.HHS	Internal use file for stop keywords, keywords, and number of topics
FILENAME.FTP	List of files needed for distribution
Associated .GIF files	Navigation and Browse Sequence images

Distributing WebHelp

You can use WebHelp to create cross-platform Help for applications as well as standalone Help for online books and other electronic documents that are not shipped with software programs. If your Help system is distributed with an application, your developer will need the files so they can be included in the program's installation. For standalone Help, you can distribute the files on disk or CD.

When you are ready to generate WebHelp output, you specify the name of the folder where your WebHelp source files will be located as well as the name of the start page. The start page is the first topic that is displayed in the right-hand pane of the browser when your Help system opens.

- All the required WebHelp source files are generated and saved in a subfolder in your project folder titled WebHelp (or whatever name you assign).
- If your project includes subfolders, the WebHelp folder will include these subfolders with the appropriate files located in them. Do not rename or move any of these folders (leave the project structure intact).

To distribute WebHelp with an application:

- If you are distributing WebHelp with an application, copy the WebHelp folder and its contents to the location designated by your developer. Be sure to let your developer know the name of the start page (.HTM file).

To distribute WebHelp on disk or CD:

1 If you are distributing WebHelp as a standalone product, copy the WebHelp folder and its contents to a disk or CD.

2 You can copy these files to any location on your end-users' systems.

3 Set up WebHelp so users select the start page (.HTM file) when they want to run your Help system.

Tips:

- Be sure to view and test your WebHelp project on your authoring system before it's posted. This way, you can make sure it includes all the features you want to use with your project and you can check your links, review topics, etc. so they suit your needs.
- You can also use WebHelp to create projects for your company's intranet and for Web pages. For distribution information, see Posting WebHelp on a Web server.

Programming context-sensitive Help for WebHelp

Note: The information in this topic is intended for developers who need to create context-sensitive WebHelp topics.

Launching a context-sensitive WebHelp topic from a Web page is similar to launching an HTML file in a secondary window.

The example below shows one way to declare a JavaScript function that will display a secondary window. The ShowHelp function shows the specified topic in a secondary window with the attributes defined by the strHelpOptions variable.

```
<SCRIPT LANGUAGE="JavaScript">
var strHelpOptions = "location=no";
strHelpOptions += ",toolbar=no";
strHelpOptions += ",menubar=yes";
strHelpOptions += ",status=yes";
strHelpOptions += ",scrollbars=yes";
strHelpOptions += ",resizable=yes";
strHelpOptions += ",top=0";
strHelpOptions += ",left=0";
strHelpOptions += ",width=400";
strHelpOptions += ",height=400";
function ShowHelp(strUrl)
{
window.open(strUrl, "Help", strHelpOptions);
}
</SCRIPT>
```

Using ShowHelp

To use ShowHelp, place the above script between the </HEAD> and <BODY> tags in the HTML file. Include an anchor tag like the following where you want the user to get help:

```
<A Href='javascript:ShowHelp("widget.htm")'>Help!</A>
```

This tag makes a hyperlink with the text "Help!" When a user clicks the hyperlink, "widget.htm" will be launched in a secondary browser window.

Context-sensitive WebHelp for Java applets

Note: The information in this topic is intended for developers who need to connect context-sensitive WebHelp topics to Java applets.

Launching a context-sensitive WebHelp topic from a Java applet is similar to launching an HTML file in a secondary window.

In the example below, ShowHelp is a member of the derived Applet class. It displays the specified topic in a secondary browser window:

```
public void ShowHelp(String strTopic)
{
URL url = new URL(getDocumentBase(), "help/" + strTopic);
getAppletContext().showDocument(url, "secondary");
}
```

Context-sensitive WebHelp for Visual Basic

Note: The information in this topic is intended for developers who need to connect context-sensitive WebHelp topics to Visual Basic applications.

The sample code shown below uses a function called ShowHelp that will launch a local or remote WebHelp topic regardless of the browser type.

ShowHelp takes two arguments. The first, szTopic, is a URL or local filename. The second argument, bIsLocal, is a boolean value that indicates whether the file is a local filename (TRUE) or a remote URL (FALSE). If the file is local, ShowHelp tells the browser to find the file in the "Help" subfolder of the folder containing the application program.

```
Public Function ShowHelp(strTopic As String, _
bIsLocal As Boolean) As Boolean
Dim strDir As String
If bIsLocal Then
' Get registry entry pointing to Help
strDir = App.Path + "\Help\"
End If
' Launch topic
Dim hinst As Long
hinst = ShellExecute(Me.hwnd, vbNullString, _
strTopic, vbNullString, _
strDir, SW_SHOWNORMAL)
' Handle less than 32 indicates failure
ShowHelp = hinst > 32
End Function
```

To call ShellExecute, declare the function like this:

```
Private Declare Function ShellExecute Lib _
"shell32.dll" Alias "ShellExecuteA" _
(ByVal hwnd As Long, _
ByVal lpOperation As String, _
ByVal lpFile As String, _
ByVal lpParameters As String, _
ByVal lpDirectory As String, _
ByVal nShowCmd As Long) As Long
```

ShowHelp requires an explicit filename or URL. But for context-sensitive Help, it is better to code the application to use context IDs rather than explicit topic names. Use the following sample function, ShowHelp Context, which maps integer context IDs to topic name strings, then calls ShowHelp to launch the topic.

```
Public Function ShowHelpContext(nContextId As Integer) As Boolean
Dim strTopic As String
Dim bIsLocal As Boolean
bIsLocal = True
Select Case nContextId
Case HH_GADGET_DIALOG
strTopic = "gadget.htm"
Case HH_WHATSIT_DIALOG
strTopic = "whatsit.htm"
Case HH_WIDGET_DIALOG
strTopic = "widget.htm"
Case HH_TECH_SUPPORT:
strTopic = "http://www.mycompany.com"
```

```
bIsLocal = False
Case Else
strTopic = "unknown-context.htm"
End Select
ShowHelpContext = ShowHelp(strTopic, bIsLocal)
End Function
```

Using ShowHelpContext makes maintaining context-sensitive Help much easier because if a topic name changes, only one function has to be modified. Note that the context IDs (for example, HH_GADGET_DIALOG) should be declared as constants that can be shared among the various program modules that use context-sensitive WebHelp.

```
Const HH_GADGET_DIALOG As Integer = 1
Const HH_WHATSIT_DIALOG As Integer = 2
Const HH_WIDGET_DIALOG As Integer = 3
Const HH_TECH_SUPPORT As Integer = 4
```

Context-sensitive WebHelp for Visual C++

Note: The information in this topic is intended for developers who need to connect context-sensitive WebHelp topics to Visual C++ applications.

The sample code shown below uses a function called ShowHelp that will launch a local or remote WebHelp topic regardless of the browser type.

ShowHelp takes two arguments. The first, szTopic, is a URL or local filename. The second argument, bIsLocal, is a boolean value that indicates whether the file is a local filename (TRUE) or a remote URL (FALSE). If the file is local, ShowHelp will tell the browser to find the file in the "Help" subfolder of the folder containing the application program.

```
BOOL ShowHelp(LPCTSTR szTopic, BOOL bIsLocal)
{
TCHAR szDir[MAX_PATH] = "";
if (bIsLocal)
{
// Get directory of application
DWORD dw = GetModuleFileName(AfxGetInstanceHandle(), szDir, MAX_PATH);
TCHAR* pchEnd = _tcsrchr(szDir, '\\') + 1;
ASSERT_POINTER(pchEnd, TCHAR);
*pchEnd = '\0';
// Append subfolder name
_tcscat(szDir, _T("Help"));
}
// Launch topic
HINSTANCE hinst = ShellExecute(NULL, //no parent hwnd
NULL, // open
szTopic, // topic file or URL
NULL, // no parameters
szDir, // folder containing file
SW_SHOWNORMAL); // yes, show it
// handle less than 32 indicates failure
return hinst > (HINSTANCE)32;
}
```

ShowHelp requires an explicit filename or URL. But for context-sensitive Help, it is better to code the application to use context IDs rather than explicit topic names. Use the following sample function, ShowHelpContext, which maps integer context IDs to topic name strings, then calls ShowHelp to launch the topic.

```
BOOL ShowHelpContext(int nContextId)
{
CString strTopic;
BOOL bIsLocal = TRUE;
switch (nContextId)
{
case HH_GADGET_DIALOG:
strTopic = _T("gadget.htm");
break;
case HH_WHATSIT_DIALOG:
strTopic = _T("whatsit.htm");
break;
case HH_WIDGET_DIALOG:
strTopic = _T("widget.htm");
break;
case HH_TECH_SUPPORT:
strTopic = _T("http://www.mycompany.com");
bIsLocal = FALSE;
break;
default:
strTopic = _T("unknown-context.htm");
break;
}
return ShowHelp(strTopic, bIsLocal);
}
```

Using ShowHelpContext makes maintaining context-sensitive Help much easier because if a topic name changes, only one function has to be modified. Note that the context IDs (for example, HH_GADGET_DIALOG) should be declared in a header file that can be shared among the various program modules that use context-sensitive WebHelp. This can be the same file where the prototypes for ShowHelp and ShowHelpContext are defined.

Programming WebHelp for C/C++

The sample code shown below uses a function called ShowHelp that will launch a local or remote WebHelp topic regardless of the browser type.

ShowHelp takes two arguments. The first, szTopic, is a URL or local filename. The second argument, bIsLocal, is a boolean value that indicates whether the file is a local filename (TRUE) or a remote URL (FALSE). If the file is local, ShowHelp will tell the browser to find the file in the "Help" subfolder of the folder containing the application program.

```
BOOL ShowHelp(LPCTSTR szTopic, BOOL bIsLocal)
{
TCHAR szDir[MAX_PATH] = "";
if (bIsLocal)
{
// Get directory of application
DWORD dw = GetModuleFileName(AfxGetInstanceHandle(), szDir, MAX_PATH);
TCHAR* pchEnd = _tcsrchr(szDir, '\\') + 1;
ASSERT_POINTER(pchEnd, TCHAR);
*pchEnd = '\0';
// Append subfolder name
_tcscat(szDir, _T("Help"));
}
// Launch topic
HINSTANCE hinst = ShellExecute(NULL, //no parent hwnd
NULL, // open
szTopic, // topic file or URL
NULL, // no parameters
szDir, // folder containing file
SW_SHOWNORMAL); // yes, show it
// handle less than 32 indicates failure
return hinst > (HINSTANCE)32;
}
```

ShowHelp requires an explicit filename or URL. But for context-sensitive Help, it is better to code the application to use context IDs rather than explicit topic names. Use the following sample function, ShowHelpContext, which maps integer context IDs to topic name strings, then calls ShowHelp to launch the topic.

```
BOOL ShowHelpContext(int nContextId)
{
CString strTopic;
BOOL bIsLocal = TRUE;
switch (nContextId)
{
case HH_GADGET_DIALOG:
strTopic = _T("gadget.htm");
break;
case HH_WHATSIT_DIALOG:
strTopic = _T("whatsit.htm");
break;
case HH_WIDGET_DIALOG:
strTopic = _T("widget.htm");
break;
case HH_TECH_SUPPORT:
strTopic = _T("http://www.mycompany.com");
bIsLocal = FALSE;
break;
default:
strTopic = _T("unknown-context.htm");
break;
}
return ShowHelp(strTopic, bIsLocal);
}
```

Using ShowHelpContext makes maintaining context-sensitive Help much easier because if a topic name changes, only one function has to be modified. Note that the context IDs (for example, HH_GADGET_DIALOG) should be declared in a header file that can be shared among the various program modules that use context-sensitive WebHelp. This can be the same file where the prototypes for ShowHelp and ShowHelpContext are defined.

WebHelp and context-sensitive Help

With WebHelp, authors can create full-featured Help systems that include context-sensitive Help. WebHelp files are supported by applications created in C and C++, Java applets, Visual Basic forms and Web pages.

As with any type of context-sensitive Help, developing context-sensitive WebHelp is a cooperative effort between Help authors and development teams.

Author's role. Creates topics and authors content in the WYSIWYG Editor. Each context-sensitive Help topic is a separate HTML file that describes how to use an application at the dialog or window-level (individual topics for each field or control are not supported).

Application developer's role. Programs the Help topics in the application so the correct ones are displayed when users request help.

WebHelp browser limitations

The following list shows known limitations that occur when end users view a WebHelp project using a specific browser. This information is only for reference and may not apply to every WebHelp project.

Internet Explorer 2.0

- Cascading Style Sheets are not supported. Only embedded and inline styles display.
- Dynamic HTML is not supported. All DHTML effects display as static effects.
- Java is not supported.
- Frames are not supported.

Internet Explorer 3.0x

- Cascading Style Sheets are not supported. Only embedded and inline styles display.
- Dynamic HTML is not supported. All DHTML effects display as static effects.
- Java applets do not resize when browser is resized.
- FONT FACE tag is not supported. Text uses the default font of the browser.
- Microsoft has documented that HTML files do not always display if users request the files in rapid succession. Users can return to a previously displayed page and then reselect the page that didn't display.
- In Internet Explorer 3.00 and 3.01 Java applets cannot read or write local or network media. Therefore, WebHelp can only be accessed via a Web server.

Internet Explorer 5.x

- Due to security issues, references to pages located at the root of any drive will not work properly. This includes references in the table of contents, the index, and within topics.

Netscape Navigator (all versions)

- Navigator cannot load Java applets from locations specified by UNC file names (for example, \\server\share\start.htm) on local area networks. This problem can be solved by mapping a drive letter to the share.

Netscape Navigator 3.0x

- Java applets do not resize when browser is resized.

Netscape Navigator 3.04 or earlier

- Cascading Style Sheets are not supported. Only embedded and inline styles display.
- Dynamic HTML is not supported. All DHTML effects display as static effects.
- ActiveX technology is not supported.
- Margin settings are ignored. This results in a small, unused margin area surrounding the WebHelp navigation pane.

Netscape Navigator 4.60

- Java applets do not resize when browser is resized. (This was fixed in version 4.61.)

Also, please note the following:

See Also references and popup links are not supported in:

- Internet Explorer 2.0 or before
- Netscape Navigator 1.22 or before

Hyperlinks in a popup display the target in the popup and not the main content frame in:

- Netscape Navigator 2.02
- Opera 3.51 and 3.60

Posting WebHelp on a Web server

You can use WebHelp to develop intranet products and pages for Web sites. In order to get your system up and running, you'll need to post (deploy) it on your company's server. Once posted, all network users will be able to access it, from a variety of computing platforms (Windows, MacOS, UNIX, Linux), using Internet Explorer, Netscape Navigator or other browsers.

At most companies and organizations, a Webmaster or network administrator is responsible for managing the information that is posted on the servers. This person is knowledgeable about the policies and procedures for posting your intranet product and Web pages. Find out where you should save your WebHelp files so they are available for posting.

- When you are ready to generate WebHelp output, you specify the name of the folder where your WebHelp source files will be located as well as the name of the start page. The start page is the first topic that is displayed in the right-hand pane of the browser when your Help system opens.
- All the required WebHelp system files are generated and saved in a subfolder in your project folder titled WebHelp (or whatever name you assign).
- If your project includes subfolders, the WebHelp folder will include these subfolders with the appropriate files located in them. Do not rename or move any files (leave the project structure intact).
- Copy the WebHelp folder and its contents to the server where your Webmaster requests you to do so. Be sure to let your Webmaster know the name of the start page (.HTM file).

Tips:

- Be sure to view and test your WebHelp project on your authoring system before it’s posted. This way, you can make sure it includes all the features you want to use with your project and you can check your links, review topics, etc., so they suit your needs.
- Whenever you make changes to your WebHelp project, provide your Webmaster with the new files so they can be updated on the server.

About Help formats

In addition to WinHelp and WinHelp 2000, there are other Help formats available, such as Microsoft HTML Help and JavaHelp. All Help formats have many common characteristics and meet the same needs. A few universal features you'll see in the major Help formats include:

- Graphical navigation and organization support (including multi-level indexes and tables of contents)
- Popup and drop-down text
- Automatic page-turners (called browse sequences)
- Related topic cross-references (called See Also or related topics)
- Full-text searches
- Help windows describing what's currently displayed (context-sensitive Help)

End users' needs also remain constant across Help formats: To find information quickly so they can complete a task or read more about a concept or terminology. Most Help formats also include field-level and screen-level Help, overview topics, and task-oriented topics. No matter what the format or what information is required, Help meets these needs consistently and effectively by providing superior navigation and organization tools.

The main *difference* between Help formats is what type of viewer (such as the WinHelp viewer or a Web browser) or platform they're used with (such as Windows or UNIX). It may be necessary to use JavaHelp on an application written in the Java programming language. Or you may need to deliver Help for a Web-based application. Regardless of the platform or viewer, you can create effective Help systems using RoboHELP.

An added bonus of RoboHELP is the ability to use a single Help project to generate multiple types of Help formats. All the features that make Help systems useful are retained, such as a detailed table of contents, a multi-level index, full-text search capability, and graphical navigation aids – no matter which Help format you select. For example, you can create a Help system *and* printed documentation from one file, eliminating the need to maintain two source files.

Choosing a Help format

RoboHELP Office is an eHelp Corporation package that includes the ability to create the following Help formats:

Microsoft HTML Help

This format is developed by Microsoft to display Help in Windows applications. Microsoft HTML Help is based on Hypertext Markup Language (HTML) files. You can run Microsoft HTML Help on Windows 95, 98, NT 4.0, and 2000 using Internet Explorer 4.x or later. Use the Microsoft HTML Help format to create application Help, stand-alone Help, online policies and procedures, and more, for end users that run Windows and Internet Explorer.

Use RoboHELP HTML to generate Microsoft HTML Help.

WebHelp

WebHelp is a Help format developed by eHelp Corporation that runs reliably on any popular browser and platform. WebHelp provides online Help for multiple computing platforms including Microsoft Windows, Macintosh, Linux, and UNIX with any browser including Internet Explorer or Netscape Navigator. Use WebHelp files to create Web site Help, Help for Web-based applications, intranet Help or content, extranet Help or content, Help for applications in virtually any programming language, online books, EPSS, and more. Select WebHelp to have all end users view your Help files consistently each time, in any environment.

Use RoboHELP HTML to generate WebHelp files.

JavaHelp

This format, developed by Sun Microsystems, is designed to work with applications written in the Java programming language. JavaHelp and Java applications run on a variety of platforms (such as Windows, UNIX, Mac, and Linux).

JavaHelp is a delivery system, not a Help authoring tool. All JavaHelp features are automatically created including contents, index, full-text search, navigation controls, and popup windows, along with HTML-based features such as HTML topics and links, Related Topics buttons, and image files.

Use RoboHELP HTML to generate JavaHelp files.

Oracle Help

Oracle Help for Java is a Help format developed by Oracle Corporation for creating Help in Java or Oracle systems. Oracle Help for Java works with applications written in Java or any other programming language. Oracle Help and Java applications run on a variety of platforms (Windows, UNIX, Macintosh, Linux, etc.). Generate Oracle Help for Java for all your Java-based applications and Oracle servers, tools and applications. RoboHELP Office is the only Help development tool that seamlessly creates Oracle Help for Java.

Use RoboHELP HTML to generate Oracle Help files.

WinHelp

This format is used primarily as application Help and standalone Help for computers that run on Microsoft Windows. WinHelp is the "original" Windows online Help format, based on RTF (Rich Text Format) files. You can run WinHelp on Windows 95, 98, NT 4.0, or 2000. (WinHelp 3 Help systems run primarily on Windows 3.x, but they are supported on Windows 95, 98, and NT 4.0. WinHelp 4 Help systems run on Windows 95, 98, and NT.)

Like WinHelp, WinHelp 2000 is used for application and standalone Help. It offers all of the same features as WinHelp with the added benefits available in the Word 2000 release. With WinHelp 2000 you can easily integrate and update Help systems that are in different Help formats, add powerful new navigation aids, and quickly create the look and feel of Microsoft HTML Help using your WinHelp projects.

Use RoboHELP Classic to generate WinHelp systems.

Printed documentation

RoboHELP Office provides single-source technology for creating printed documentation from your Help projects.

The Documentation Wizard enables you to turn your Help systems, intranet content, online books, and more into high-quality user guides that are suitable for printed documentation.

The Documentation Wizard provides all the features you need for producing printed publications in Microsoft Word format, including text formatting, styles, images, organized files, table of contents, and index.

This format supports the formatting in your HTML and WinHelp topics. You can even use your HTML style sheets or Word templates with the wizard and avoid redoing styles for the printed documentation.

You can generate printed documentation in RoboHELP HTML and RoboHELP Classic.

End-user viewing requirements

The table below shows how each type of online information system is displayed and on which platforms.

End-user system	WinHelp	Microsoft HTML Help	WebHelp	JavaHelp *
Windows 3.1	Built-in viewer	Not available	Web browser	Java virtual machine
Windows 95	Built-in viewer	Internet Explorer 4.0 or later	Web browser	Java virtual machine
Windows NT 4	Built-in viewer	Internet Explorer 4.0 or later	Web browser	Java virtual machine
Windows 98	Built-in viewer	Built-in viewer	Web browser	Java virtual machine
Windows 2000	Built-in viewer	Built-in viewer	Web browser	Java virtual machine
UNIX	Third-party viewer	Not available	Web browser	Java virtual machine
Macintosh	Third-party viewer	Not available	Web browser	Java virtual machine

* Requires JavaHelp viewer, JDK 1.2, and JavaHelp 1.0 or 1.1 components

Notes:

- For systems with built-in viewers, there is no need to ship a browser. For example, if you are creating an HTML Help system for Windows 98 users, you do not need to distribute a Web browser since Internet Explorer is built into the operating system.

Help format features comparison

The following table shows some common Help features and in which Help format you can find them.

Feature	Microsoft HTML Help	WinHelp	WinHelp 2000	WebHelp	JavaHelp	Oracle Help
Expandable and collapsible table of contents	Yes	Yes	Yes	Yes	Yes	Yes
Multi-level index	Yes	Yes	Yes	Yes	Yes	Yes
Merged indexes	Yes	Yes	Yes	No	No	Yes
Merged tables of contents	Yes	Yes	Yes	Yes (3)	No	Yes
Custom TOC icons	Yes	No	No	Yes	No	Yes
Full-text search	Yes (4)	Yes	Yes	Yes	Yes	Yes
Online glossary	Yes (2)	No	No	No	No	No
Related topics (See Also)	Yes	Yes	Yes	Yes	Yes (2)	Yes
ALinks	Yes	Yes	Yes	Yes	Yes	Yes
HTML Help controls (splash screens, WinHelp topic controls, shortcuts, etc.)	Yes	No	No	No	No	No
Browse sequences	Yes (2)	Yes	Yes	Yes	Yes (2)	Yes
Cross-platform	No (5)	Yes (7)	No	Yes	Yes	Yes
Browser-independent	No	Yes	Yes	Yes	No	Yes
Compilation and Compression	Yes	Yes	Yes	No	Yes	Yes
Tri-pane window	Yes	No	Yes	Yes	Yes	Yes
Custom windows	Yes	Yes	Yes	No	No	Yes
Popup windows	Yes	Yes	Yes	Yes	Yes	Yes
Information types	Yes	No	No	No	No	No
ActiveX controls	Yes	No	Yes (6)	Yes (1)	No	No
Dynamic HTML	Yes (1)	No	Yes (6)	Yes (1)	No	No
Display images	Yes	Yes	Yes	Yes	Yes	Yes
Multimedia	Yes	Yes	Yes	Yes	No	No
Forms	Yes	No	Yes (6)	Yes	No	Yes
Scripts	Yes	No	Yes (6)	Yes	No	No
Context-sensitive Help	Yes	Yes	Yes	Yes	Yes	Yes

(1) Only supported on Internet Explorer 4.0 or later. Not supported on Netscape Navigator.

(2) Supported through RoboHELP extensions

(3) When generated from RoboHELP Classic

(4) When compiled

(5) Works on all Windows 32-bit platform

(6) Supported from HTML topics

(7) Third-party version

Inline styles

Inline styles are similar to Microsoft Word's character and paragraph formatting. Inline styles are not named or applied to text. All formatting is done manually with toolbar buttons and dialog options.

Build tags

Build tags enable you to design and create several different HTML Help projects from a single set of source files. They give you control over which topics are included in a file. You create the build tags, then mark the topics you want to use with them.

DLL

Dynamic Link Library. A file containing executable content that can be loaded by an application to complete or extend its functionality. An application can actually be an executable (.EXE) combined with one or more DLLs. DLLs can also contain images, strings, or other resources that can be used by the application. Multiple applications can use the same DLL simultaneously.

Choosing an authoring environment

RoboHELP Office contains two authoring tools you can use to create Help projects: RoboHELP Classic and RoboHELP HTML. Choosing which tool to use is an important decision. Consider the following:

- Is your management or development team requiring you to use a particular tool?
- Are you more comfortable working with Microsoft Word or a WYSIWYG HTML Editor? Do you need to work with both?
- If you are creating a Help system for an application or a particular platform, will one Help format integrate more smoothly into the application than another? (Ask your developer for help with this question.)
- What environment or platform will your end users use to run your Help project? This will help determine which Help format to use.
- Which Help formats do you require? Does your system only run in a Web-based environment or UNIX, for example?

RoboHELP Classic

If you are creating WinHelp or WinHelp 2000 projects, use RoboHELP Classic. In RoboHELP Classic, you can develop WinHelp systems with all the Help features you need using the familiar Microsoft Word authoring environment, including printed documentation. When you have RoboHELP Office, you can also generate several other Help formats in RoboHELP Classic.

RoboHELP HTML

If you are creating HTML-based Help projects (such as Microsoft HTML Help or WebHelp), use RoboHELP HTML. Using the powerful WYSIWYG editor (or another HTML editor), you can create compiled Microsoft HTML Help (.CHM files) and printed documentation. When you have RoboHELP Office, you can create WebHelp (cross-platform, cross-browser Help), JavaHelp, and Oracle Help in addition to Microsoft HTML Help.

WinHelp 2000 overview

WinHelp 2000 provides Help authors with features that take WinHelp into the next millennium. These exciting enhancements offer you new options when designing and developing Help systems. Selecting WinHelp 2000 features is a breeze – just open the Project Settings: WinHelp 2000 tab and select one or more options, click OK, and compile the Help system. It's the easiest way to get your WinHelp systems ready for the next century.

WinHelp 2000 Key Features

- **Explorer View Help.** Explorer View Help updates the look and feel of WinHelp. It displays Help content and navigation inside the same Explorer View Help window, which is similar to Microsoft HTML Help's tri-pane window – Contents, Index, and Search tabs appear to the left of the topic pane, so users never lose their place.
- **Smart See Also button.** The Smart See Also button provides "smart" access to related topics by placing a See Also button on your Help window's button bar. This See Also button is "smart" because if a topic is part of one or more See Also keyword groups, clicking the button automatically displays the list of other topics associated with the same See Also keyword groups.
- **Watermarks.** Watermarks allow you to add background textures, colors, images, and even logos to the Explorer View Help window to give your Help system a distinctive look.
- **Link to HTML topics and Web sites from inside your Help system.** WinHelp 2000 allows you to create HTML links to HTML topics and Internet/intranet pages embedded in your WinHelp system. When users click the hotspot link, WinHelp 2000 launches the selected HTML topic or Web page within the WinHelp window.
- **Insert HTML topics and Web sites inside your Help system.** WinHelp 2000 lets you completely integrate HTML topics and Web pages in your WinHelp system. You can insert a reference to the HTML topic or Internet or intranet Web page as a "topic" in your WinHelp system. This topic can even be included in your WinHelp system's browse sequence.

Smart Publishing Wizard

The Smart Publishing Wizard enables you to easily copy or "publish" WebHelp files or folders to different locations. For example, you can copy individual .HTM files, groups of .HTM files, or an entire project to a corporate intranet, Internet site, your local hard drive, or a network. You can even use various methods to publish the files including FTP, HTTP, FrontPage Extensions (if you have Microsoft FrontPage installed), or your network protocol.

The publishing wizard is "smart" because it only publishes files that are newer and different than the last time the files were published. This saves valuable publishing time by not copying duplicate files to the destination folder.

The Smart Publishing Wizard is available with RoboHELP Office.

Note:

You may need to work with your network system administrator to determine the best method of publishing your files and to obtain access permissions.

Publishing files and folders

The Smart Publishing Wizard provides an easy way to publish files or folders to locations such as a corporate intranet, an Internet site, your local hard drive, or a network.

The Smart Publishing Wizard is available with RoboHELP Office.

Note:

You may need to work with your network system administrator to determine the best method of publishing your files and to obtain access permissions.

To publish files or folders:

1. Open your project and generate WebHelp.
2. After you generate your project, click **Publishing** from the Wizard Results dialog. This displays the Smart Publishing Wizard.
3. In **Source Location** type in the path to the source files you want to publish or click **Select File/Folder** and browse to the files. If you want to include all subfolders within the source location, select **Publish Subfolders**.
4. Click **Next**.
5. Choose the server on which you want to publish your files. You can select one from the list, add a new server to the list, edit an existing server, or delete a server from the list. To sort the list, click on **Server Name or URL**.

   Tip: As a shortcut, you can right-click on a listed server to edit or delete it.
6. Select options from the **Options** list and click **Next**.
7. The Smart Publishing Wizard displays the source location of the files and the location where the files are to be published so you can check to be sure the information is correct. If it is correct, click **Finish**. If you want to edit any information, click **Back** and make changes.
8. When the Smart Publishing Wizard is finished it displays the results. You can click **Print** to print the list of all files published.
9. When you are finished, click **Close**.