

ComponentOne®

DoctoHelp® 7.5

☑ True Single Source Help Authoring



Copyright © 1987-2005 ComponentOne LLC. All rights reserved.

Printed in Canada

Corporate Headquarters
ComponentOne LLC
4516 Henry Street
Suite 500
Pittsburgh, PA 15213 USA

Internet: info@ComponentOne.com
Web site: <http://www.componentone.com>

Sales

E-mail: sales@componentone.com
Telephone: 1.800.858.2739 or 1.412.681.4343 (Pittsburgh, PA USA Office)

Technical Support

See *Technical Support* in this manual for information on obtaining technical support.

Trademarks

ComponentOne Doc-To-Help 7.5 and the ComponentOne Doc-To-Help 7.5 logo are trademarks, and ComponentOne is a registered trademark of ComponentOne LLC. All other trademarks used herein are the properties of their respective owners.

Warranty

ComponentOne warrants that the original CD (or diskettes) are free from defects in material and workmanship, assuming normal use, for a period of 90 days from the date of purchase. If a defect occurs during this time, you may return the defective CD (or disk) to ComponentOne, along with a dated proof of purchase, and ComponentOne will replace it at no charge. After 90 days, you can obtain a replacement for a defective CD (or disk) by sending it and a check for \$25 (to cover postage and handling) to ComponentOne.

Except for the express warranty of the original CD (or disks) set forth here, ComponentOne makes no other warranties, express or implied. Every attempt has been made to ensure that the information contained in this manual is correct as of the time it was written. We are not responsible for any errors or omissions. ComponentOne's liability is limited to the amount you paid for the product. ComponentOne is not liable for any special, consequential, or other damages for any reason.

Copying and Distribution

While you are welcome to make backup copies of the software for your own use and protection, you are not permitted to make copies for the use of anyone else. We put a lot of time and effort into creating this product, and we appreciate your support in seeing that it is used by licensed users only. Please read *License Agreement* and *Licensing* sections in this manual before copying and redistributing any ComponentOne Doc-To-Help 7.5 files.



Table of Contents

Table of Contents	iii
ComponentOne Doc-To-Help 7.5	1
About ComponentOne Doc-To-Help 7.5.....	1
What's Available in Doc-To-Help Professional.....	2
What's New in Doc-To-Help 7.2.....	2
What's New in Doc-To-Help 7.5.....	3
END-USER LICENSE AGREEMENT FOR COMPONENTONE SOFTWARE.....	4
License and Redistributable Files	14
Technical Support.....	15
System Requirements.....	16
Using Anti-virus Software with Doc-To-Help.....	16
Installation	16
Authoring Templates	16
Global Templates	18
Help Files	18
About This Document.....	19
Help Authoring Basics	21
What is a Help System?	21
Building Help Systems with Doc-To-Help.....	26
Help Targets in Doc-To-Help.....	31
The Doc-To-Help Project Editor	39
Navigating the Project Editor.....	39
The Project Icon.....	41
The Views Icon.....	44
The Topics Icon	45
The Contents Icon.....	46
The Index Icon.....	47
The Build Icon	48
Exploring the Doc-To-Help Toolbar for Microsoft Word	48
A Guided Tour of Doc-To-Help	53
Tutorial Conventions	53
Converting Doc-To-Help 2000 Projects	54
Starting a New Help Project.....	55
Working with the First Source File.....	57
Saving the Document and Building a Help File	59
Using Styles to Enhance a Help Project.....	60
Adding Multiple Source Documents to a Help Project	67

Navigating Large Projects	69
Adding Tables in Doc-To-Help.....	72
Adding Topic Links and Jumps.....	74
Adding Graphics and Choosing a Default Topic	76
Using Cross-references	79
Working With Margin Notes	81
Adding Glossary Terms.....	84
Adding Conditional Text.....	85
Applying Lists in Doc-To-Help	90
Indexing in Doc-To-Help.....	92
Creating Printed Manuals	96
Working with Microsoft Word.....	99
The Options Dialog Box.....	99
Word 2000 and Infrequently Used Menu Items	101
Working With Styles in Word.....	102
Working with Projects	107
Creating or Opening a Project Upon Startup.....	107
Compacting a Project File.....	110
Converting Projects with Doc-To-Help	111
Creating a New Project from the Project Editor	115
Opening an Existing Project from the Project Editor	116
Project Settings Properties.....	116
Changing the Current Help Target	118
Building the Current Help Target	118
Viewing the Current Help Target.....	118
Adding a New Help Target.....	119
Changing the Help Target Directory	120
Help Target Properties.....	120
Documents and Templates.....	133
Organizing Your Project Files	133
Adding a New Document to a Project	134
Adding Templates to a Project.....	138
Opening a File From a Project.....	140
Removing a File From a Project	141
Compiling Source Documents	141
Using Styles in Doc-To-Help.....	143
Using Paragraph Styles.....	143
Using Character Styles	149
Importing Styles From a Microsoft Word Document or Template.....	151
Importing Styles From Another Project.....	153
Using Topic Types	154
Using List Styles.....	161

Working with Cascading Style Sheets	163
Defining and Organizing Topics.....	165
The Topics Icon	165
Topic Properties	167
Making a paragraph style active	169
Specifying the Help Contents	170
Specifying the Navigation Sequence.....	171
Providing Links to Subtopics.....	172
Providing Links to Specific Topics	173
Customizing the Related Topic Jumps	174
Labeling Subtopic Links	174
Providing Links to the Next Topic	176
Editing Topic Properties	177
Adding a Link Tag to a Topic	178
Modifying the Help Contents	180
Context Sensitive Help.....	186
Sorting Topics For Easy Viewing With Outlook Style Grouping	192
Filtering Topics in Doc-To-Help.....	194
Links and Hot Spots.....	197
Making a Character Style Active	197
Creating a Jump to a Topic in Another Document	198
Creating Popup Topics	199
Creating Keyword Links (KLinks)	200
Creating Group Links (ALinks).....	201
Using the Add Topic Link Dialog Box.....	203
Creating Cross References.....	204
Using the Word Hyperlink Field to Create Cross Document Links.....	206
Using Margin Notes	207
Mid Topic Jumps	213
Using Dynamic Links.....	215
Building an Index	219
Project Editor Index View.....	219
Using the Index Toolbar	220
Adding Index Keywords Manually	221
Creating Index Keywords Automatically Using Styles.....	221
Adding Secondary Index Keywords Manually.....	223
Renaming an Index Keyword	224
Assigning Topics to an Index Keyword	224
Removing Unused Index Keywords.....	226
Adding an Index Keyword to a Topic	227
Creating Topic Groups.....	228
Renaming a Group	228

Assigning Topics to a Group.....	229
Adding a Group to a Topic.....	229
Removing Topics From Index Keywords or Topic Groups.....	230
Using the Index Elements Auto-Completion Feature	230
Customizing Help Windows	233
HTML Help Window Properties	233
WinHelp 4.0 Window Properties	236
Creating a New Help Window	237
Assigning a Window to a Style.....	237
Editing Help Window Properties.....	238
Using Themes to Modify the Help Window.....	239
Modifying the Background of the Window Pane	242
Modifying the Window Size Using the Size Tool	245
Tips and Techniques	249
Working with Graphics	249
Inserting Standard Tables with Doc-To-Help.....	253
Linking to Internet Sites.....	255
Generating Project Reports.....	260
Managing Project Files	264
Modifying the Glossary	266
Deleting Help Topics.....	271
Using HTML Help Object Tags	271
Using WinHelp Macros	273
Doc-To-Help Shortcut Keys	295
Conditional Text and Attributes.....	299
Specifying Conditional Text.....	299
Using Attributes.....	305
Using the View Conditional Text Dialog	309
Using Modular Help	317
Managing Your Modular Help System	317
Creating a Modular Help Project for WinHelp	319
Creating a Modular Help Project for HTML Help.....	324
Scripting Techniques	329
What are Scripts?	329
Working with Scripts	332
Common Script Operations.....	335
Docucenter for .NET	341
Generation Speed and Supported Word Versions.....	341
Docucenter for .NET Guided Tour	342
Docucenter Controls and Toolbars.....	357
Examining the Automatically Generated Document and Project.....	360

Adding Narrative to a Documenter Project	362
Adding Topic Links to a Documenter Project	364
Multilanguage Support in Documenter for .NET	367
Customization and Localization of Documenter for .NET	367
Formatter Versions in Documenter for .NET	370
Using Documenter Styles to Create Links.....	371
Using Natural Search for Doc-To-Help	387
Natural Search Properties.....	387
Enabling Natural Search.....	388
Defining Your Natural Search Content	388
Modifying the Natural Search Tab Caption.....	390
Defining the "No Match" Message	391
Defining the Number of Natural Search Returns.....	392
Modifying the Search Root Word.....	392
Manually Adding Natural Search Support.....	393
Altering the Natural Search Noise Word List.....	394
Distributing Natural Search with your Help Files	394
Searching a Help File Using Natural Search.....	394
Using the Image Map Editor.....	397
Image Map Editor.....	397
Exploring the Image Map Toolbar Controls	398
Exploring the Topic Link Toolbar Controls	398
Creating Image Hot Spots	399
Modifying the Size of an Image Hot Spot	400
The Modular TOC Utility	403
Using the Modular TOC Utility	403
Exploring the Modular TOC Utility Editor	409
Using the Context String Editor	411
Adding Context Strings	411
Linking Topics to Context Strings	413
Using the Theme Designer	417
Theme Designer	417
Creating a New Theme	418
Exploring the Navigation Bar	419
Exploring the Topic Text Bar.....	427
Exploring the Popup Window Bar	428
Exploring the Secondary Window Bar	429
Exploring the Table of Contents Bar (HTML 4.0 Only).....	430
Exploring the Index Bar (HTML 4.0 Only).....	432
Including Custom HTML Content in Themes	434
Using the Topic Tools	437

Using the Add Topic Button	437
Using the Rename Topic Button	440
Using the Delete Topic Button	442
Object Model Reference	443
Index.....	519

ComponentOne Doc-To-Help 7.5

About ComponentOne Doc-To-Help 7.5

Thank you for purchasing ComponentOne Doc-To-Help 7.5 to convert your Microsoft Word documents into multi-target Help systems! You can create HTML Help 1.x, cross-platform HTML 4.0, WinHelp 4.0, JavaHelp 1.1, *and* print manuals from one set of Word documents – there's no need to purchase separate editions. Plus, Doc-To-Help 7.5 is Microsoft Help 2.0 ready!

Whether you're creating an online Help system or a book, your goal is to communicate clearly with your audience and provide them with the information they need, when they need it. You do that by developing content on one or more subjects (*topics*), often pointing your audience to places where they can find additional information or images (*links*). You give them a hierarchical view, or outline, of the information they'll find (*contents*); a searchable list of words that correspond to one or more topics (*index*); and ways to access (*navigate*) the information.

Doc-To-Help helps you to do all of that by analyzing the styles and templates you apply to Word documents. One of the many plusses of Doc-To-Help is that you're not forced to use specific document styles or templates – you can use your own style conventions, Word's default normal template, or the templates provided with Doc-To-Help. If you wish, you can also use Visual Basic script to govern how topics, links, and other aspects are handled. Doc-To-Help also makes it easy to customize your Help systems by designating portions (*conditional text*) of your documents for individual platforms, Help targets, or author-defined criteria.

Porting your Help system to a different platform is as easy as selecting a list item and clicking a button! Doc-To-Help generates all input files required by the various Help compilers and builds your project accordingly. Graphics can be linked or embedded within source documents and are automatically converted to an appropriate format.

You can use Doc-To-Help in three ways: run it independently of Microsoft Word to process finished documents, by using the default method of running the two programs in tandem while you are writing and editing, or by running Microsoft Word within Doc-To-Help's own authoring environment for easy, in-place editing.

In short, if you know Microsoft Word, you're on your way to creating robust Help systems! Let Doc-To-Help free you from complex Help system mechanics so you can focus on content, not production. The upcoming sections will take you through the process step-by-step. In no time, you'll be up and running!

What's Available in Doc-To-Help Professional

Produce Reference Documentation Automatically

DocuMenter for .NET generates reference documentation from one or more .NET assemblies, producing completely MSDN formatted source documents for use with Doc-To-Help. To learn more about DocuMenter for .NET, see DocuMenter for .NET (page 341).

Add and Modify Topics Without Rebuilding

With **Add**, **Rename** and **Delete** options, insert, modify and link topics without the additional time of multiple project rebuilds. To learn more about modifying topics, see Using the Topic Tools. (page **Error! Bookmark not defined.**)

Natural Language Search

Implement natural language search functionality in your Help files. To learn more about natural language search, see Using Natural Search for Doc-To-Help (page 387).

Graphic Hot Spots

Add hot spot topic links to your images with the image map editor. To learn more about the image map editor, see Using the Image Map Editor (page 397).

Modular Help Improvements

With the Modular TOC Utility, develop modular Help systems that have full tables of contents in both the parent and child Help files. To learn more about improving your modular Help systems, see The Modular TOC Utility (page 403).

Microsoft Help 2.0 Context Strings

Add context sensitive Help to your Help 2.0 projects with the context string editor. To learn more about adding context strings, see Using the Context String Editor (page 411).

Customize your HTML Based Themes

Design and save customized themes for your HTML Help, HTML 4.0 and Help 2.0 targets with the theme designer. To learn more about modifying themes, see Using the Theme Designer (page 417).

What's New in Doc-To-Help 7.2

Produce Reference Documentation Automatically

Docucenter for .NET generates reference documentation from one or more .NET assemblies, producing completely MSDN formatted source documents for use with Doc-To-Help. To learn more about Docucenter for .NET, see [Docucenter for .NET](#) (page 341).

Add and Modify Topics Without Rebuilding

With **Add**, **Rename** and **Delete** options, insert, modify and link topics without the additional time of multiple project rebuilds. To learn more about modifying topics, see [Using the Topic Tools](#) (page 437).

Remove Unused Index Keywords

Clean up your index by removing any unused index keywords from the Doc-To-Help project manager by selecting the new purge command. To learn more, see [Removing Unused Index Keywords](#) (page 226).

New Sorting Options

In addition to alphabetical, by document and style, topic sorts now include topic type and order. To learn more about the new sorting options, see [Sorting to View Order by Document](#) (page 193).

Use Word Hyperlink Field

Use the Word hyperlink field to create a link to any topic in any document in your project. To learn more about creating cross-references using the hyperlink field, see [Using the Word Hyperlink Field to Create Cross Document Links](#) (page 206).

Enhanced Template Functionality

Easily add templates to your project through the target template dropdown. Store custom templates in your project directories for easy transfer and maintenance. To learn more about templates, see [Adding Templates to a Project](#) (page 138) and [Using Custom Templates](#) (page 140).

What's New in Doc-To-Help 7.5

Build Time Speed Enhancement

Help target building time for all HTML-based targets (HTML, HTML Help, Help 2.0 and JavaHelp) is now up to five times faster.

Include Custom HTML Content in Themes in Theme Designer

You are now able to include arbitrary additional HTML content in your themes. To learn more about this, see Including Custom HTML Content in Themes (page 434).

Generation Speed Enhancement in Docucenter for .NET

Generation speed in Docucenter for .NET (page 341) has been dramatically optimized - up to 20 times faster - with Word 2003. To learn more about this, please see Generation Speed and Supported Word Versions (page 341).

Multilanguage Support in Docucenter for .NET

Docucenter for .NET can now generate documentation in English, French, German, Spanish, Italian and Russian. For more information, see Multilanguage Support in Docucenter for .NET (page 367).

Customization and Localization of Docucenter for .NET

Now you can customize Docucenter for .NET output. This new feature allows you to change and add texts, formatting and graphics. To learn more, see Customization and Localization of Docucenter for .NET (page 367).

Formatter Versions in Docucenter for .NET

Docucenter output formatting has been improved for even closer compliance with MSDN formatting conventions. See Formatter Versions in Docucenter for .NET (page 370) for more information.

Improved JavaHelp Target Support

This version has improved support for JavaHelp target. To learn more about JavaHelp, see JavaHelp Target (page 34).

END-USER LICENSE AGREEMENT FOR COMPONENTONE SOFTWARE

IMPORTANT-READ CAREFULLY: This End User License Agreement (this "EULA") contains the terms and conditions regarding your use of the SOFTWARE (as defined below). This EULA contains material limitations to your rights in that regard. You should read this EULA carefully and treat it as valuable property.

I. THIS EULA.

1. **Software Covered by this EULA.** This EULA governs your use of the ComponentOne, LLC ("C1") software product(s) enclosed or otherwise accompanied herewith (individually and collectively, the "SOFTWARE"). The term "SOFTWARE" includes, to the extent provided by C1: 1) any revisions, updates and/or upgrades thereto; 2) any data, image or executable files,

databases, data engines, computer software, or similar items customarily used or distributed with computer software products; 3) anything in any form whatsoever intended to be used with or in conjunction with the SOFTWARE; and 4) any associated media, documentation (including physical, electronic and on-line) and printed materials (the "**Documentation**").

2. **This EULA is a Legally Binding Agreement Between You and C1.** If you are acting as an agent of a company or another legal person, such as an officer or other employee acting for your employer, then "you" and "your" mean your principal, the entity or other legal person for whom you are acting. However, importantly, even if you are acting as an agent for another, you may still be personally liable for violation of federal and State laws, such as copyright infringement.

This EULA is a legally binding agreement between you and C1. You intend to be legally bound to this EULA to the same extent as if C1 and you physically signed this EULA. By installing, copying, or otherwise using the SOFTWARE, you agree to be bound by the terms and conditions contained in this EULA. If you do not agree to all of the terms and conditions contained in this EULA, you may not install or use the SOFTWARE. If, for whatever reason, installation has begun or has been completed, you should cancel installation or un-install the SOFTWARE, as the case may be. (You may click on the "exit" button or its equivalent to immediately abort installation.) If you do not agree to all of these terms and conditions, then you must promptly return the SOFTWARE to the place of business from which you obtained it in accordance with any return policies of such place of business. Return policies may vary between or among resellers, and you must comply with your particular reseller's return policies as agreed at the point of purchase. If the place of business from which you purchased the SOFTWARE does not honor a complete refund for a period of thirty (30) days from the date of proof of purchase, then you may return the SOFTWARE directly to C1 for a period of thirty (30) days from the date of your purchase. To return the product directly to C1, you must obtain a C1 Return Authorization Number by contacting C1, and you must forward all items purchased, including the proof of purchase, directly to C1. The return must be postage-prepaid, and post-marked within thirty (30) days from the proof of purchase, time being of the essence. The return option to C1 is only available to the original purchaser of an unopened factory packaged item.

II. YOUR LICENSE TO DEVELOP AND TO DISTRIBUTE.

As provided in more detail below, this EULA grants you two licenses: 1) a license to use the SOFTWARE to develop other software products (the "**Development License**"); and 2) a license to use and/or distribute the Developed Software (the "**Distribution License**").

Both of these licenses (individually and collectively, the "**Licenses**") are explained and defined in more detail below.

1. **Definitions.** The following terms have the respective meanings as used in this EULA:

"**Network Server**" means a computer with one or more computer central processing units (CPU's) that operates for the purpose of serving other computers logically or physically connected to it, including, but not limited to, other computers connected to it on an internal network, intranet or the Internet. "**Web Server**" means a type of Network Server that serves other computers more particularly connected to it over an intranet or the Internet.

"**Developed Software**" means those computer software products that are developed by or through the use of the SOFTWARE. "**Developed Web Server Software**" means those Developed Software products that reside logically or physically on at least one Web Server and are operated (meaning the computer software instruction set is carried out) by the Web Server's central processing unit(s) (CPU). "**Developed Legacy Software**" means those Developed Software products that are not Developed Web Server Software, including, for example, stand-alone applications and applications accessed by a file server only.

"**Redistributable Files**" means the SOFTWARE files or other portions of the SOFTWARE that are provided by C1 and are identified as such in the Documentation for distribution by you with the Developed Software.

"**Developer**" means a human being or any other automated device using the SOFTWARE in accordance with the terms and conditions of this EULA.

"**Developer Seat License**" means that each Developer using or otherwise accessing the programmatic interface or the SOFTWARE must obtain the right to do so by purchasing a separate End User License.

"**Source Code**" shall mean computer software code or programs in human readable format, such as a printed listing of such a program written in a high-level computer language. The term "Source Code" includes, but is not limited to, documents and materials in support of the development effort of the SOFTWARE, such as flow charts, pseudo code and program notes.

2. **Your Development License.** You are hereby granted a limited, royalty-free, non-exclusive right to use the SOFTWARE to design, develop, and test Developed Software, on the express condition that, and only for so long as, you fully comply with all terms and conditions of this EULA.

The SOFTWARE is licensed to you on a Developer Seat License basis.

The Developer Seat License means that you may perform an installation of the SOFTWARE for use in designing, testing and creating Developed Software by a single Developer on one or more computers, each with a single set of input devices, so long as such computer/computers is/are used only by one single Developer at any given time and not concurrently. Conversely, you may not install or use the SOFTWARE on a computer that is a network server or a computer at which the SOFTWARE is used by more than one Developer. You may not network the SOFTWARE or any component part of it, where it is or may be used by more than one Developer unless you purchase an additional Development License for each Developer. You must purchase another separate license to the SOFTWARE in order to add additional developer seats, whether the additional developers are accessing the SOFTWARE in a stand-alone environment or on a computer network.

In all cases, you may not use C1's name, logo, or trademarks to market your Developed Software without the express written consent of C1; (b) you must include the following C1 copyright notice in your Developed Software documentation and/or in the "About Box" of your Developed Software, and wherever the copyright/rights notice is located in the Developed Software ("Portions Copyright © ComponentOne, LLC 1991-2002. All Rights Reserved."); (c) agree to indemnify, hold harmless, and defend C1, its suppliers and resellers, from and against any claims or lawsuits, including attorney's fees that may arise from the use or distribution of your Developed Software; (d) you may use the SOFTWARE only to create Developed Software that is significantly different than the SOFTWARE.

3. **Your Distribution License.**

License to Distribute Developed Software. Subject to the terms and conditions in this EULA, you are granted the license to use and to distribute Developed Software on a royalty-free basis, provided that the Developed Software incorporates the SOFTWARE as an integral part of the Developed Software in machine-language compiled format (customarily an ".exe", or ".dll", etc.). You may not distribute, bundle, wrap or subclass the SOFTWARE as Developed Software which, when used in a "design-time" development environment, exposes the programmatic interface of the SOFTWARE. You may distribute, on a royalty-free basis, Redistributable Files with Developed Software only.

4. **Specific Product Limitations.** Notwithstanding anything in this EULA to the contrary, if the license you have purchased is for any of the following products, then the following additional limitations will apply:

a. ComponentOne Reports for .NET Designer Edition. ComponentOne Reports for .NET Designer Edition includes at least: 1) one dynamic link library

(c1.win.c1reportdesigner.dll) file known as C1ReportDesigner Component, 2) one executable (ReportDesigner.exe) file known as C1ReportDesigner Application and, 3) the Source Code of the C1ReportDesigner Application. The C1ReportDesigner Component is subject to the general terms and restrictions set forth in this EULA. The C1ReportDesigner Application is an executable file used to design and prepare reports; the C1ReportDesigner Application may be distributed, free of royalties, only in conjunction with the Developed Software.

Subject to the terms and conditions in this EULA, C1 hereby grants you the right to use the C1ReportDesigner Application Source Code. You are hereby also granted the right to modify such Source Code and to create derivative works that are based on the licensed Source Code. You may distribute the derivative works that you develop, solely in object code format and exclusively in conjunction with and/or as a part of the Developed Software. You are expressly not granted the right to distribute, disclose or otherwise make available to any third party the licensed Source Code, any modified version, derivative work, or any portion thereof, in source code format.

C1 shall retain all right, title and interest in and to the licensed Source Code, and all C1 updates, modifications or enhancements thereof. Nothing herein shall be deemed to transfer any ownership or title rights in and to the licensed Source Code from C1 to you.

SOURCE CODE IS LICENSED TO YOU AS IS. C1 DOES NOT AND SHALL NOT PROVIDE YOU WITH ANY TECHNICAL SUPPORT FOR YOUR SOURCE CODE LICENSE.

b. VSView Reporting Edition. VSView Reporting Edition includes at least one executable file listed as "VSRptX.exe" (where X indicates the version number i.e.7,8, etc.), known as "**Designer.**" The file "VSRptX.exe", or any upgrade or future versions of the Designer, are subject to the restrictions set forth in this EULA and may not be distributed with your Developed Software or in any other way.

c. Doc-to-Help and ComponentOne Natural Search. You may use Doc-To-Help to create online help, manuals or other documentation in electronic or printed format (the "**Output Documents**"). You may distribute and incorporate in such Output Documents those files identified in the documentation as Redistributable Files. Except for those specific Redistributable Files, you MAY NOT distribute the SOFTWARE, in any format, to others.

d. Studio Products. You may not share the component parts of the Studio Products licensed to you with other Developers, nor may you allow the use and/or installation of such components by other Developers.

e. ComponentOne Response and SOAP Channel. ComponentOne Response is intended to be installed on a Network Server. C1 grants to you the following rights to the SOFTWARE: a) Installation: You may install one copy of the SOFTWARE on a single Network Server; b) Use: When installed and initialized, the SOFTWARE creates a database file which contains the embodiment of a solution knowledge base (the database hereinafter referred to as the "**Knowledge Base**").

You may use the SOFTWARE to create Knowledge Bases on one Network Server only. To create or to operate Knowledge Bases in more than one Network Server, you must purchase one additional SOFTWARE license for each additional Network Server.

5. **Updates/Upgrades; Studio Subscription.** Subject to the terms and conditions of this EULA, the Licenses are perpetual. Updates and upgrades to the SOFTWARE may be provided by C1 from time-to-time, and, if so provided by C1, are provided upon the terms and conditions offered at that time by C1 in its sole discretion. C1 may provide updates and upgrades to the SOFTWARE for free or for any charge, at any time or never, and through its chosen manner of access and distribution, all in C1's sole and complete discretion.

C1 licenses certain of its separately-licensed products bundled together in a product suite, called the C1 "Studio" product line (the "**Studio Products**"). The exact separately-licensed products that are bundled into the Studio Products may change from time-to-time in C1's sole discretion. If the SOFTWARE is identified as a C1 "Studio" product, then the SOFTWARE is one of the Studio Products. The SOFTWARE and the Studio Products are revised from time-to-time (meaning, for example, revised with updates, upgrades and, in the case of Studio products, possibly changes to which specific products are included in the bundle). For you to be entitled to receive any such revisions to the SOFTWARE or the Studio Products, as the case may be, you must have a valid SOFTWARE license or a valid Studio subscription. The original purchaser of the SOFTWARE or of a Studio product receives a one-year subscription from the date of purchase of the SOFTWARE. After one year, the Studio subscription and/or the SOFTWARE license must be renewed to continue to be entitled to receive the SOFTWARE and/or the Studio Products revisions as the case may be.

6. **Serial Number.** Within the packaging of the SOFTWARE, a unique serial number (the "**Serial Number**") is included, which allows for the registration of the SOFTWARE. The Serial Number is subject to the restrictions set forth in this

EULA and may not be disclosed or distributed either with your Developed Software or in any other way. The disclosure or distribution of the Serial Number shall constitute a breach of this EULA, the effect of which shall be the automatic termination and revocation of all the rights granted herein.

7. **Evaluation Copy.** If you are using an "evaluation copy" or similar version, specifically designated as such by C1 on its website or otherwise, then the Licenses are limited as follows: a) you are granted a license to use the SOFTWARE for a period of thirty (30) days counted from the day of installation (the "**Evaluation Period**"); b) upon completion of the Evaluation Period, you shall either i) delete the SOFTWARE from the computer containing the installation, or you may ii) contact C1 or one of its authorized dealers to purchase a license of the SOFTWARE, which is subject to the terms and limitations contained herein; and c) any Developed Software may not be distributed or used for any commercial purpose.

III. INTELLECTUAL PROPERTY.

1. **Copyright.** You agree that all right, title, and interest in and to the SOFTWARE (including, but not limited to, any images, photographs, animations, video, audio, music, text, and "applets" incorporated into the SOFTWARE), and any copies of the SOFTWARE, and any copyrights and other intellectual properties therein or related thereto are owned exclusively by C1, except to the limited extent that C1 may be the rightful license holder of certain third-party technologies incorporated into the SOFTWARE. The SOFTWARE is protected by copyright laws and international treaty provisions. The SOFTWARE is licensed to you, not sold to you. C1 reserves all rights not otherwise expressly and specifically granted to you in this EULA.
2. **Backups.** You may either: (a) copy the SOFTWARE solely for backup or archival purposes; or (b) install the SOFTWARE on a single computer, provided you keep the original solely for backup or archival purposes. Notwithstanding the foregoing, you may not copy the Documentation.
3. **General Limitations.** You may not reverse engineer, decompile, or disassemble the SOFTWARE, except and only to the extent that applicable law expressly permits such activity notwithstanding this limitation.
4. **Software Transfers.** You may not rent or lease the SOFTWARE. You may transfer the SOFTWARE to another computer, provided that it is completely removed from the computer from which it was transferred. You may permanently transfer all of your rights under the EULA, provided that you retain no copies, that you transfer all the SOFTWARE (including all component parts, the media and printed materials, any dates, upgrades, this EULA and, if applicable, the Certificate of Authenticity), and that the recipient agrees to the terms and conditions of this EULA as provided herein. If the SOFTWARE is an

update or upgrade, any transfer must include all prior versions of the SOFTWARE.

5. **Termination.** Without prejudice to any other rights it may have, C1 may terminate this EULA and the Licenses if you fail to comply with the terms and conditions contained herein. In such an event, you must destroy all copies of the SOFTWARE and all of its component parts.
6. **Export Restrictions.** You acknowledge that the SOFTWARE is of U.S. origin. You acknowledge that the license and distribution of the SOFTWARE is subject to the export control laws and regulations of the United States of America, and any amendments thereof, which restrict exports and re-exports of software, technical data, and direct products of technical data, including services and Developed Software. You agree that you will not export or re-export the SOFTWARE or any Developed Software, or any information, documentation and/or printed materials related thereto, directly or indirectly, without first obtaining permission to do so as required from the United States of America Department of Commerce's Bureau of Export Administration ("**BXA**"), or other appropriate governmental agencies, to any countries, end-users, or for any end-uses that are restricted by U.S. export laws and regulations, and any amendments thereof, which include, but are not limited to, the following:

Restricted Countries: Restricted Countries currently include, but are not necessarily limited to Cuba, Iran, Iraq, Libya, Montenegro, North Korea, Serbia, Sudan, and Syria.

Restricted End-Users: Any End-User whom you know or have reason to know will use SOFTWARE or Developed Software in the design, development, or production of missiles and missile technology, nuclear weapons and weapons technology, or chemical and biological weapons. Any national of any of the Restricted Countries, wherever located, who intends to transmit or transport the SOFTWARE or Developed Software to one of the Restricted Countries.

Restricted End-Uses: Any use of SOFTWARE and Developed Software related to the design, development, or production of missiles and missile technology, nuclear weapons and weapons technology, or chemical and biological weapons.

These restrictions change from time to time. You represent and warrant that neither the BXA nor any other United States federal agency has suspended, revoked or denied your export privileges. C1 acknowledges that it shall use reasonable efforts to supply you with all reasonably necessary information regarding the SOFTWARE and its business to enable you to fully comply with the provisions of this Section. If you have any questions regarding your obligations under United States of America export regulations, you should contact the Bureau of Export Administration, United States Department of

Commerce, Exporter Counseling Division, Washington DC. U.S.A. (202) 482-4811, <http://www.bxa.doc.gov>.

7. **U.S. Government Restricted Rights.** The SOFTWARE and documentation are provided with RESTRICTED RIGHTS. For solicitations issued before December 1, 1995, by the United States of America, its agencies and/or instrumentalities (the "**Government**"), other than the Department of Defense, the use, duplication or disclosure of the software and documentation provided to the Government under this EULA shall be subject to the RESTRICTED RIGHTS as set forth in subparagraphs (c)(1) and (2) of the Commercial Computer Software - Restricted Rights clause at 48 CFR ch.1 52.227-19. For solicitations issued before September 29, 1995, by the Department of Defense, the use, duplication or disclosure of the software and documentation provided under this EULA shall be subject to the RESTRICTED RIGHTS as set forth in subparagraph (c)(1)(ii) of the Rights in Technical Data and Computer Software clause at 48 CFR ch.2 252.227-7013. You will comply with any requirements of the Government to obtain such RESTRICTED RIGHTS protection, including without limitation, the placement of any restrictive legends on the SOFTWARE, and any license agreement used in connection with the distribution of the SOFTWARE. Manufacturer is ComponentOne, LLC, 4516 Henry Street, Suite 501, Pittsburgh, Pennsylvania 15213 USA. For solicitations issued by the Government on or after December 1, 1995 and the Department of Defense on or after September 29, 1995, the only rights provided in the software and documentation provided herein shall be those contained in this EULA. Under no circumstances shall C1 be obligated to comply with any Governmental requirements regarding the submission of or the request for exemption from submission of cost or pricing data or cost accounting requirements. For any distribution of the SOFTWARE that would require compliance by C1 with the Government's requirements relating to cost or pricing data or cost accounting requirements, you must obtain an appropriate waiver or exemption from such requirements for the benefit of C1 from the appropriate Government authority before the distribution and/or license of the SOFTWARE to the Government.

IV. WARRANTIES AND REMEDIES.

1. **Limited Warranty.** C1 warrants that the original media, if any, are free from defects for ninety (90) days from the date of delivery of the SOFTWARE. **EXCEPT AS OTHERWISE PROVIDED IN THE PRECEDING SENTENCE, AND TO THE MAXIMUM EXTENT PERMITTED BY APPLICABLE LAW, C1 EXPRESSLY DISCLAIMS ANY WARRANTY FOR THE SOFTWARE, DOCUMENTATION AND ANYTHING ELSE PROVIDED BY C1 HEREBY AND C1 PROVIDES THE SAME IN "AS IS" CONDITION WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESS OR IMPLIED, INCLUDING, WITHOUT LIMITATION, THE IMPLIED WARRANTIES OF MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE. THE**

ENTIRE RISK ARISING OUT OF USE OR PERFORMANCE OF THE SOFTWARE AND DOCUMENTATION REMAINS WITH YOU. THIS LIMITED WARRANTY GIVES YOU SPECIFIC LEGAL RIGHTS. YOU MAY HAVE OTHERS WHICH VARY FROM STATE TO STATE.

- Limited Remedy.** C1's entire liability and your exclusive remedy under this EULA shall be, at C1's sole option, either (a) return of the price paid for the SOFTWARE; (b) repair the SOFTWARE through updates distributed online or otherwise in C1's discretion; or (c) replace the SOFTWARE with SOFTWARE that substantially performs as described in the SOFTWARE documentation, provided that you return the SOFTWARE in the same manner as provided in Section I.2 for return of the SOFTWARE for non-acceptance of this EULA. Any media for any repaired or replacement SOFTWARE will be warranted for the remainder of the original warranty period or thirty (30) days, whichever is longer. **THESE REMEDIES ARE NOT AVAILABLE OUTSIDE OF THE UNITED STATES OF AMERICA. TO THE MAXIMUM EXTENT PERMITTED BY APPLICABLE LAW, IN NO EVENT SHALL C1 BE LIABLE FOR ANY DAMAGES WHATSOEVER (INCLUDING, WITHOUT LIMITATION, DAMAGES FOR LOSS OF BUSINESS PROFIT, BUSINESS INTERRUPTION, LOSS OF BUSINESS INFORMATION, OR ANY OTHER PECUNIARY LOSS) ARISING OUT OF THE USE OR INABILITY TO USE THE SOFTWARE, EVEN IF C1 HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES. BECAUSE SOME STATES/JURISDICTIONS DO NOT ALLOW THE EXCLUSION OR LIMITATION OF LIABILITY FOR CONSEQUENTIAL OR INCIDENTAL DAMAGES IN CERTAIN CASES, THE ABOVE LIMITATION MAY NOT APPLY TO YOU.**

V. MISCELLANEOUS.

- This is the Entire Agreement.** This EULA (including any addendum or amendment to this EULA included with the SOFTWARE) is the final, complete and exclusive statement of the entire agreement between you and C1 relating to the SOFTWARE. This EULA supersedes any prior and contemporaneous proposals, purchase orders, advertisements, and all other communications in relation to the subject matter of this EULA, whether oral or written. No terms or conditions, other than those contained in this EULA, and no other understanding or agreement which in any way modifies these terms and conditions, shall be binding upon the parties unless entered into in writing executed between the parties, or by other non-oral manner of agreement whereby the parties objectively and definitively act in a manner to be bound (such as by continuing with an installation of the SOFTWARE, "clicking-through" a questionnaire, etc.) Employees, agents and other representatives of C1 are not permitted to orally modify this EULA.

2. **You Indemnify C1.** . You agree to indemnify, hold harmless, and defend C1 and its suppliers and resellers from and against any and all claims or lawsuits, including attorney's fees, which arise out of or result from your breach of any of the terms and conditions of this EULA.
3. **Interpretation of this EULA.** If for any reason a court of competent jurisdiction finds any provision of this EULA, or any portion thereof, to be unenforceable, that provision of this EULA will be enforced to the maximum extent permissible so as to effect the intent of the parties, and the remainder of this EULA will continue in full force and effect. Formatives of defined terms shall have the same meaning of the defined term. Failure by either party to enforce any provision of this EULA will not be deemed a waiver of future enforcement of that or any other provision. Except as otherwise required or superseded by law, this EULA is governed by the laws of the Commonwealth of Pennsylvania, without regard to its conflict of laws principles. The parties consent to the personal jurisdiction and venue of the Commonwealth of Pennsylvania, in the County of Allegheny, and agree that any legal proceedings arising out of this EULA shall be conducted solely in such Commonwealth. If the SOFTWARE was acquired outside the United States, then local law may apply.

License and Redistributable Files

Doc-To-Help 7.5 is developed and published by ComponentOne LLC. You may use it for Help authoring in conjunction with Microsoft Word. You may also distribute the following files, royalty free, with any online Help system or printed manual you develop:

- Any file generated by Doc-To-Help within an output subdirectory relative to a Doc-To-Help project file.

End-users of your online Help systems and print manuals are **not** licensed to use Doc-To-Help for authoring, and may **not** redistribute any file not cited above.

You are **not** licensed to distribute any Doc-To-Help file to users for development purposes. You are **not** allowed to add or transfer the Doc-To-Help license key to the registry of your users' computer(s).

You are **not** licensed to redistribute any version of Microsoft Internet Explorer provided with Doc-To-Help.

It is your responsibility to make such restrictions clear to your users.

Technical Support

Doc-To-Help 7.5 is developed and supported by ComponentOne LLC, a company formed by the merger of APEX Software Corporation and VideoSoft. You can obtain technical support using any of the following methods:

ComponentOne Web site

The ComponentOne Web site at www.componentone.com provides a wealth of information and software downloads for Doc-To-Help users, including:

- Descriptions of the various support options, including phone support.
- Answers to frequently asked questions (FAQ's) about Doc-To-Help, organized by functionality. Please consult the FAQ's before contacting us directly, as this can save you time and also introduce you to other useful information pertaining to Doc-To-Help.
- Free product updates, which provide you with bug fixes and new features.

Internet e-mail

For technical support through the Internet, e-mail us at:

support.d2h@componentone.com

To help us provide you with the best support, please include the following information when contacting ComponentOne:

- Your ComponentOne product serial number.
- The version and name of your operating system.
- Your development environment and its version.

For more information on technical support, go to:

www.componentone.com/support

Peer-to-Peer newsgroup

ComponentOne also sponsors a peer-to-peer newsgroup for Doc-To-Help users. ComponentOne does not offer formal technical support in this newsgroup, but instead sponsors it as a forum for users to post and answer each other's questions regarding Doc-To-Help. However, ComponentOne may monitor the newsgroup to ensure accuracy of information and provide comments when necessary. You can access the newsgroup from the ComponentOne Web site at www.componentone.com/newsgroups.

System Requirements

Computer/Processor	PC with Pentium II 500 MHz or greater processor.
Memory	128 MB of RAM Minimum.
Hard Disk Space	25 MB hard disk space for the Doc-To-Help application and related files. 13 MB hard disk space for MDAC 2.5.
Operating System	Microsoft Windows 98, ME, XP, 2000 or NT 4.0 (with Service Pack 3 or later).
Microsoft Word	Microsoft Word 2000 (version 9.0) or greater.
.NET Framework	Microsoft .NET framework version 1.0 or 1.1.

Using Anti-virus Software with Doc-To-Help

Note that some anti-virus software can limit the functionality of a range of software applications, including Doc-To-Help.

For example, anti-virus software may:

- Interfere with the integration between Doc-To-Help and Microsoft Word
- Limit or prohibit the use of scripting technology

If your anti-virus software alerts you with a warning while using Doc-To-Help, it may be necessary to modify the properties of your anti-virus software.

For known issues relating to anti-virus software, contact Technical Support.

Installation

Insert the Doc-To-Help 7.5 product CD in your CD-ROM drive and follow the instructions presented.

If your Doc-To-Help 7.5 product CD does not auto start, click **Start** from the Windows Task Bar, choose **Run**, then type SETUP.EXE in the **Open** textbox and click **OK**.

Authoring Templates

Doc-To-Help includes the Microsoft Word document templates used to produce its own documentation. Although you are not required to use them, you may find them helpful as a starting point.

C1H_DOTNET_HLP.DOT	This template is used to format WinHelp for Docucenter for .NET projects.
C1H_DOTNET_HTML.DOT	This is the template used to format both standard HTML and Microsoft HTML Help for Docucenter for .NET projects..
C1H_DOTNET_PRN.DOT	This is the template used to format the printed manual target for Docucenter for .NET projects.
C1H_DOTNET_SRC.DOT	This is the template used to author the source documents for Docucenter for .NET.
C1H_HELP.DOT	This is the template used to format WinHelp.
C1H_HTML.DOT	This is the template used to format both standard HTML and Microsoft HTML Help.
C1H_IN.DOT	This template is used during the Doc-To-Help 2000 to Doc-To-Help 7.5 conversion process.
C1H_IN_A4.DOT	This template is used during the Doc-To-Help 2000 to Doc-To-Help 7.5 conversion process (A4 size paper).
C1H_INSD.DOT	This template is used during the Doc-To-Help 2000 to Doc-To-Help 7.5 conversion process.
C1H_INSD_A4.DOT	This template is used during the Doc-To-Help 2000 to Doc-To-Help 7.5 conversion process (A4 size paper).
C1H_INSM.DOT	This template is used during the Doc-To-Help 2000 to Doc-To-Help 7.5 conversion process.
C1H_INSM_A4.DOT	This template is used during the Doc-To-Help 2000 to Doc-To-Help 7.5 conversion process (A4 size paper).
C1H_NORM.DOT	This is the template used to author the source documents.
C1H_NORM_A4.DOT	This is the template used to author the source documents (A4 size paper).
C1H_PRNORM.DOT	This is the template used to format the printed manual target.
C1H_PRNORM_A4.DOT	This is the template used to format the printed manual target (A4 size paper).
C1H_PRSIDE.DOT	This is the template used to format the standard sidehead printed manual.

C1H_PRSIDE_A4.DOT	This is the template used to format the standard sidehead printed manual (A4 size paper).
C1H_PRSMAL.DOT	This is the template used to format the standard small printed manual. Note that the actual fonts used may not be present on your system.
C1H_PRSMAL_A4.DOT	This is the template used to format the standard small printed manual (A4 size paper). Note that the actual fonts used may not be present on your system.
C1H_SIDE.DOT	This is the template used to format the standard sidehead source documents.
C1H_SIDE_A4.DOT	This is the template used to format the standard sidehead source documents (A4 size paper).
C1H_SMAL.DOT	This is the template used to format the standard small source documents. Note that the actual fonts used may not be present on your system.
C1H_SMAL_A4.DOT	This is the template used to format the standard small source documents (A4 size paper). Note that the actual fonts used may not be present on your system.

Doc-To-Help setup installs the templates to Microsoft Word's default template folder and also places backup copies in the *Templates* subdirectory of the Doc-To-Help installation directory.

For more information, see [Adding Templates to a Project](#) (page 138).

Global Templates

Doc-To-Help also installs two global templates named `C1D2HAuthor.dot` and `C1D2HEngine.dot` into the Microsoft Office startup directory.

Help Files

Doc-To-Help places two HTML Help files in the `DocToHelp6` directory.

C1D2H70.CHM	This is the primary help file, providing you with online documentation for Doc-To-Help
C1D2H70API.CHM	This subsidiary help file is used to populate the Help Pane of the Doc-To-Help project editor and is not a “stand alone” file. This file should not be renamed or moved from the directory where the Doc-To-Help application resides.

About This Document

Help Authoring Basics (page 21) describes the key elements of a Help system and provides a brief overview of how they are implemented using Doc-To-Help.

The Doc-To-Help Project Editor (page 39) describes the main features of the user interface as well as a brief description of the properties for each feature.

A Guided Tour of Doc-To-Help (page 53) contains a detailed set of instructions for transforming a set of ordinary Microsoft Word documents (included with Doc-To-Help) into online Help for multiple platforms.

Working with Microsoft Word (page 99) provides a description of the suggested application settings as well as instructions on working with styles and templates in Microsoft Word.

Working with Projects (page 107) contains instructions on creating, opening and working with the Help project within the project editor.

Documents and Templates (page 133) provides instructions on working with your source documents and templates within the project editor.

Using Styles in Doc-To-Help (page 143) provides information on how to define and modify paragraph and character styles within the project editor.

Defining and Organizing Topics (page 165) provides information on how to use the project editor to develop your topic structure, organize the navigation sequence and modify the contents of your Help project.

Links and Hot Spots (page 197) contains instructions on developing your links, cross-references, margin notes and mid topic jumps.

Building an Index (page 219) contains instructions on how to define you index, add primary and secondary keywords and create topic groups.

Customizing Help Windows (page 233) provides information on how to define and modify your Help windows.

Tips and Techniques (page 249) illustrates the advanced features of Doc-to-Help: graphics and Internet links, printing, reporting, modifying the glossary and project file management.

Conditional Text and Attributes (page 299) provides information on how to specify conditional text and how to use attributes to refine your Help files for specific audiences.

Using Modular Help (page 317) explains how to create and manage a modular Help project for WinHelp or HTML Help.

Scripting Techniques (page 329) explains the use of Microsoft VBScript code in Doc-to-Help projects, with examples of common operations.

Using Natural Search for Doc-To-Help (page 387) describes how to implement our natural language search component.

Using the Image Map Editor (page 397) demonstrates how to define topic link hot spots by using the image map control.

The Modular TOC Utility (page 403) contains instructions on how to modify your modular Help system so that both parent and child Help files have a full table of contents.

Using the Context String Editor (page 411) demonstrates how to add Microsoft Help 2.0 context strings to your Help project.

Using the Theme Designer (page 417) provides information on how to define and save custom themes for HTML Help, HTML 4.0 and Help 2.0 targets.

Documenter for .NET (page 341) contains instructions that allow you to use Documenter to automatically create reference documentation by simply selecting a .NET assembly.

Using the Topic Tools (page 437) describes how to add, rename and remove fully functional topics without rebuilding your project.

Object Model Reference (page 443) describes the individual properties and methods exposed by the Doc-to-Help object model.

Help Authoring Basics

If you are new to Help authoring, please read this chapter to learn about the basic elements of a Help system and how you would implement them with ComponentOne Doc-To-Help 7.5.

If you are an experienced Help author, you can skip the introductory section, but the remainder of this chapter covers key information that you need to understand in order to get the most out of Doc-To-Help.

What is a Help System?

From the reader's perspective, a Help system incorporates some or all of the following five elements:

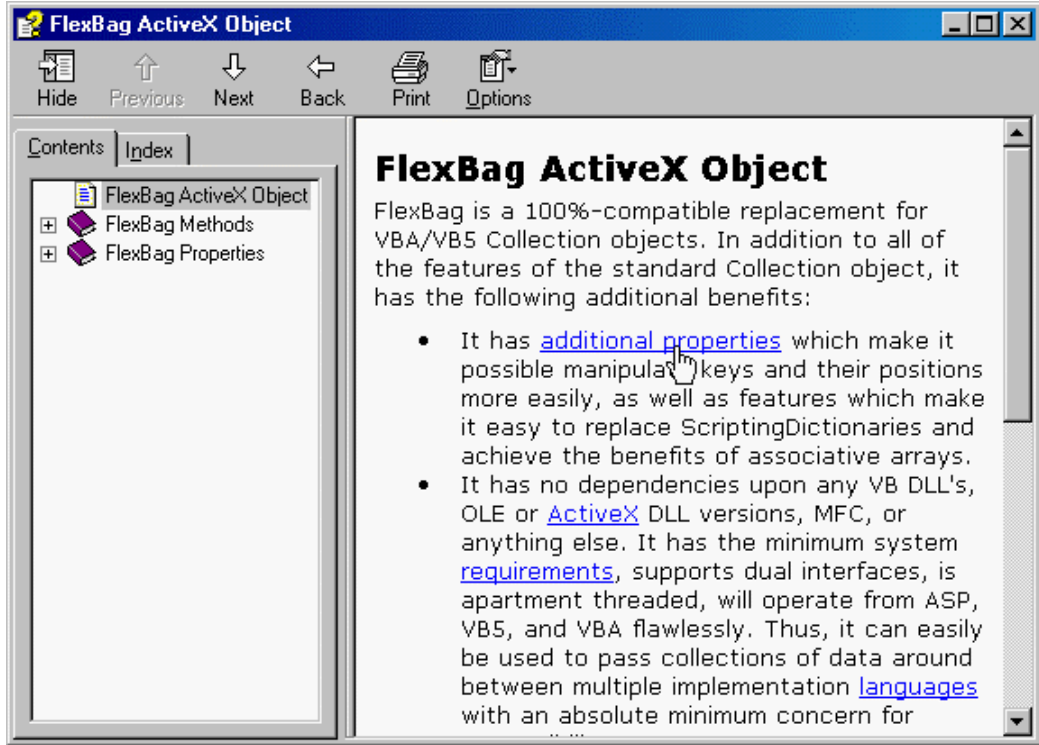
Topics	A set of "pages" containing static text and graphics.
Contents	A hierarchical view, or outline, of the entire Help system.
Index	A searchable list of keywords, each of which corresponds to one or more topics.
Navigation	A mechanism for accessing topics sequentially or by subtopic.
Hypertext	Interactive elements that link individual words and phrases with other topics or topic groups.

Note that this definition of a Help system doesn't presume a particular platform or even a computer! A printed manual is a Help system of sorts, as it has a set of topics, a table of contents, an index, and navigation (via your thumb and forefinger). A typical home page on the World Wide Web doesn't have a site map or a searchable index, but may include hyperlinks to other "Help topics" or a navigation sequence that steps through a set of vacation photos or baby pictures.

Topics Defined

A topic comprises static text and graphics that represents a "page" of information. In a well-organized Help system, topics are concise and serve a specific purpose, such as explaining how to carry out a procedure, defining a glossary term, or providing a list of related topics.

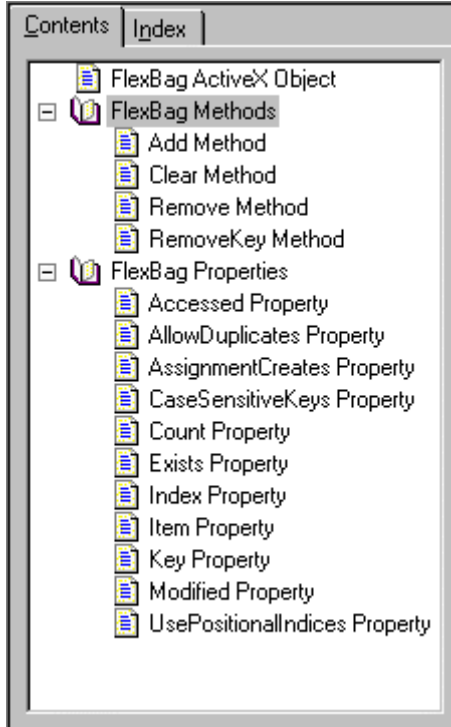
The following figure shows a topic in HTML Help.



Contents Defined

The term contents denotes a hierarchical view, or outline, of the entire Help system that lists topic titles in a way that shows parent-child relationships. In an online Help system, selecting a title displays the corresponding topic. In a printed manual, the table of contents shows the corresponding page number.

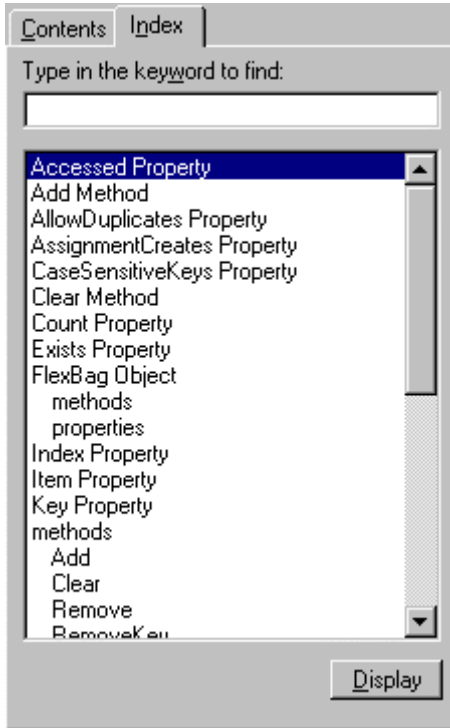
The following figure shows the Contents tab in HTML Help.



Index Defined

An index is a searchable list of keywords, each of which corresponds to one or more topics. The author of a well-designed index anticipates which words and phrases the reader is likely to look for and associates them with matching Help topics. In an online Help system, selecting a keyword displays a list of the associated topics (or the topic itself if there is only one). In a printed manual, the index at the back of the book shows the page numbers that correspond to each keyword.

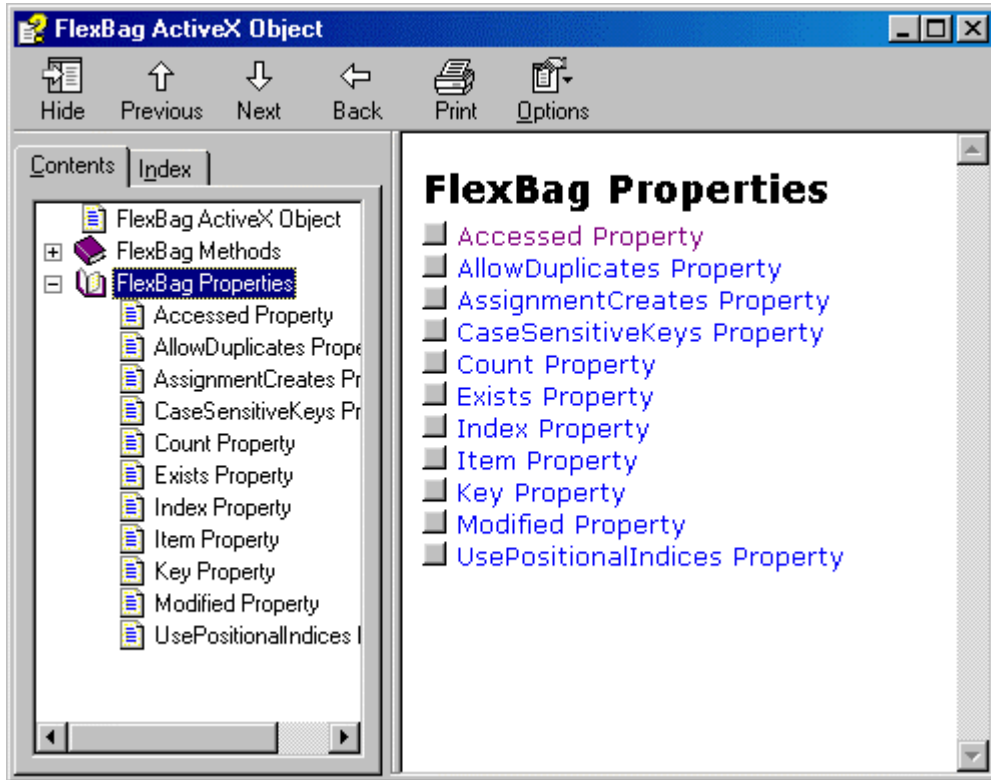
The following figure shows the Index tab in HTML Help.



Navigation Defined

The term navigation refers to a means whereby the reader can access topics sequentially in an order defined by the author. Individual topics can also contain navigational aids that provide quick access to a list of subtopics.

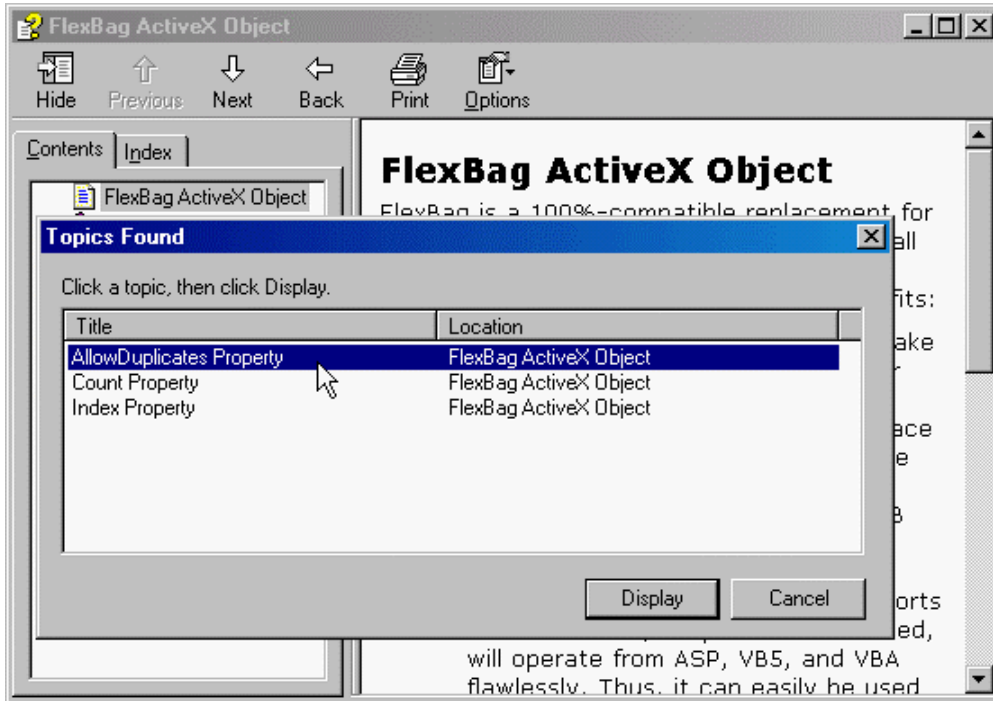
The following figure shows a HTML Help topic that provides two kinds of navigation. The Previous, Next and Back buttons on the HTML Help toolbar let the reader browse the topics. The small buttons within the topic itself let the reader jump directly to one of its subtopics.



Hypertext Defined

The term hypertext refers to interactive elements that link individual words and phrases with other topics or topic groups. A good interactive Help system does more than present concise information in a logical order; it incorporates hypertext elements such as jumps and pop-ups that guide the reader to related topics without imposing a particular sequence. Authors can also use hypertext to keep related topics accessible without presenting an overwhelming amount of information to the reader in a single topic.

The following figure shows a list of related topics displayed by a hypertext link in HTML Help.



Building Help Systems with Doc-To-Help

As an author, you implement the five fundamental elements of a Help system by correlating your source documents with objects you define in a Doc-To-Help project. This sounds more complicated than it is; if you start with a good outline in Microsoft Word, you're well on your way to producing a full-fledged Help system.

Doc-To-Help defines the elements of a Help system as follows:

- Topics** Contiguous source document regions delimited by active paragraph styles.
- Contents** A hierarchy derived from paragraph style outline levels, similar to Word's document map.
- Index** A set of associations between keywords and topics. Keywords can be generated from styles automatically, manually entered by the author, or scripted.
- Navigation** Sequential navigation is defined by the order of topics within a document. Subtopic navigation is derived from paragraph style outline levels.

Hypertext

Words or phrases, formatted with an active character style, that textually match a single topic, or a set of topics via an index element.

Understanding the Doc-To-Help Build Options

When working with Doc-To-Help 7.5, you will encounter three terms, that while somewhat similar in function, are distinct in how they are used in the context of our product.

Compile

In general, the term **Compile** is used to describe the internal process by which Doc-To-Help translates your Microsoft Word source documents into help target files.

By right-clicking on any document, you can use the **Compile** command to update an individual source document, without updating the help target itself. For example, if a build fails because a topic title is too long, you can edit the document containing the error, then compile it by itself to determine if the error has been fixed.

Make Target and Rebuild Target

The **Make Target** and **Rebuild Target** commands are similar in functionality, but different in scope. They both govern compilation – the difference lies in *what* each command actually compiles.

The **Rebuild Target** command removes all files from the output directory and recompiles your entire project, placing all new files in your output directory. This is most commonly used when you are making some sort of “global” change that affects the entire help target. For example, when adding a new term to the glossary, it is necessary for Doc-To-Help to search all source documents for instances of the new term. In this case, a **Rebuild Target** is required.

Alternately, the **Make Target** command modifies only those items (source documents, properties) that have changed since the last build. **Make Target** can be a time saver when working with a very large help project that includes many source documents because it compiles only those documents that you have modified recently.

Defining Help Topics

If you have ever authored a Windows Help file using the Microsoft Help Workshop or a third-party tool, you are no doubt familiar with the different types of topic footnotes processed by the Help compiler. For example, a \$-footnote that precedes a topic specifies the title displayed in the Topics Found dialog box, while a #-footnote defines a topic ID to be referenced within hidden text that implements a jump or popup hot spot. Since most topics also have at least one index keyword (K-footnote), it is not uncommon for a Help system to contain hundreds if not thousands of footnotes.

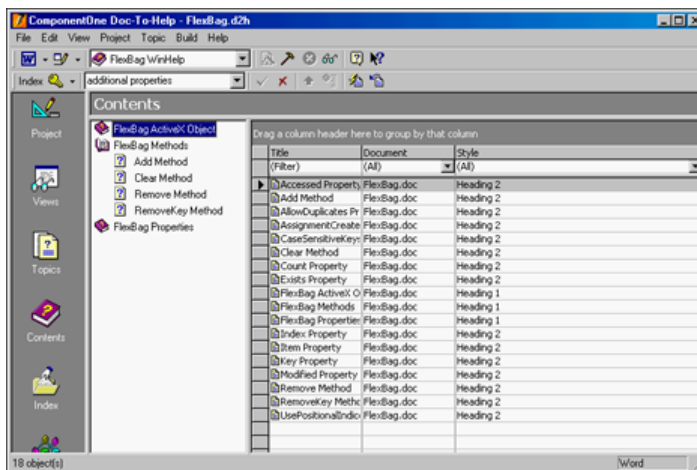
With Doc-To-Help, you define topics with paragraph styles rather than with custom footnotes. The topic consists of all text and graphics beginning with the topic title (the selected text in the figure) and ending just before the next active paragraph style (not shown). The end of file marker terminates the last topic in a source document.

A key feature of Doc-To-Help is that you define topics by applying paragraph styles to source documents – you do not have to run special commands or macros within Microsoft Word. For large projects, this can save substantial amounts of time, freeing you to concentrate your efforts on the text itself.

Organizing the Help Contents

In Windows Help, the Contents tab displays an expandable hierarchy of Help topics described in an associated contents file (with a *.cnt* extension). Most authors use the Microsoft Help Workshop or a third-party tool to build the contents file, although you could also use Notepad or any other text editor.

Doc-To-Help greatly simplifies this process by automatically generating a contents file (or HTML equivalent) based upon the outline level (1 through 9) of the paragraph styles that define topic boundaries for a Doc-To-Help project. The result mirrors the document map displayed by Microsoft Word. The following figure shows an equivalent Help contents view within the Doc-To-Help authoring environment.



For more information, see [Modifying the Help Contents](#) (page 180).

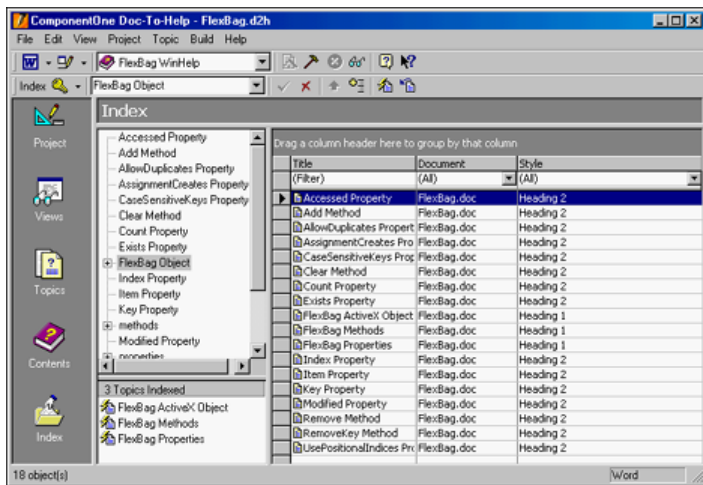
Constructing the Index

In Windows Help, the Index tab displays an alphabetized list of keywords, each of which is associated with one or more Help topics. The author makes these associations by assigning to each topic one or more K-footnotes containing the appropriate keywords.

Apart from writing the actual text, this process is widely regarded as the most tedious and painstaking aspect of Help authoring, particularly when done well.

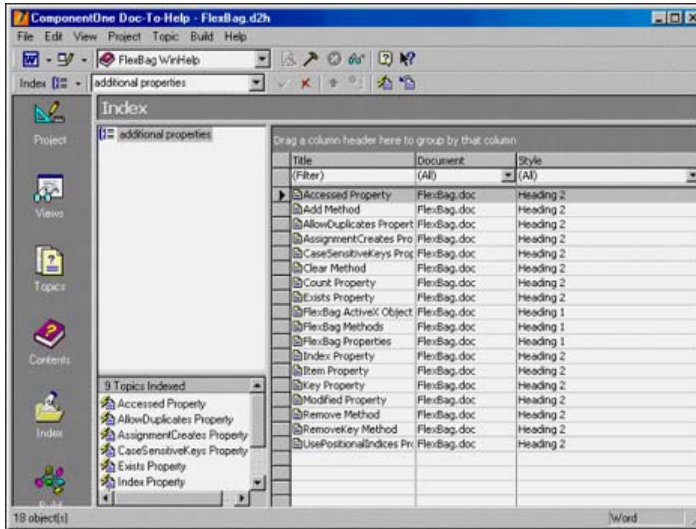
Doc-To-Help does more than just wrap a fancy interface around a tedious process. It *automates* the process by adding indexing behavior to styles, and gives you even more control by exposing an intuitive object model for scripting. For example, you can write small modules to analyze the titles of reference topics and generate several index entries for each automatically. As new topics are added, they are indexed by your script code automatically at compile time!

You can also add index keywords and associate them with topics manually using a Windows Explorer-like interface, as shown in the following figure.



In addition to index keywords that are visible to the reader, Doc-To-Help supports associative indexing via named topic groups that are hidden from the reader. If you are an experienced Windows Help author, you will recognize these as ALinks. For example, you can use groups to categorize topics according to the reader's experience level, then implement hot spots for novice, intermediate, and advanced topics.

The interface for creating and assigning groups is virtually identical to the one used to build the visible keyword index.



For more information, see [Building an Index](#) (page 219).

Specifying Navigational Elements

In Windows Help, the author must create +--footnotes to define sequence numbers that govern the behavior of the navigation buttons (<< and >>).

Doc-To-Help derives the topic navigation sequence from the physical order of topics within a source document. However, you can exclude topics defined with a particular paragraph style from the navigation sequence. For example, suppose that a source document contains a set of function descriptions at the *Heading 2* level, each of which is immediately followed by a code example at the *Heading 3* level. If each function description provides an explicit hyperlink to its code sample, and *Heading 3* is excluded from the navigation sequence, then the reader can step through the function descriptions and bypass code samples that are not of interest.

Doc-To-Help provides another form of automatic topic navigation that takes advantage of subtopic relationships defined by paragraph styles. For styles having an outline level of 2 or greater, you can opt to display a list of subtopic buttons at the end of each parent topic.

For HTML-based Help targets, Doc-To-Help provides navigation mechanisms that are functionally equivalent to those of WinHelp.

For more information, see [Providing Links to Subtopics](#) (page 172).

Linking Related Topics

In Windows Help, the author implements inter-topic links by manually assigning a unique string known as a topic ID to each destination topic, then citing the topic ID

within specially formatted text that defines a hot spot. Topic IDs are specified as #-footnotes but are referenced as hidden text. The reader never sees the topic ID; it is used only by the Help compiler as a destination for a jump (or popup) hot spot, which consists of double-underlined (or single-underlined) text followed by a topic ID formatted as hidden text. If you have ever authored a Windows Help file, you are well aware that linking topics in this manner is a time consuming and error prone process.

Doc-To-Help offers several methods for implementing hotspots. By using the **Add Topic Link** and **Add Dynamic Link** commands, you can define links by simply highlighting the text where you want a hotspot and selecting the target from a convenient dialog box. Another method of defining links is through the use of textual matches. Doc-To-Help derives topic IDs, or link tags, from topic text formatted with an active paragraph style, and also from hot spot text formatted with an active character style.

For more information, see *Links and Hot Spots* (page 197).

Note: Doc-To-Help uses the term *link tag* instead of the WinHelp term *topic ID* to describe a unique string that identifies a topic. This term was introduced to distinguish these strings from the unique numeric topic ID assigned by Doc-To-Help during compilation.

Help Targets in Doc-To-Help

ComponentOne Doc-To-Help 7.5 allows you to create a wide range of outputs, called “targets” from a single project.

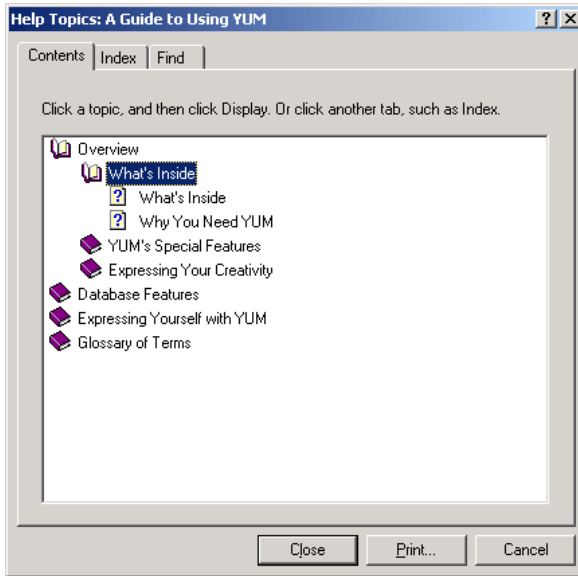
WinHelp Target

In 1995, Microsoft released WinHelp 4.0 as the online Help standard for desktop applications for the Windows operating system. It offered a sleek new look (for its day), as well as a host of navigation and access features. While WinHelp’s grip on the Help market is diminishing, its feature set provides a baseline for comparison among all the available Help targets.

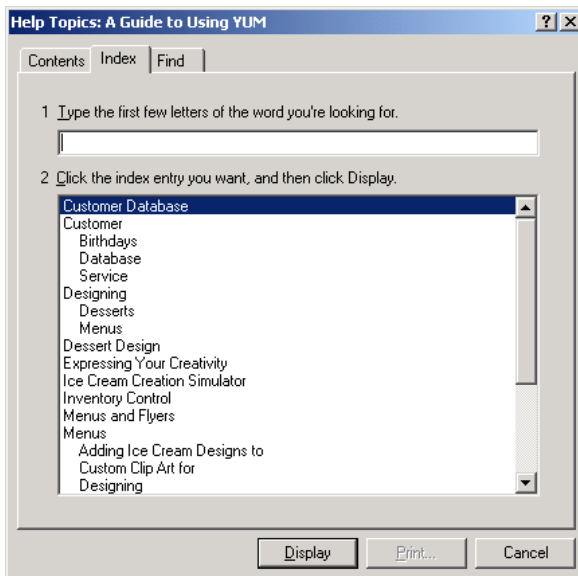
WinHelp 4.0 introduced a model where major navigational components reside separately from content. The contents, index, and search functions display in a separate, somewhat annoying modal dialog box, the Help Topics dialog box. The major weakness inherent in this display model is the lack of ease in switching back and forth between the Help Topics dialog box and the WinHelp viewer – between navigation and content.

The Help Topics dialog box supports three standard tabs: Contents, Index, and Find (full-text search).

The Contents tab replaces the Contents topic in earlier WinHelp systems, and it displays the Help contents in an expandable tree format, containing books and pages. Books divide the content into a logical structure, while pages link to specific topics.

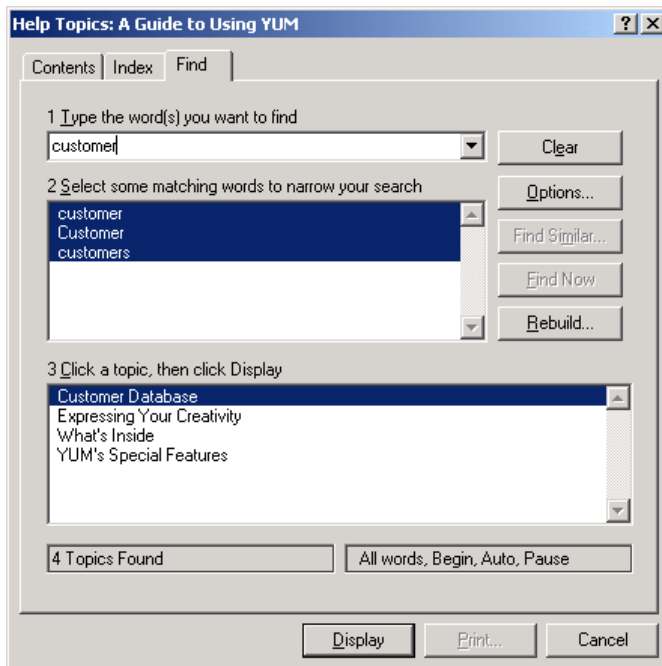


WinHelp 4.0 also introduced true second-level keyword entries. Help authors gained the functionality required to create a proper index two levels deep—an index that more closely resembles a back-of-the-book index.



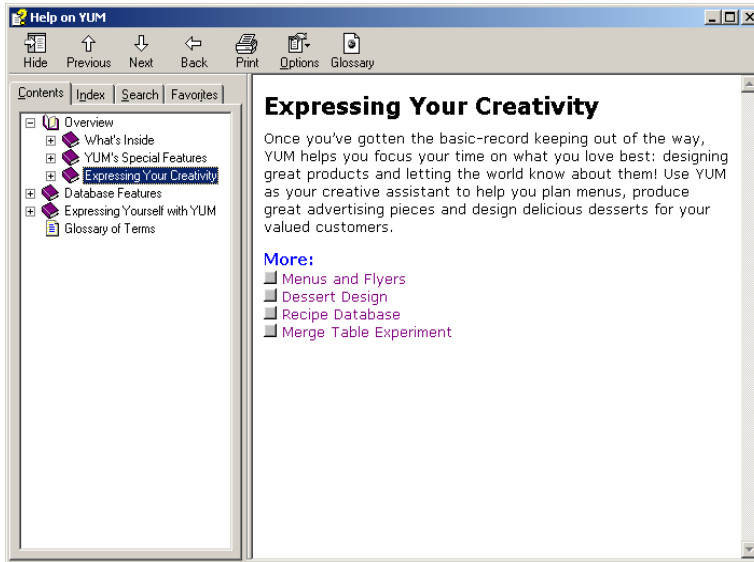
The Find tab supports full-text search capability. With no stop list to filter noise, the full-text search is a less powerful search function than a well-constructed Index. In addition, the full-text search shows every occurrence of a word, it is not selective; and it is literal. It

only shows the words that exist. In order to find something, you must know what words the author used – not what you might use.



Microsoft HTML Help 1.x Target

In 1996, Microsoft announced that HTML Help would replace WinHelp as the standard desktop online Help delivery mechanism, a radical departure from the WinHelp .rtf format. With its debut in 1997, HTML Help became the first widely accepted online Help system to use HTML as its underlying language. While it has gained widespread acceptance as a Help system for desktop applications, it cannot fulfill every need; and it certainly hasn't been able to "kill off" WinHelp six years later.



HTML Help 1.x builds on the popular feature set in WinHelp 4.0, and improves upon it, providing the author greater flexibility and additional opportunities for customization.

HTML Help introduced the tripane viewer, which combines navigation and content in one location. This tripane model has since become the standard for most HTML-based Help technologies. In addition to the familiar Contents, Index, and Search tabs, HTML Help supports a Favorites tab, allowing users to add topics for ready reference.

The second-generation, auto-synchronizing Help contents, the multi-level Index, the advanced full-text search function, and the Favorites options improve on the WinHelp model.

JavaHelp Target

JavaHelp software was developed to provide a standard Help solution for pure Java applications. JavaHelp software was released in April 1999, by Sun Microsystems, and is currently in release 2.0. While there is no “standard” viewer, JavaHelp uses components from the HotJava browser for its display. To view a JavaHelp HelpSet, the Java Run time Environment (JRE) or the Java Development Kit (JDK) must reside on the computer in addition to JavaHelp.

Doc-To-Help 7.5 has improved support for JavaHelp target. As before, JavaHelp versions starting from 1.1.3 are supported, including JavaHelp 2.0.

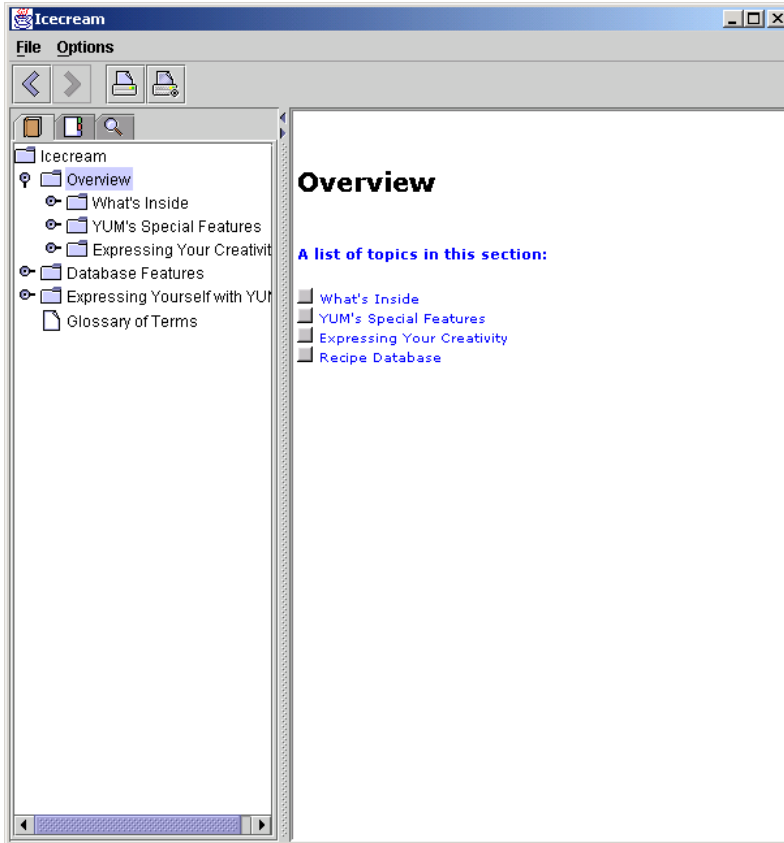
- JavaHelp secondary and pop-up windows are now supported. Topics designated for display in a secondary or pop-up window are now shown in a JavaHelp secondary or pop-up window. Previously, all topics were shown in the main window.

- This version fully supports fonts in JavaHelp. JavaHelp has known problems with font support. Therefore, help authors who used other tools or authored JavaHelp manually were usually limited to using just one default font for the entire text. Now Doc-To-Help authors can make JavaHelp display fonts properly.

Font support is controlled by three new properties in Help Target (JavaHelp): **IgnoreFontNames**, **IgnoreFontSizes** and **ScaleFontSizes**. By default, `IgnoreFontNames = True` and `IgnoreFontSizes = True`. It means that fonts specified in the source document are ignored, default JavaHelp fonts are used instead (default font for text, special default fonts for headings). This is the same behavior as in previous versions. If both **IgnoreFontNames** and **IgnoreFontSizes** properties are set to `False`, JavaHelp will use fonts exactly as they are defined in the source document. This is normally what authors want, but the matter is slightly complicated because of a known JavaHelp bug that makes all fonts appear smaller (approximately 1.3 times smaller) than they should be. Doc-To-Help provides two alternative ways of rectifying this. Setting **IgnoreFontSizes** to `True` will allow you to have consistent default JavaHelp font sizes while retaining proper font families (font names). Alternatively, you can set the **ScaleFontSizes** property to a real number other than the default 1.0 that will scale all their fonts to adjust their sizes. For example, if you see that JavaHelp displays fonts 1.3 times smaller than they should be, you can set `ScaleFontSizes` to 1.3 to fix that.

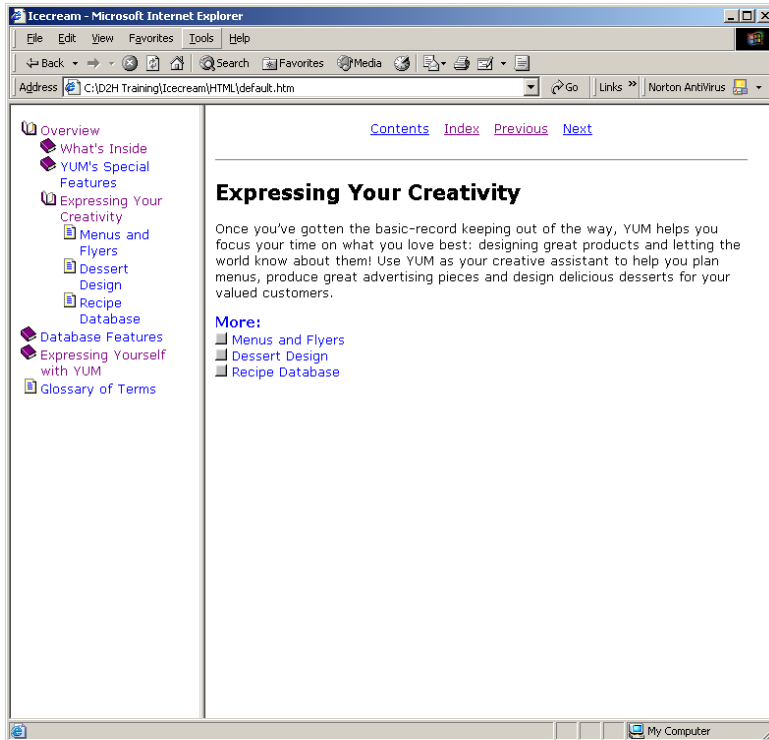
- All colors are now supported in JavaHelp themes. Previously, only basic colors were supported.
- Help window caption can now be specified using the new `Caption` property in Help Target (JavaHelp).
- Default topic, the topic that is shown when opening help, is now properly supported.
- The size of the main and secondary windows can now be specified in the Windows node of the Project tab using the `Left`, `Top`, `Width`, `Height` properties or the `Size Tool` invoked from the context menu.
- In JavaHelp themes in Theme Designer, it is now possible to hide the navigation bar (Next and Previous commands) by selecting `Position = None` in `Navigation Bar\Layout`.
- JavaHelp tabs `Contents`, `Index`, `Search`, `Favorites` can be shown/hidden by setting new properties in the Help Target (JavaHelp): `ContentsTab`, `IndexTab`, `SearchTab` and `FavoritesTab`. Tooltip strings for these tabs can be customized/localized using the properties `StringContents`, `StringIndex`, `StringSearch` and `StringFavorites`. Please note that `Favorites` tab is available only in JavaHelp 2.0; it is not supported in JavaHelp 1.1.3.

JavaHelp uses the three-pane viewer model, combining navigational components, viewer pane, and buttons in the viewer. Navigational components include contents, index, and search. Unlike the HTML Help standard viewer, the HelpSet viewer is a demo viewer, not a standard viewer. JavaHelp is fully extensible and customizable, should a programmer wish to take advantage of this flexibility. In practice, however, the HelpSet viewer has become the de facto standard viewer.



HTML 4.0 Target

Doc-To-Help 7.5 introduces its first cross-browser, cross platform solution, HTML 4.0, in this release. Currently, the HTML 4.0 target is a one-size-fits-all solution for providing online Help for Web-based applications. HTML 4.0 Help systems work across all modern browsers and operating systems.



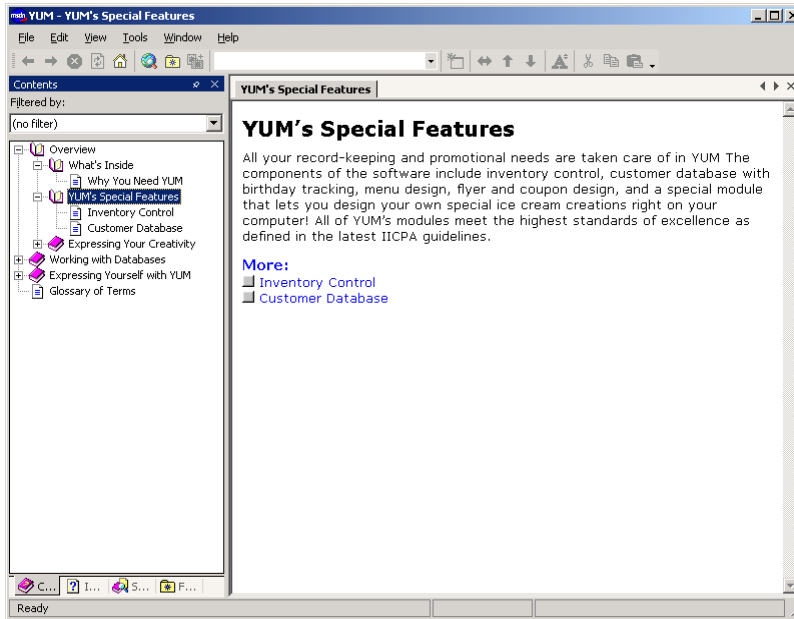
The HTML 4.0 target can be viewed using any W3C-compliant browser. The navigation conforms to the three-pane Help standard, simultaneously displaying basic navigational components in the left frame and content in the right frame. The browser buttons provide additional basic navigation.

HTML 4.0 supports two Help contents options: Dynamic – an expanding and collapsing tree structure, for Internet Explorer, and a pure HTML version for Netscape. The contents and index are accessible from each topic, and they display one at a time, in the left-hand navigation pane.

MS Help 2.0 Target

Microsoft Help 2.0 closely resembles the Microsoft MSDN collection, and uses some of the same underlying technology. Programmers who distribute .NET components can create Help 2.0, because the .NET tools include the run time components necessary to view Help 2.0 Help systems.

Note: The development of Microsoft Help 2.0 has been discontinued, though it is still the Help authoring format for .NET developers.



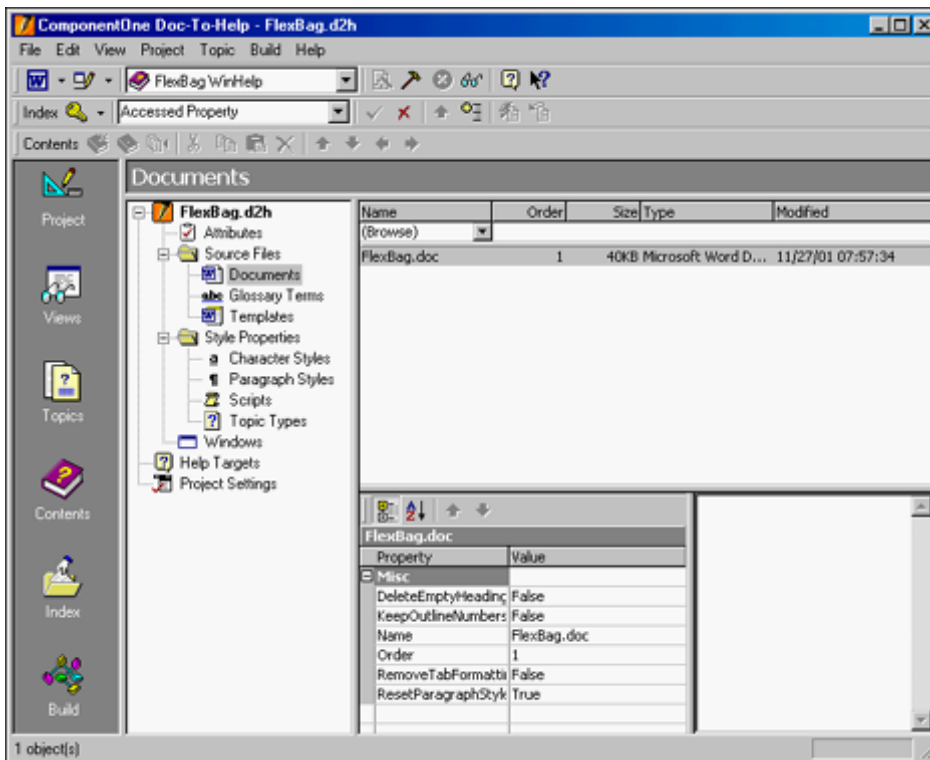
Printed Manual Target

Doc-To-Help 7.5 allows you to produce a Microsoft Word master document as a printed manual target. If unmodified, the master document includes:

- The content from your source documents.
- A default table of contents based on your heading titles.
- A default title page.
- A default index.

The Doc-To-Help Project Editor

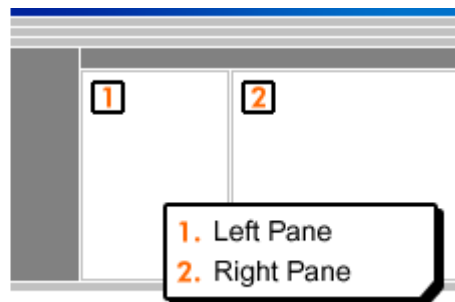
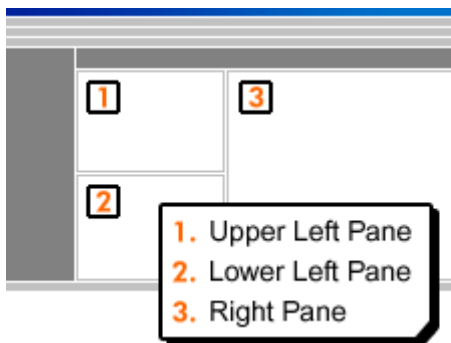
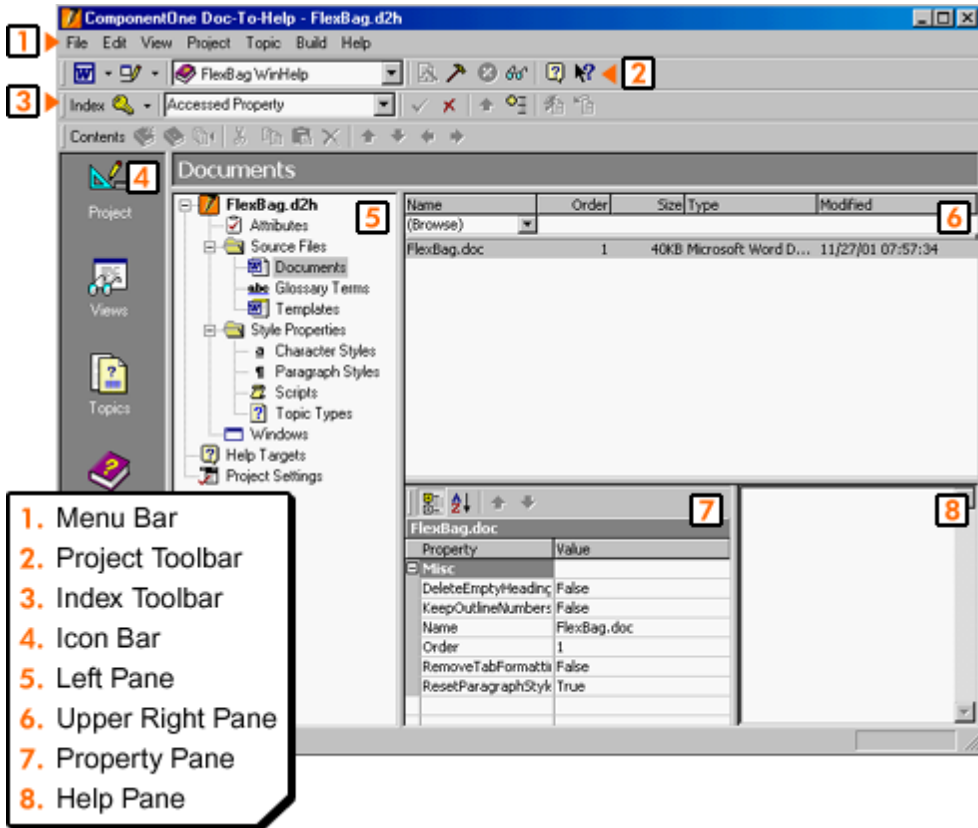
This chapter provides a description of the common operations performed with the ComponentOne Doc-To-Help 7.5 Project Editor. The Project Editor allows you the flexibility to perform a wide range of Help authoring tasks from within one simple user interface. Through the Project Editor, you can access and edit your source documents in Microsoft Word and you can produce and modify the properties of a wide range of Help target types. Additionally, with the click of a mouse you can produce a finished paper manual from your Help documentation. Along with the menu items and control buttons located at the top of the Project Editor, each of the icons located in the Outlook style Icon Bar on the left of the Project Editor allows you access to various options contained within the editor.



Navigating the Project Editor

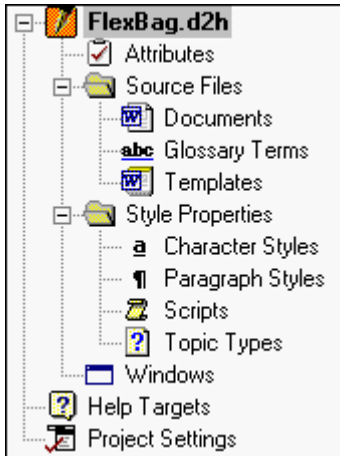
The Icon Bar, located on the left side of the project editor provides the user with a simple way to navigate through the various "views" provided with Doc-To-Help 7.5. As you

select the icons, the window "panes" to the right of the icon bar configure to display the desired controls and information. Each configuration is designed to clearly display the information provided by that icon. The naming conventions used to describe each pane in the project editor are defined in the illustrations below.



The Project Icon

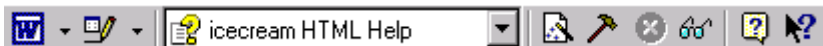
The Project Icon, as its name suggests, provides the user with an interface used to define and modify most of the features of the Help project. The left hand window pane, provides a selectable tree list of the items related to Help features, while, in most cases, the lower right hand pane provides you with information on the properties for each item.



The Project Toolbar

On the **View** menu, choose **Toolbars**, then click **Project**. A check mark next to the command indicates that the toolbar is displayed.

After a Doc-To-Help project has been compiled, you can use the project toolbar to open or edit a document, choose a Help target, compile files, build or view a target and get online Help.



From left to right, the Project toolbar contains the following commands:

Open Document	Opens the document in Microsoft Word
Edit Document	Opens the document in the right hand pane of the project editor
Choose Help Target	Specifies which Help target to build
Compile File	Compiles a specific Help document
Make Target	Builds the specified Help target
Stop Build	Stops the build in progress

View Target	Opens the currently compiled Help target
Help	Opens the Doc-To-Help Help file
What's This?	Activates context sensitive Help popups

The Project Properties Toolbar

After a Doc-To-Help project has been compiled, you can use the project properties toolbar to view the properties by property type or alphabetically.



From left to right, the Project Properties toolbar contains the following commands:

View By Property Type	Displays the project properties by type
View Alphabetically	Displays the project properties alphabetically
View Attributes	Displays the attribute choices available to Help files

Attributes

Attributes provide the Help author with the ability to create conditional text based upon specific Help files. In some instances, you may wish to include specific text in one group Help files while having alternate text included in a separate group of Help files. By creating attributes and associating the attribute values with Help targets, you can determine which targets contain which text. For example, you might create an attribute grouping named Audience that contains the values Novice, Expert and Administrator. In this way, you can modify the text contained in Help files for each audience by simply attaching the value to a given group of Help targets and formatting the conditional text with those attributes values.

By highlighting the **Attributes** item on the project editor tree list, you are presented with a list of the default attributes in the right pane. By choosing any one of the default attributes, the lower left pane displays the property settings for that attribute.

For more information, see [Conditional Text and Attributes](#) (page 299).

Documents

By highlighting the **Documents** item on the project editor tree list, you are presented with a list of the source documents currently used by the Help project in the right pane. By choosing any one of the source documents, the lower left pane displays the property settings for that document. You can modify most properties by simply clicking in the box next to the property.

Opening or editing the documents can be achieved by highlighting the **Document** item on the project editor tree and right-clicking a specific document. New documents can be added by simply clicking in the right hand document list pane.

For more information, see Documents and Templates (page 133).

Paragraph Styles

In Microsoft Word, you can use styles to apply paragraph formats (such as indentation and spacing) to give your documents a uniform appearance.

Doc-To-Help extends Word's concept of styles to include Help authoring behavior. For each paragraph style that begins a topic, you define a like-named paragraph style in the project file. By customizing the properties of Doc-To-Help styles, you define the behavior of the compiled Help system.

For more information, see Using Paragraph Styles (page 143).

Scripts

Doc-To-Help scripts are code modules written in the VBScript language that you can use to modify the behavior of paragraph and character styles during compilation. If you are familiar with event-driven languages such as Visual Basic, you can think of a script as an "event handler" for a style. Just as the Click event is fired when the user clicks a button in a Visual Basic application, a script is executed whenever Doc-To-Help encounters a topic or a hot spot defined by a scripted style.

Scripts can be added by simply clicking in the right hand script list pane. Script editing can be performed in the code text editor located to the right of the Code property.

For more information, see Scripting Techniques (page 329).

Topic Types

A topic type is a named collection of topic attributes: what window the Help topic appears in, how it's accessed, whether it gets a map number, where the heading paragraph appears, or if it appears at all. Topics are mapped to topic types by the style of their headings. The use of topic types streamlines the customizing process for the Help author, while Helping maintain consistency. Adding a topic type can be achieved by right-clicking anywhere in the topic type list pane on the right side of the project editor.

For more information, see Using Topic Types (page 154).

Windows

The Help targets use the default window, named main, to specify the size, location, appearance, and user interface presented to the reader. You can change the default characteristics by editing the properties of this window.

You can also define secondary windows for displaying certain kinds of information. For example, if a topic lists a series of steps for carrying out a procedure, you may want to provide links to other topics with illustrations or additional instructions. By associating a secondary window with the character style that implements the link, you can display multiple topics simultaneously while giving the reader control over the level of detail and the layout of the Help windows. Adding a window can be achieved by right-clicking anywhere in the window list pane on the right side of the project editor.

For more information, see *Customizing Help Windows* (page 233).

Help Targets

By highlighting the **Help Targets** item on the project editor tree list, you are presented with a list of the default Help targets in the right pane. Choosing any one of the default Help targets causes the lower left pane to display the property settings for that target. You can modify most properties by simply clicking in the box next to the property. In addition to the five default Help targets, you can create additional Help targets with the right-click of your mouse.

For more information, see *Modifying the Help Contents* (page 180).

Project Settings

By highlighting the **Project Settings** item on the project editor tree list, you are presented with the Make and View Project commands as well as links to all document files contained within your project. The lower right property pane contains the default project properties. You can modify most properties by simply clicking in the box next to the property.

For more information, see *Project Settings Properties* (page 116).

The Views Icon

The Views Icon provides the Help author with an “at a glance” examination of many of the key attributes related to the Help project. This, read only, informational window contains a left side tree view pane for choosing a topic and a right side display pane for attributes. The tree view consists of:

Attributes	Displays the Help targets by attribute and topics by attribute
Context IDs	Displays the topic context IDs
Groups	Displays the topic groups and topics attached to each topic group

Help Targets	Displays the current Help targets and attributes by each Help target
Keywords	Displays primary and secondary keywords and topics by keyword
LinkTags	Displays the LinkTags for each topic
Topics	Displays the topic attributes, attributes by topic, context IDs by topic, groups by topic and keywords by topic.

The Topics Icon

The Topics Icon displays an Explorer-like view of the Help topics derived from the source documents during compilation. It also provides an interface for setting properties of individual topics, such as index keywords and context-sensitive Help IDs. The left hand pane of the project editor contains a tree list allowing the author to view specific types of topics. You can view all topics, deleted topics or topics with and without Attributes, Context IDs, Groups and Keywords. Additionally, you can sort the topics in the right project editor pane by using the convenient outlook style grouping and filtering features provided with Doc-To-Help 7.5.

Since Doc-To-Help derives topics from your source documents, it does not provide an explicit way for you to create a new topic from scratch. Instead, you create topics indirectly by applying a paragraph style to one or more source document paragraphs in Microsoft Word, defining a like-named active paragraph style in your Doc-To-Help project if one does not already exist, then compiling the affected documents either individually or by building the Help target. After compilation, you can view the newly created topics and edit their properties in the Topics window.

For more information, see *Defining and Organizing Topics* (page 165).

The Topic Properties Toolbar

After a Doc-To-Help project has been compiled, you can use the topic properties toolbar to view the properties by property type or alphabetically, view topic attributes or insert context IDs, topic groups, keywords or link tags.



From left to right, the Project Properties toolbar contains the following commands:

View By Property Type	Displays the topic properties by type
------------------------------	---------------------------------------

View Alphabetically	Displays the topic properties alphabetically
View Attributes	Displays the topic attributes, allowing specific topics to be linked to the attribute values
Topic Collections	Displays Topic Elements
Choose Collection	Allows you to choose the collection to be modified
Element Textbox	Enter the collection element value
Add	Adds the element value
Delete	Deletes the element value

The Contents Icon

The Contents Icon displays the hierarchy of Help topics as determined by the outline level of the defining styles. This hierarchy is used to build the contents for all online Help targets. Note that the left pane contains the same books and topics as the Contents section in the compiled Help file, and the right pane displays a sorted list of topics as in the Topics window. The topics can be sorted and filtered in the same manner as in the Topic window. Doc-To-Help supports multiple topic selection by using the **Shift** and **Ctrl** keys. Simply press and hold the **Shift** or **Ctrl** keys and choose the record selector buttons to the left of the topic titles.

Doc-To-Help's Contents Window is an interface in which the final Help system's contents can be fully customized. The ability to redefine the default topic hierarchy, create new books, insert and delete topics, and perform common editing functions is available through this icon. The interface utilizes toolbars, drag-and-drop editing, and right-click shortcut menus to make the editing of contents easier and you more productive.

For more information, see [Specifying the Help Contents](#) (page 170).

The Contents Toolbar

On the **View** menu, point to **Toolbars**, then click **Contents**. A check mark next to the command indicates that the toolbar is displayed

After a Doc-To-Help project has been compiled, you can use the contents toolbar to cut, copy, paste, delete, and rearrange contents and the topics.



From left to right, the Contents toolbar contains the following commands:

Add Book	Adds a new book to the Help contents
Convert Topic	Changes the selected topic into a book
Insert Topic	Inserts the selected topic(s) from the right pane into the Help contents
Cut	Removes the selection and places it on the clipboard
Copy	Copies the selection to the clipboard
Paste	Inserts the contents of the clipboard at the insertion point, replacing any selection
Delete	Deletes the selection
Move Up	Moves the selection up within the outline
Move Down	Moves the selection down within the outline
Move Left	Moves the selection left within the outline
Move Right	Moves the selection right within the outline

The Index Icon

The Index Icon provides an interface for associating index keywords or named groups with specific Help topics. Index keywords are visible to the reader; only the author sees named groups. Note that the left upper pane contains the index keywords that the reader can see while the lower right pane shows which topics are indexed to the keywords. The right pane displays a sorted list of topics. The topics can be sorted and filtered in the same manner as in the Topic window.

For more information, see *Building an Index* (page 219).

The Index Toolbar

On the **View** menu, choose **Toolbars**, then click **Index**. A check mark next to the command indicates that the toolbar is displayed

After a Doc-To-Help project has been compiled, you can use the index toolbar to view index collections and modify the Help project index.



From left to right, the Project toolbar contains the following commands:

Index Collections	Allows you to view the groups or keywords collections.
--------------------------	--

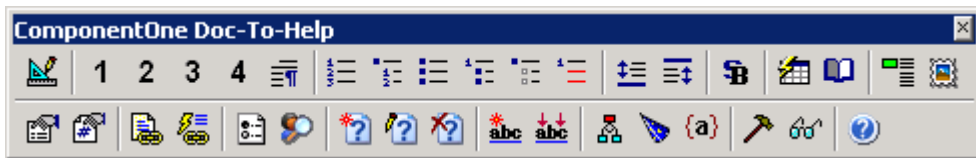
Index Elements	Indicates which index element is being viewed currently
Add	Adds an index element
Delete	Deletes and index element
Up One Level	Moves a index element up one level
Show Secondary	Shows secondary or child index elements
Index Topics	Adds the currently highlighted topic to the current index keyword
Unindex Topics	Removes the currently highlighted topic to the current index keyword

The Build Icon

The Build Icon Window lists the files generated for the active Help target as well as any Help compiler errors or unresolved links that occurred during the previous build. Debug output from scripts is also displayed within this window.

Exploring the Doc-To-Help Toolbar for Microsoft Word

Doc-To-Help installs two global templates named C1D2HAuthor.dot and C1D2HEngine.dot into the Microsoft Office startup directory. These templates add a single toolbar to Microsoft Word that allows you to edit many of the Help authoring aspects of your source documents.



Toolbar Components












Show Project Editor












Maximizes the Doc-To-Help project editor. For more information, see The Doc-To-Help Project Editor (page 39.)



Heading 1 Style

Formats the specified text with the Heading 1 paragraph style. For more information, see Working with Styles in Word (page 102).

	Heading 2 Style	Formats the specified text with the Heading 2 paragraph style. For more information, see <i>Working with Styles in Word</i> (page 102).
	Heading 3 Style	Formats the specified text with the Heading 3 paragraph style. For more information, see <i>Working with Styles in Word</i> (page 102).
	Heading 4 Style	Formats the specified text with the Heading 4 paragraph style. For more information, see <i>Working with Styles in Word</i> (page 102).
	Body Text Style	Formats the specified text with the Body Text paragraph style. For more information, see <i>Working with Styles in Word</i> (page 102).
	C1H Number Style	Formats the specified text with the C1H Number paragraph style. For more information, see <i>Working with Styles in Word</i> (page 102).
	C1H Number 2 Style	Formats the specified text with the C1H Number 2 paragraph style. For more information, see <i>Working with Styles in Word</i> (page 102).
	C1H Bullet Style	Formats the specified text with the C1H Bullet paragraph style. For more information, see <i>Working with Styles in Word</i> (page 102).
	C1H Bullet 2 Style	Formats the specified text with the C1H Bullet 2 paragraph style. For more information, see <i>Working with Styles in Word</i> (page 102).
	C1H Bullet 2A Style	Formats the specified text with the C1H Bullet 2A paragraph style. For more information, see <i>Working with Styles in Word</i> (page 102).

	C1H Continue List	Formats the specified text with the C1H Continue paragraph style. For more information, see <i>Working with Styles in Word</i> (page 102).
	Space Before	Adds or subtracts space before the current paragraph.
	Space After	Adds or subtracts space after the current paragraph.
	Special Bold	Formats the specified text with the Special Bold character style.
	Apply Default Page Setup	Automatically sets your page formatting to match the template attached to the document.
	Standard Table	Inserts the standard Doc-To-Help table at the cursor location. For more information, see <i>Inserting Standard Tables with Doc-To-Help</i> (page 253).
	Margin Note	Inserts a margin note table adjacent to the active paragraph. Creates a hyperlink between the highlighted text and a margin note. For more information, see <i>Using Margin Notes</i> (page 207).
	Image Map Editor	Opens the Image Map Editor Utility. For more information, see <i>Using the Image Map Editor</i> (page 397).
	Insert Cross-reference	Inserts a cross reference hyperlinks for online Help and page references for print manuals. For more information, see <i>Creating Cross References</i> (page 204).
	Complete Cross-reference	Completes the cross reference. For more information, see <i>Creating Cross References</i> (page 204).
	Add Topic Link	Creates a hyperlink between highlighted text and a Help topic heading. For more information, see <i>Using the Add Topic Link Dialog Box</i> (page 203).

**Add Dynamic Link**

Creates a hyperlink between the highlighted text and a index keyword or group of index keywords. For more information, see *Using Dynamic Links* (page 215).

**Apply Conditional Text**

Creates conditional text from highlighted text or graphics. For more information, see *Specifying Conditional Text* (page 299).

**View Conditional Text**

Modify the conditional text colors and determine which text is viewable in the source document. For more information, see *Using the View Conditional Text Dialog* (page 309).

**Add Topic**

Add a topic to your project without rebuilding your source documents. For more information, see *Using the Topic Tools* (page 437).

**Rename Topic**

Rename a topic to your project without rebuilding your source documents. For more information, see *Using the Topic Tools* (page 437).

**Delete Topic**

Delete a topic to your project without rebuilding your source documents. For more information, see *Using the Topic Tools* (page 437).

**Add Glossary Terms**

Inserts glossary terms into the glossary source document. For more information, see *Using the Add Glossary Terms Tool* (page 268).

**Sort Glossary**

Sorts the glossary entries in the glossary document alphabetically. For more information, see *Sorting Glossary Entries* (page 270).

**Insert Help Macro**

Inserts a WinHelp macro into your source document. For more information, see *Using WinHelp Macros* (page 273).



Insert HTML ActiveX Control

Inserts an HTML ActiveX control into your document. For more information, see Using HTML Help Object Tags (page 271).



View Field Codes

Toggles the field codes in Microsoft Word so that they are visible.



Make Target

Compiles the currently selected Help target type. For more information, see Building the Current Help Target (page 118).



View Target

Activates the currently selected Help target. For more information, see Viewing the Current Help Target (page 118).



Help

Opens the Doc-To-Help online Help.

A Guided Tour of Doc-To-Help

Welcome to the ComponentOne Doc-To-Help 7.5 Guided Tour! If you are an experienced Doc-To-Help user, this latest version includes many new design features bundled with the functionality you are familiar with from previous versions of this product. If you are a new user, Doc-To-Help provides an effortless way to create professional looking online Help from your existing Microsoft Word documents.

The purpose of this guided tour is to give you a “quick start” understanding of the features incorporated into the new version of Doc-To-Help. By working through the tutorial procedures, you will learn the fundamentals of producing printed manuals and online Help systems using Doc-To-Help.

The Doc-To-Help guided tour assumes a level of familiarity with Microsoft Word and Help authoring in general. Before beginning the guided tour, you may wish to read the sections on Help Authoring Basics (page 21)and Working with Microsoft Word (page 99). Additionally, it may be Helpful for you to familiarize yourself with the Doc-To-Help project editor and toolbar by reading The Doc-To-Help Project Editor (page 39)and Exploring the Doc-To-Help Toolbar for Microsoft Word (page 48.)

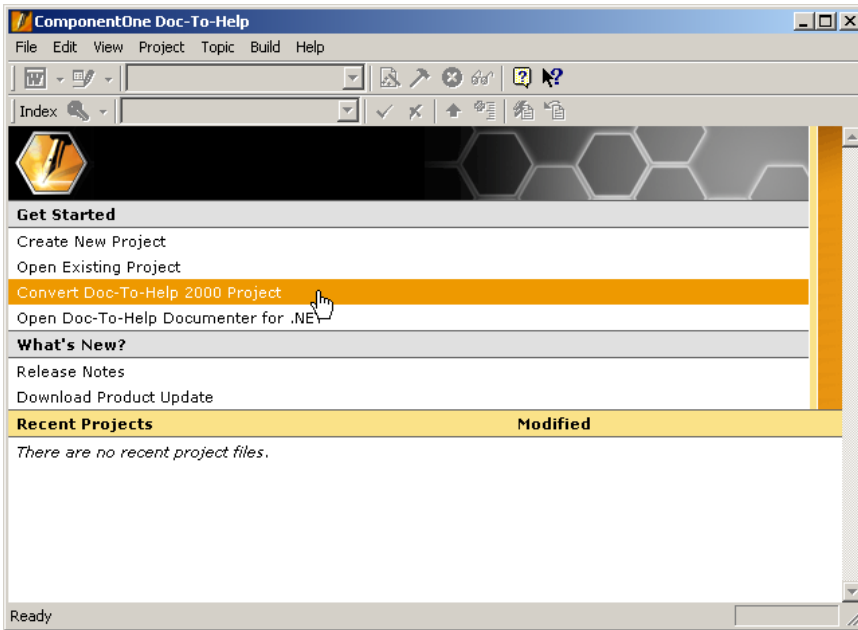
Tutorial Conventions

- Instructions to select submenu items are given in the **Menu | Submenu** format.
- Throughout the tutorial, dialog box names, control labels and controls are formatted in a bold font.
- The tutorial instructions assume that you installed Doc-To-Help in the default location. If you installed in another location, you must navigate to files in that location instead of the location specified in the instructions.
- The tutorial instructions use Microsoft Windows 2000 and Microsoft Word 2000 for all explanations. If you are using a different operating system or different version of Word, your commands and dialog boxes may vary somewhat.
- The terms, “make target”, “rebuild target”, and “compile” are used throughout the tutorials. If you are unfamiliar with how these terms are used in relation to Doc-To-Help 7.5, please read Understanding the Doc-To-Help Build Options (page 27.)

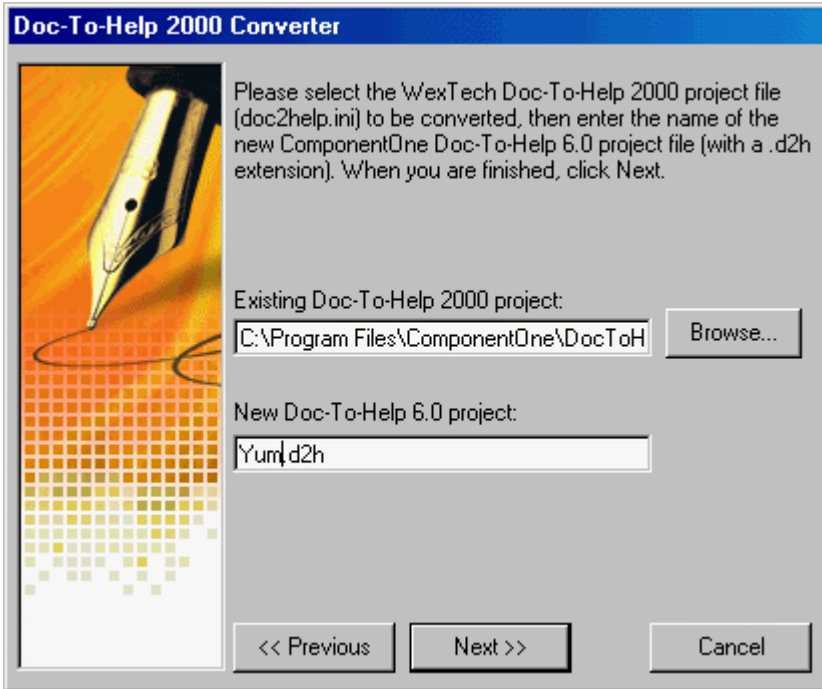
Converting Doc-To-Help 2000 Projects

Doc-To-Help 7.5 provides a simple to use utility to convert Doc-To-Help 2000 projects. For conversion, the project must be in Doc-To-Help 2000 format and the project must have been “built” (output directories must exist).

1. Open Doc-To-Help 7.5 and click **Convert Doc-To-Help 2000 Project**.



2. When the Doc-To-Help conversion dialog box opens, click **Next**.
3. Click the **Browse** button to locate the project C:\Program Files\ComponentOne\DocToHelp7\Tutorial\IceCream\DOC2HELP.INI file.
4. Select the file, then click **Open**.



5. In the **New Doc-To-Help 7.5 project** textbox, type Yum.d2h and click **Next**.
6. Review your selections and click **Finish**.

The Doc-To-Help Converter status dialog box opens to display the conversion progress.

7. Once the conversion is complete you are prompted with “Project converted successfully. Click **OK** to close this window, then press **F5** to build the current Help target”.

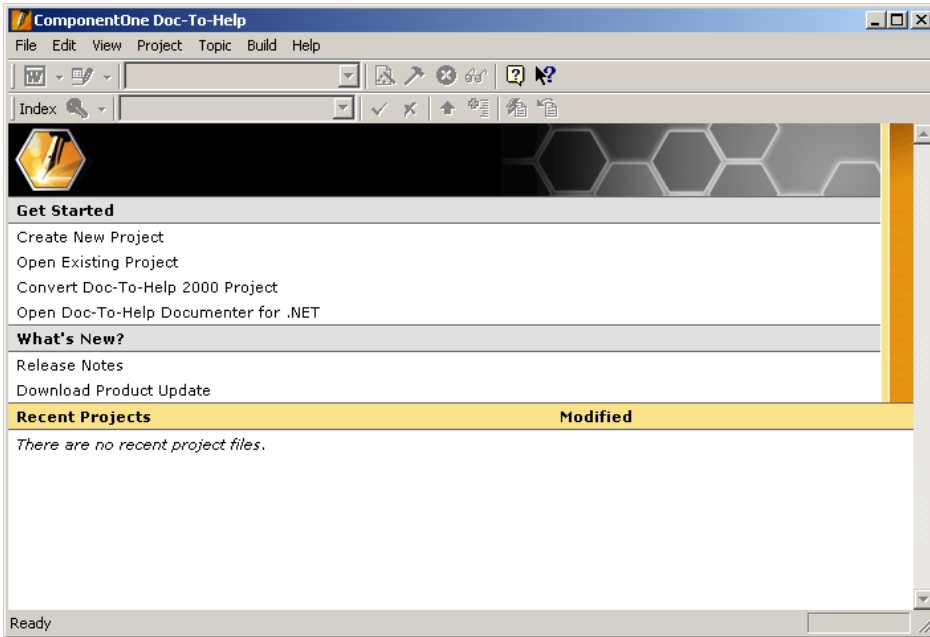
Note: A back-up of your existing Doc-To-Help 2000 project is saved in a D2HconverterBackup directory.

Starting a New Help Project

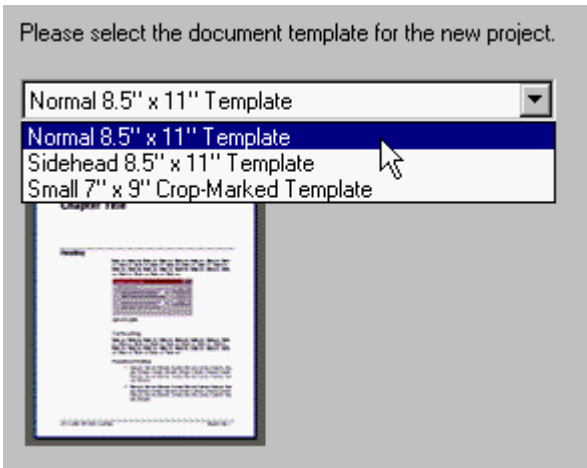
The first step in starting a new Help project is to name the project file and select a folder where it will reside. Unlike previous versions of Doc-To-Help, version 7.5 provides you with an “all in one” project editor which allows you to create and edit source documents, add Help features, and build Help files for any platform.

Open the Doc-To-Help program.

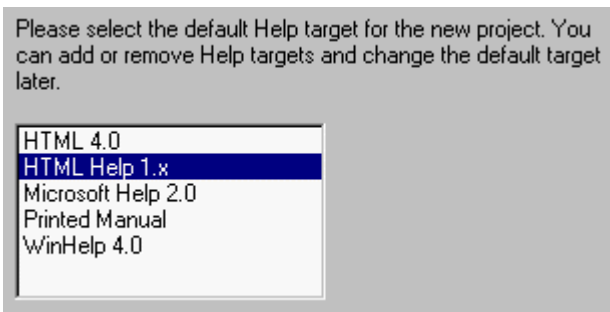
You are presented with the Doc-To-Help start page. From here you can choose some of your most recent projects, open an existing project, convert an older Doc-To-Help project or create a new project.



1. Click **Create New Project**.
2. The first screen of the New Project Wizard opens.
3. Click **Browse**.
4. The **Save New Project As** dialog box opens as shown below.
5. Open the C:\Program Files\ComponentOne\DocToHelp7\Tutorial\StyleGuide folder.
6. Type StyleGuide.d2h in the **File name** textbox, then click **Save**.
7. Click **Next**.
8. From the drop-down, choose the template that you wish to attach to your new project. In this case, make sure that **Normal 8.5" x 11" Template** is selected.



9. Click **Next**.
10. From the list, choose your default Help target. Make sure that **HTML Help 1.x** is selected.



Note: Though you can build any Help target for this project, this selection determines which Help target your project compiles initially.

11. Click **Next**.
12. Review your directory location and file name and click **Finish**.

The Doc-To-Help Wizard produces a new Help project and opens the project editor and a “start-up” document in Microsoft Word.

Working with the First Source File

You are now ready to start writing your first Help document. The default document in Microsoft Word contains a Chapter 1 placeholder where you can start entering your text.

Before working in the document, you may want to set Doc-To-Help and Word preferences to make your work easier. If you do not want to change your Word options, skip *Setting Word Preferences* and start with *Entering Text in the Document*.

Setting Word Preferences

1. Select **Tools | Options** to open the **Options** dialog box.
2. Select the **View** tab.
3. In the **Formatting marks** area, select **All** to display nonprinting characters.
4. In the **Show** area, select **HiddenText** if you want to view Word {XE} fields.
5. Click **OK**.

For more information, see *Working with Microsoft Word* (page 99).

Entering Text in the Document

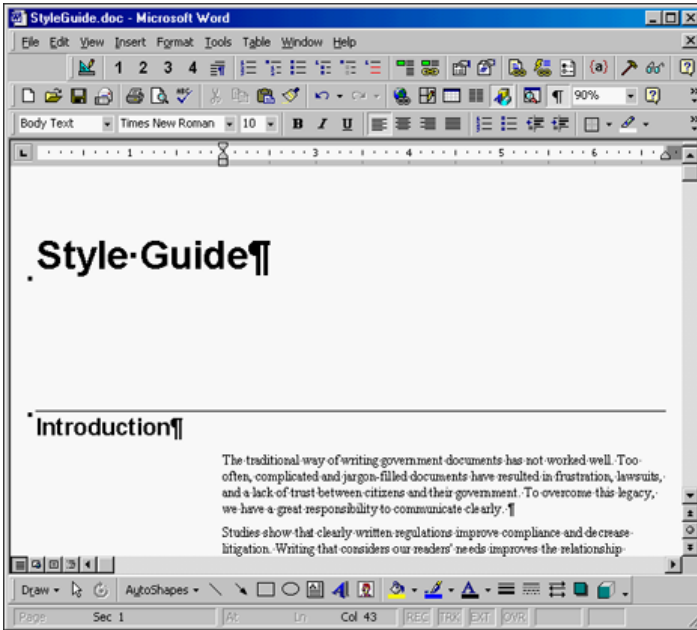
1. Highlight the **Chapter 1** placeholder.
2. Change the text to “**Style Guide**” then press **Enter**.
3. Notice that the next paragraph has been formatted as heading 2 and has a line above it.
4. Type “**Introduction**” and press **Enter**.
5. You should now have a heading 1 and a heading 2, followed by a third empty paragraph formatted as body text.

Inserting a File

It’s time to add some text to your source document, to save time, we will insert a file containing the remaining style guide documentation.

1. With your cursor in the body text paragraph, select **Insert | File**.
2. The **Insert** dialog box opens.
3. Locate the file `C:\Program Files\ComponentOne\DocToHelp7\Tutorial\StyleGuide\StyleGuideContents.doc`.
4. Select the file and click **Insert**.

Doc-To-Help inserts the text for the style guide.



Saving the Document and Building a Help File

It is time to save your first source document and build a Help file. As you will see, Doc-To-Help allows you to build multiple Help files in multiple formats using your existing source documents.

Saving the Document

- Select **File | Save**.

Doc-To-Help saves the first source document to the project folder with the same name as the project (in this case, StyleGuide.doc).

- Close the StyleGuide document.

After closing the document, you will notice that the right pane of the project editor contains options for making and viewing the Help target and links to each of the current project documents (Glossary.doc and StyleGuide.doc). An empty Glossary.doc file is automatically produced when Doc-To-Help creates your Help project. Later in the tutorial, you will learn more about using glossaries.

Building a Help File

1. If necessary, select **View Toolbars | Project** to display the Project toolbar.


2. Verify that **StyleGuide HTML Help** appears in the Project Toolbar drop-down.

3. 

4. The Project Toolbar drop-down allows you to select from a wide range of Help targets including a Manual option.

5. To build the HTML Help file, click the **Make Target** button  or the **Make Help Target** link in the right Help pane.

6. The right pane of the Doc-To-Help project editor shows the status of the Help file as it builds and compiles.

7. After the project compiles, you can examine your work by clicking the **View Target** button  on the tool bar.

8. When you are finished, close the Help file.

For more information on Help targets, see *Working with Projects* (page 107).

Using Styles to Enhance a Help Project

By customizing the properties of paragraph and character styles in Doc-To-Help, you can easily define and control the behavior of the compiled Help.

Doc-To-Help supplies templates for you to author your source document in Word and to create output target Help files. Generally, the styles contained in the templates will fill most of your needs, but you can further customize the Help behavior by adding your own styles to these template files.

Creating a new style (whether paragraph or character) for your Help system is a 3 step process:

- Create a new style in Word (which most of you are familiar with) - with properties to suit your formatting/appearance requirement for your source document.
- Create a like-named style in your output Help target template file (e.g., C1H_HTML.DOT for HTML and HTML Help output targets). This is to control how your Help output will appear/look in your Help target.
- Create a like-named style in Doc-To-Help - with properties to define the compiled Help behavior.

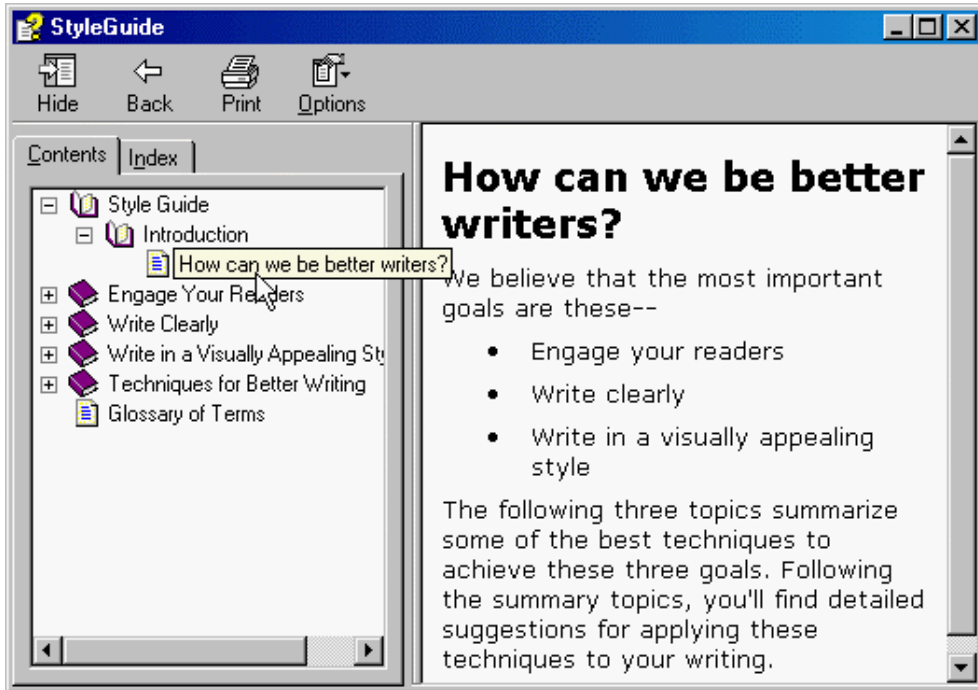
The sections below demonstrate the basic techniques to apply and create paragraph and character styles in Doc-To-Help.

Applying Paragraph Styles

The following steps guide you through the process of applying existing Doc-To-Help paragraph styles to create Help contents topics.

1. Click the **View Target** button.
2. Click the **StyleGuide** contents topic, then **Introduction**.
3. Note that the “Introduction” topic is the lowest Help level.
4. Close the Help file.
5. From the Icon Bar, select **Project**.
6. In the left pane, select **Documents**.
7. In the right pane select StyleGuide.doc, right-click and select **Open** from the shortcut menu.
8. Under the “Introduction” section, find the “How can we be better writers?” text.
9. Place your cursor anywhere in that line.
10. Apply the **Heading 3** style to that line.
11. Save and close Word and return to the Project Editor.
12. Click the **Make Target** button.
13. Click the **View Target** button.

Note that the “How can we be better writers?” text is now a topic sub-heading under the “Introduction” heading.

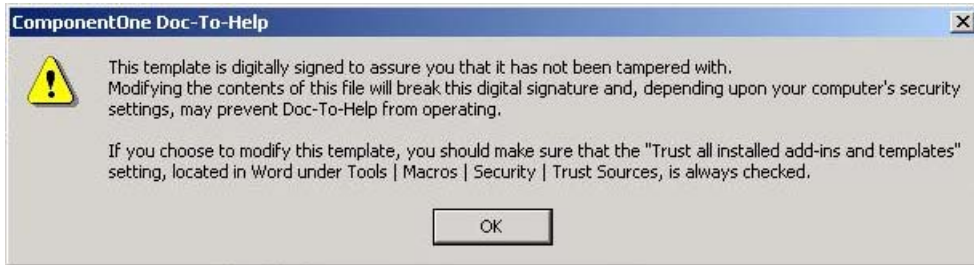


For more information on Help targets, see *Applying Paragraph Styles in Microsoft Word* (page 103).

Defining a Style in Word

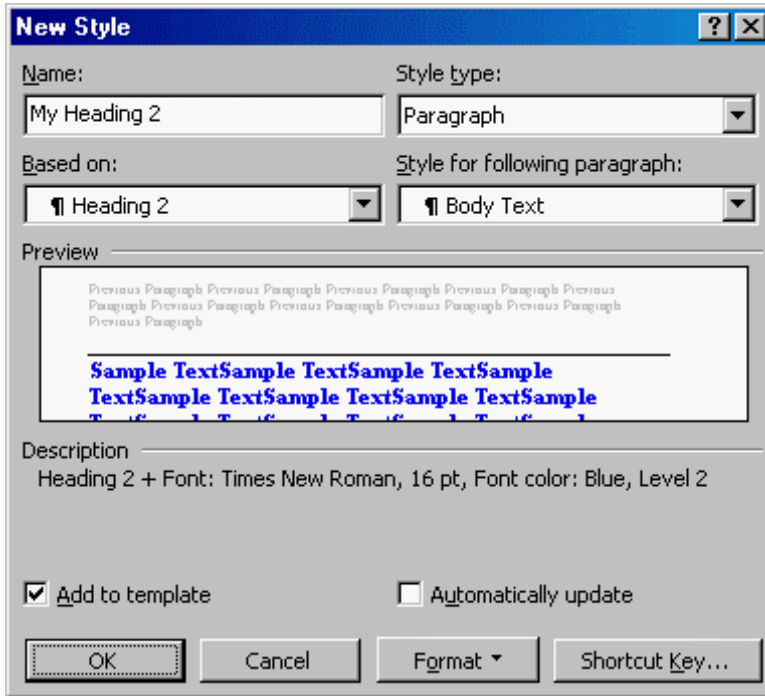
In some cases, the existing styles supplied with Doc-To-Help may not be exactly what you need. In this case, you can define a new style (paragraph or character) and add it to your “source” template.

1. In the **Icon Bar**, click **Project**.
2. In the left pane, select **Templates**.
3. In the right pane select C1H_NORM.DOT, right-click and select **Open** from the Shortcut menu.
4. This opens the C1H_NORM template that is used as the default template for formatting the source documents in Doc-To-Help.
5. Note that the following message explaining the Doc-To-Help digitally signed template displays when you open any of the preloaded templates.



6. Scroll down to the bottom of the template file and place your cursor on an empty line.
7. Select **Format | Style**.
8. The Style dialog box opens.
9. Click **New**.
10. From the **New Style** dialog box, type My Heading 2 in the **Name:** textbox.
11. From the **Based on:** dropdown, select the **Heading 2** style.
12. From the **Style for following paragraph:** dropdown, select the **Body Text** style.
13. Check the **Add to template** checkbox.
14. Click the **Format** button and select **Font**.
15. Set the font to **Times New Roman**.
16. Set the font size to **16**.
17. From the **Font Color** drop-down, select the color blue.
18. Click **OK**.

Your New Style dialog box should look like this.



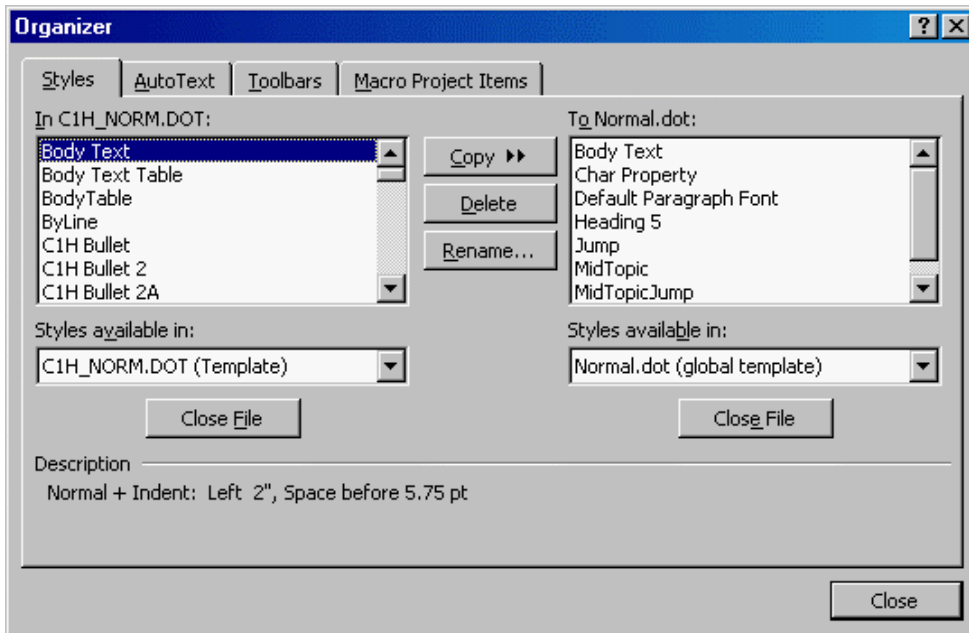
1. Click **OK**.
2. Click **Apply**.
3. Select **File | Save**.
4. At this point, do not close the template. We will start from here with our next lesson.
For more information on Help targets, see Working with Styles in Word (page 102).

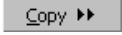
Using the Organizer to Copy a Style

After defining a new style in your “source” template, you need to add the same style to the template that your Help target will use. To do this, you will use the “Organizer” utility within Microsoft Word. In the example below our Help target is HTML Help.

1. If you have closed the C1H_NORM.DOT template, open it by following steps 1-3 in the previous lesson.
2. From the C1H_NORM.DOT template, select **Format | Style**.
3. Click the **Organizer** button.

The Organizer dialog box opens. Note that the left side of the organizer contains the styles from the currently opened template, while the right side contains the Normal.dot template.



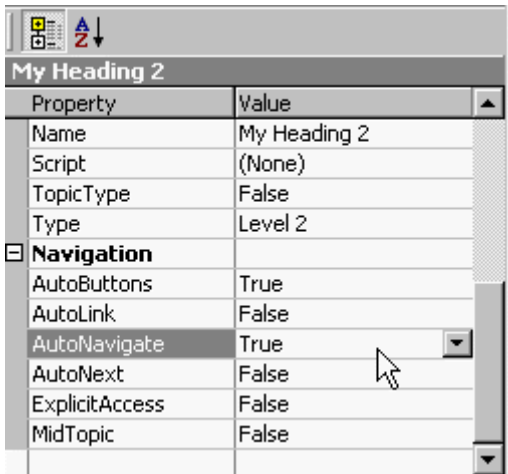
1. Close the Normal.dot by clicking **Close File** on the right side of the organizer.
2. To select the HTML Help template (C1H_HTML.DOT), click the **Open File** button on the right side of the organizer.
3. Select the C1H_HTML.DOT template from the Open dialog box and click **Open**.
4. In the **In C1H_NORM.DOT:** listbox, select **My Heading 2**.
5. Click the **Copy** button  to copy the style to C1H_HTML.DOT.
6. Click the **Close File** button for the C1H_HTML.DOT.
7. You are prompted "Do you want to save changes to C1H_HTML.DOT?", click the **Yes**.
8. Close the organizer and close the C1H_NORM.DOT file.

Defining Styles in Doc-To-Help

After creating a new paragraph style in your source template and copying it to your Help target template, the next step is to create a matching paragraph style in Doc-To-Help to control your Help system behavior.

1. If necessary, open the Doc-To-Help project editor.
2. In the **Icon Bar**, click **Project**.
3. In the left pane, select **Paragraph Styles**.

4. In the right pane, right-click and select **New**.
5. Type **My Heading 2** in the name column and press **Enter**.
6. In the property pane, change the Type property to **Level 2**.
7. To allow this topic to be included in the navigation sequence, set the AutoNavigate property to **True**.



For more information on Help targets, see [Using Styles in Doc-To-Help](#) (page 143).

Applying a User Defined Style

Applying user-defined paragraph styles is performed in the same manner as applying built-in styles. The steps below demonstrate how to apply the paragraph style we defined above in our source file.

1. In the **Icon Bar**, click **Project**.
2. In the left pane, click **Documents**.
3. In the right pane select StyleGuide.doc, right-click and select **Open** from the shortcut menu.
4. Scroll down to find the “Engage Your Readers” section, find the “How can you engage your readers?” text.
5. Place your cursor anywhere in that line.
6. Apply the **My Heading 2** style to that line.
7. Save and close Word and return to the Project Editor.
8. Click the **Make Target** button.

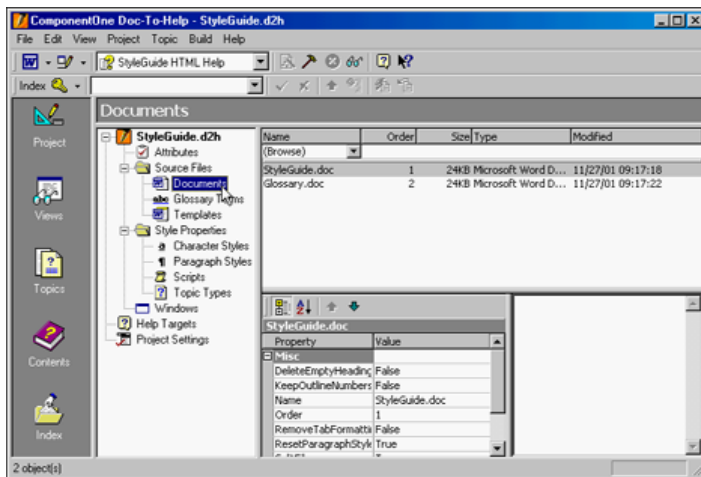
9. Click the **View Target** button.
10. Click the “Engage Your Readers” Topic.
11. Click the “How can you engage your readers?” sub-topic.
12. Note that the “How can you engage your readers?” font has changed to reflect your custom style.
13. When you’re finished, close the Help file.

Adding Multiple Source Documents to a Help Project

One of the features of Doc-To-Help is the ability to add and easily manipulate separate source documents in a Help project. One of the advantages of using separate source documents for each major “Heading 1” topic is that it provides you with the flexibility of having several authors working on the project simultaneously. Additionally, when changes are made to the documents, Doc-To-Help only needs to compile the source documents that have been modified, greatly reducing the compile time. Let’s add a second source document to our StyleGuide Help project.

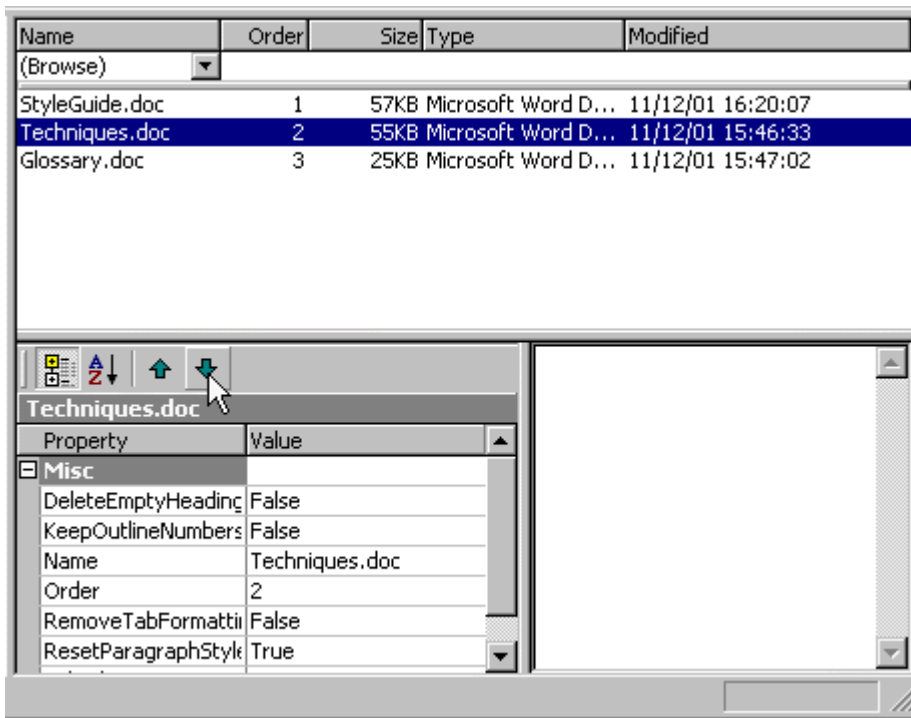
1. In the Icon Bar, click the **Project** icon.
2. In the left pane of the project editor, select **Documents**.

The right pane shows the project documents, while the properties pane shows the properties for the highlighted project document.



3. Select **Project | Add Documents**.
4. The **Add Documents** dialog box opens.

5. Locate the file C:\Program Files\ComponentOne\DocToHelp7\Tutorial\StyleGuide\Techniques.doc.
6. Select the file and click **Open**.
7. Notice that the Techniques.doc has been added to the document list in the right upper pane of project editor. The order of the documents in your Help file is noted by the order column in the right pane. To change the order of the documents, simply select the document you wish to move and change the Order property in the property pane or press the up or down arrow located in at the top of the documents property pane.



Note: When a document resides outside the project folder, Doc-To-Help displays the full path to the document. Adding a document to a project does **not** move it to the project folder. If you want the document to reside in the project folder, move it to the project folder before adding it to the project.

8. Add the new document to the Help file by clicking the **Make Target** button.
9. After the Help file is compiled, click the **View Target** button.
10. Notice that the new documentation has been added to your Help file.

11. When you are finished, close the Help file.
12. For more information on Help targets, see *Adding a New Document to a Project* (page 134).

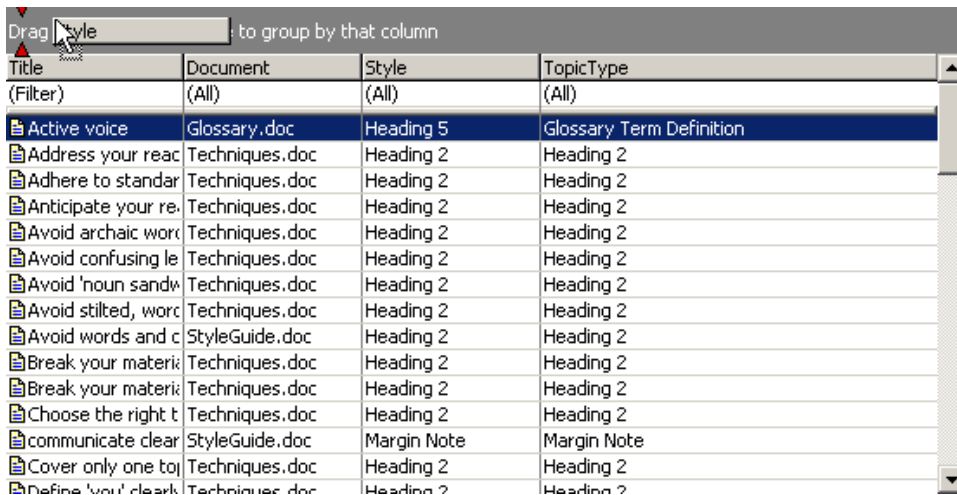
Navigating Large Projects

Large projects provide a special challenge when attempting to find a specific topic. Doc-To-Help's Topic view lets you find, sort, and open topics.

Sorting Topics For Easy Viewing

With Doc-To-Help, the Help author can modify the topic view by using the enhanced topic display and sorting capabilities in the convenient outlook style grouping feature. By simply dragging any of the topic headings (Title, Document or Style) into the grouping area located at the top of the topic list pane, Doc-To-Help sorts automatically by that heading.

1. In the Icon Bar, click the **Topics** icon.
2. To sort by style, simply select the **Style** column and while holding the mouse button down, drag its header to the column grouping area as shown below.



Title	Document	Style	TopicType
(Filter)	(All)	(All)	(All)
Active voice	Glossary.doc	Heading 5	Glossary Term Definition
Address your reac	Techniques.doc	Heading 2	Heading 2
Adhere to standar	Techniques.doc	Heading 2	Heading 2
Anticipate your re.	Techniques.doc	Heading 2	Heading 2
Avoid archaic wor	Techniques.doc	Heading 2	Heading 2
Avoid confusing le	Techniques.doc	Heading 2	Heading 2
Avoid 'noun sandw	Techniques.doc	Heading 2	Heading 2
Avoid stilted, wor	Techniques.doc	Heading 2	Heading 2
Avoid words and c	StyleGuide.doc	Heading 2	Heading 2
Break your materi	Techniques.doc	Heading 2	Heading 2
Break your materi	Techniques.doc	Heading 2	Heading 2
Choose the right t	Techniques.doc	Heading 2	Heading 2
communicate clear	StyleGuide.doc	Margin Note	Margin Note
Cover only one to	Techniques.doc	Heading 2	Heading 2
Define 'you' clea	Techniques.doc	Heading 2	Heading 2

3. Release the mouse button.

Note the change in the Doc-To-Help topic list pane. The rows are now sorted by the Style headings.

Style	Title	Document	TopicType
(All)	(Filter)	(All)	(All)
Heading 1	Engage Your Reac	StyleGuide.doc	Heading 1
	Glossary of Terms	Glossary.doc	Glossary of Terms
	Style Guide	StyleGuide.doc	Heading 1
	Techniques for Bel	Techniques.doc	Heading 1
	Write Clearly	StyleGuide.doc	Heading 1
Heading 2	Write in a Visually	StyleGuide.doc	Heading 1
	Address your reac	Techniques.doc	Heading 2
	Adhere to standar	Techniques.doc	Heading 2
	Anticipate your re.	Techniques.doc	Heading 2
	Avoid archaic worc	Techniques.doc	Heading 2
	Avoid confusing le	Techniques.doc	Heading 2
	Avoid 'noun sandw	Techniques.doc	Heading 2
	Avoid stilted, worc	Techniques.doc	Heading 2
	Avoid words and c	StyleGuide.doc	Heading 2
	Break your materi	Techniques.doc	Heading 2

4. Select the **Document** column and drag its header to the column group area.

The topics are now sorted by Style then Document as shown below.

Style	Document	Title	TopicType
(All)	(All)	(Filter)	(All)
Heading 1	Glossary.doc	Glossary of Terms	Glossary of Terms
	StyleGuide.doc	Engage Your Reac	Heading 1
		Style Guide	Heading 1
		Write Clearly	Heading 1
		Write in a Visually	Heading 1
Heading 2	Techniques.doc	Techniques for Bel	Heading 1
	StyleGuide.doc	Avoid words and c	Heading 2
		Identify your audi	Heading 2
		Include only one is	Heading 2
		Introduction	Heading 2
		Organize to meet	Heading 2
		Place words carefl	Heading 2

5. To return to the default view, drag the headers back into the header bar.



Filtering Topics

Doc-To-Help provides the author with topic filtering capabilities by using the filter boxes located beneath the headers in the topic list pane on the right. By simply choosing one or more of the drop down filtering options or by entering text in the topic textbox, you can filter the topics in any fashion you like.

1. If necessary, select the **Topics** icon to switch to topics view.
2. Click in the textbox under the Title header, type **Identify** and press **Enter**.

The Topics list displays topics that contain the word **Identify** somewhere within the title.


Drag a column header here to group by that column

Title	Document	Style	TopicType
Identify	(All)	(All)	(All)
 How do you identify passive sentenc	Techniques.doc	Heading 2	Heading 2
 Identify your audience	StyleGuide.doc	Heading 2	Heading 2

3. To filter by **Document**, click the arrow under the **Document** header and select **Techniques**.

The Topics list now only displays the topics containing the word **Identify** in the Techniques document.

Drag a column header here to group by that column

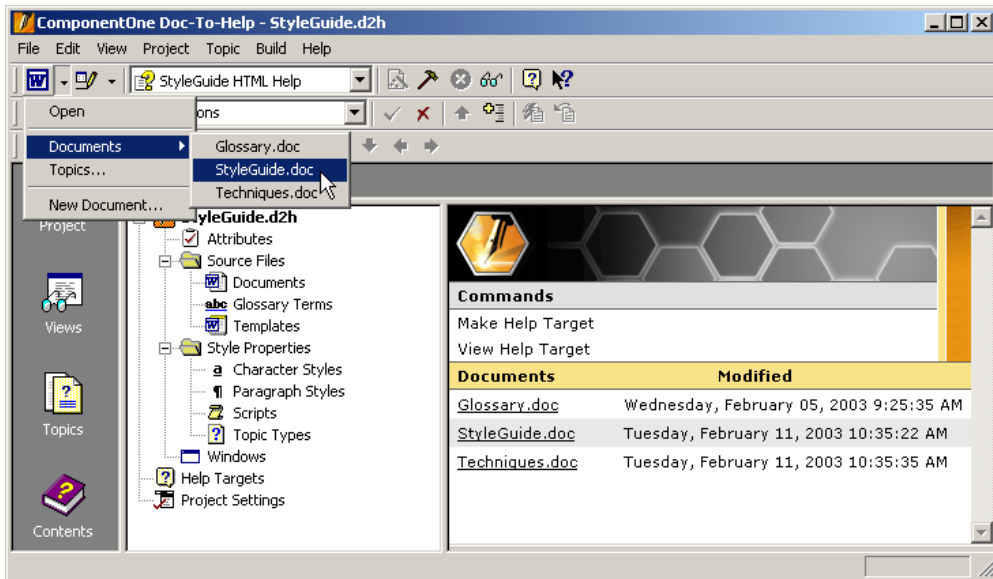
Title	Document	Style	TopicType
Identify	Techniques.doc	(All)	(All)
 How do you identify passive sentences?	Techniques.doc	Heading 2	Heading 2

By combining the Title, Document and Style filtering options, you can sort by almost any combination you like.

Adding Tables in Doc-To-Help

In many cases, it is necessary to use a table to clarify complex lists of information. With Doc-To-Help, you can use Word's table feature to include formatted tables in your documentation. In this section, you will format existing text as a table and include it in your project.

1. As an alternate method for opening documents, click the down arrow, located next to the **Open Word** icon and select **Documents | StyleGuide.doc**.



2. Scroll down the document until you locate the "Use tables" section.
3. Highlight the tab-separated text.

```

Component      →  Minimum Requirements
Processor      →  Pentium 75-MHz or greater processor
Memory         →  32-MB of RAM-Minimum
Hard-Disk Space →  10-MB hard-disk space for application files
Operating System → Microsoft Windows ME, NT (SP 3 or later), 2000
Microsoft Word → MS Word 97 (version 8.0) or greater
  
```

4. Select **Table | Convert | Text to Table**.

The **Convert Text to Table** dialog box opens.

5. Select the **AutoFit to contents option** from the dialog box.
6. Select Separate Text at Tabs.
7. Click **OK**.
8. Highlight the top row of the new table.



Component	Minimum Requirements
-----------	----------------------

9. Select Format | Borders and Shading.
10. Select the **Shading** tab and change the **Fill** to pale blue.
11. Click **OK**.
12. Save and close the StyleGuide document.
13. Click the **Make Target** button.
14. Click the **View Target** button.
15. View the table. Note that the colored table header was carried into your help file.

Use tables

Tables are used to present related information concisely. Use tables to visually display information in a way that dense text never can. The simple “System Requirements” table below presents the information very clearly:

Component	Minimum Requirements
Processor	Pentium 75 MHz or greater processor
Memory	32 MB of RAM Minimum
Hard Disk Space	10 MB hard disk space for application files
Operating System	Microsoft Windows ME, NT (SP 3 or later), 2000
Microsoft Word	MS Word 97 (version 8.0) or greater

Note: Table formatting is not available in WinHelp. Although the WinHelp compiler supports Word tables, it does not support table borders or shaded cell backgrounds. For more information on adding custom tables to a WinHelp file, see *Inserting Objects to Create Special Effects* (page 250).

16. When you're finished, close the Help file.

For more information on inserting and formatting standard tables, see *Inserting Standard Tables with Doc-To-Help* (page 253).

Adding Topic Links and Jumps


One of the keys to a good Help file is providing the reader with auxiliary information on specific topics by using links and jumps. With the click of a mouse, the reader can view additional reference materials on any number of related subjects. Use the Doc-To-Help **Add Topic Link** dialog box to create hyperlinks to topics. The steps below demonstrate how to create several jumps, including popups and jumps to separately defined windows.

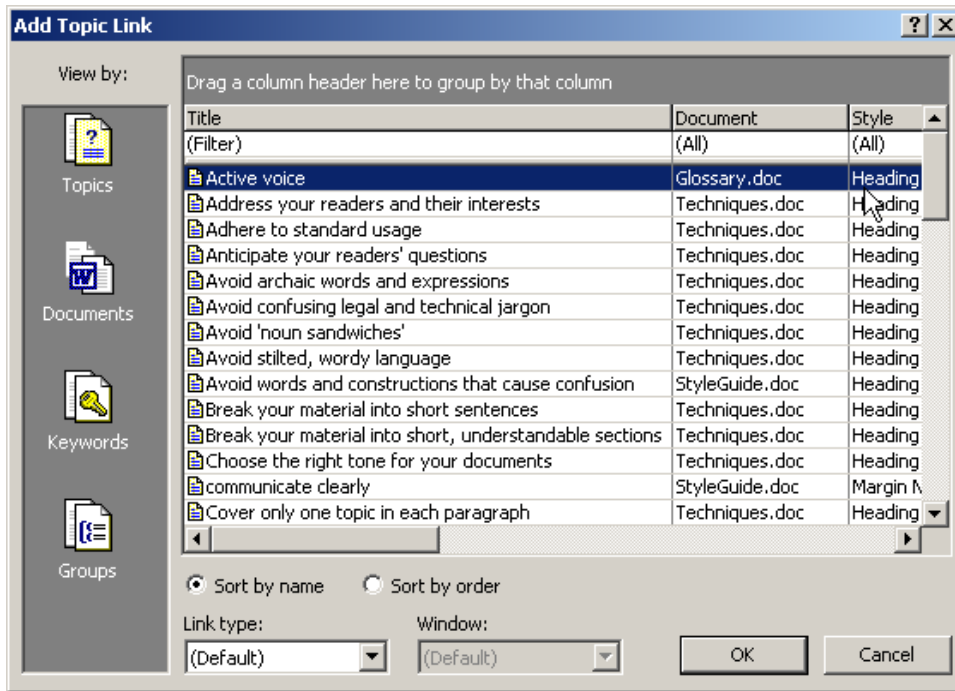
Adding a Jump

1. In the Icon Bar, click the **Topics** icon.
2. In the right pane topics list scroll down until you find the **Identify your audience** topic.
3. Select, then right-click the **Identify your audience** topic and select **Open** from the shortcut menu to open the StyleGuide.doc in Word at that topic.
4. Highlight the word "audience" as shown below.

Identify your audience { XE "audience" }

Identify your audience early and think about why the reader needs to read the document. Identify people who will be interested, even if they are not directly affected. Write to everyone who is interested, not just to technical or legal experts. Keep in mind the average reader's level of technical expertise. ¶

5. On the Doc-To-Help toolbar in Word, click the **Add Topic Link** button  to open the **Add Topic Link** dialog box.
6. In the **Title** column, select the **Address your readers and their interests** topic.
7. Select **Jump** from the **Link Type** drop-down.
8. Select **Main** from the **Window:** drop-down.
9. Click **OK**.



Note: If your hidden text is turned on, you will notice that Doc-To-Help places a [Help1] comment beside the word you had highlighted.

Adding a Popup

1. Within the StyleGuide document, locate the paragraph with the title, “How can you engage your readers?”
2. In the paragraph that follows, highlight the phrase “directly and clearly” as show below.

How can you engage your readers?¶

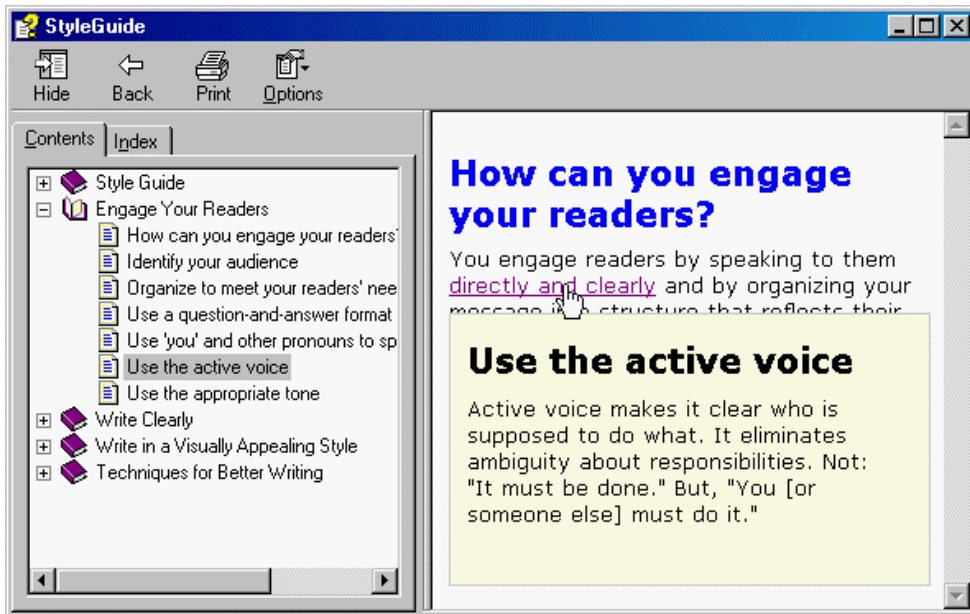
You engage readers by speaking to them **directly and clearly** and by organizing your message in a structure that reflects their interests. Here are some key ways--¶

3. Click the **Add Topic Link** button.
4. Select the **Use the active voice** topic.
5. Select **Popup** from the **Link Type** drop down and click **OK**.
6. Save the StyleGuide.doc and close Word.

Viewing the Hyperlinks

1. Restore the project editor and click the **Make Target** button.
2. Once the compile is complete, click the **View Target** button.
3. Choose the “How can you engage your readers?” topic and click the **directly and clearly** hyperlink.

A popup window displays the “Use the active voice” topic text.



4. Click outside the popup to close it.
5. In the Help contents pane, select the “Identify your audience” subtopic.
6. Click the **audience** hyperlink to open the **Address your readers and their interests** topic in the right Help pane.
7. When you’re finished, close the Help file.

For more information, see Links and Hot Spots (page 197).

Adding Graphics and Choosing a Default Topic

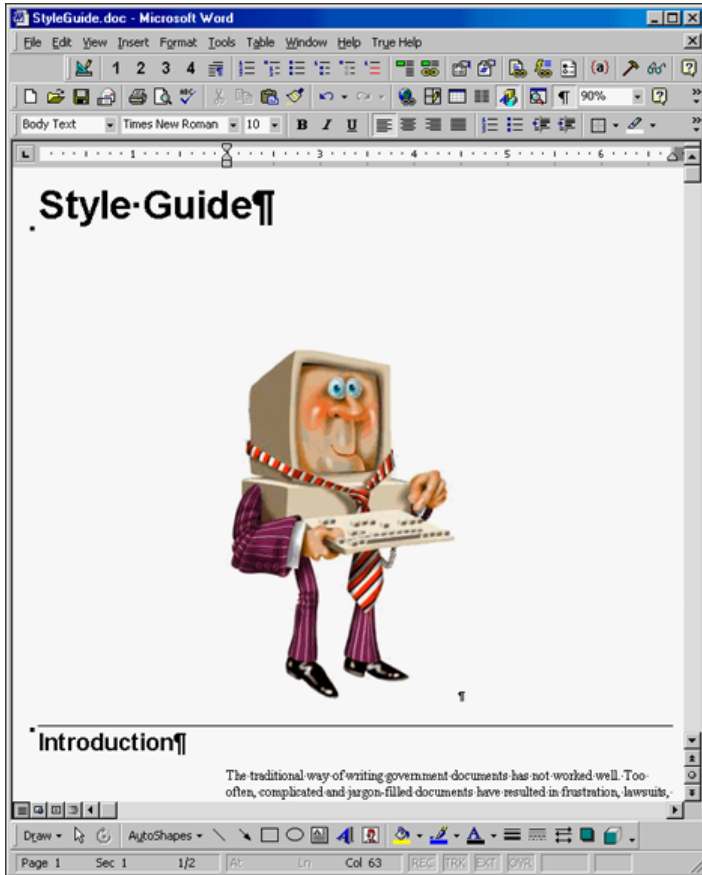
When using Doc-To-Help 7.5, it is no longer necessary to scale an image in a graphics package then insert it into the Help document. Doc-To-Help allows you to scale your graphics by using the functionality contained within Microsoft Word.

You may have noticed that we start with a blank Help window pane whenever we open the Help file. We can modify this by changing one of the topics to the **Default Topic**.

In the following steps, let's add a graphic, scale it and set our Style Guide topic as the default topic.

Inserting Graphics

1. Click the **Project** icon.
2. In the left pane, select **Documents**.
3. Select StyleGuide.doc, right-click, and select **Open** from the shortcut menu.
4. Place your cursor at the end of the Heading 1 topic "Style Guide" and press **Enter**.
5. Format the new paragraph as Body Text.
6. Select **Insert | Picture | From File**.
7. Locate the file C:\Program Files\ComponentOne\DocToHelp7\Tutorial\StyleGuide\StyleGuide.jpg.
8. Select the file, then click **Insert**.



Scaling Graphics

1. Select the graphic you just inserted, right-click, and select **Format Picture** from the shortcut menu.
2. Select the **Size** tab.
3. In the Scale area, change **Height** and **Width** to 75%.
4. Click **OK**.
5. Save and close the Word document.
6. In the Project Editor, click the **Project** icon.
7. Select **Help Targets** from the left pane.
8. Select the correct Help Target from the right pane. In the property pane, verify that the ScaleGraphics property is set to True.

9. Select **Paragraph Styles** from the left pane.
10. Select **Heading 1** from the right pane.
11. In the property pane, set the TopicType property of Heading 1 from **Contents** to **Conceptual**.

The **Contents** topic type limits text formatted with a Heading 1 style to the table of contents pane of your Help target. By selecting **Conceptual**, you can view your Heading 1 topics and the related text and graphics in the same manner as you would view any of your topics.

12. Click the **Make Target** button to compile the Help target.
13. Open the “Style Guide” topic in the Help file to view the scaled topic.
14. When you’re finished, close the Help file.

Setting the Default Topic

1. Click the **Topics** icon.
2. Select the topic **Style Guide**.
3. Right-click the topic , then select the **Default Topic** option from the shortcut menu.

Note that your selection is now displayed in a bold font to indicate that it is the default topic.

4. Click the **Make Target** button to compile the Help target.
5. After the target is compiled, click the **View Target** button.

Notice that the Style Guide topic and graphic opens in the right Help pane by default.

6. When you’re finished, close the Help file.

Using Cross-references

By utilizing Doc-To-Help’s **Insert Cross Reference** and **Complete Cross Reference** buttons, you can automatically add updateable page references in printed manuals and hyperlinks in online Help. You will use the bulleted list at the beginning of our StyleGuide document to create a few cross references.

Creating a Cross-reference


1. Click the **Project** icon.

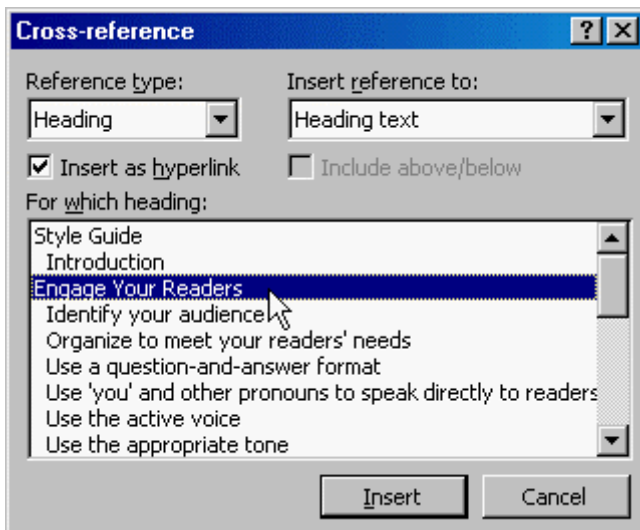
2. Select **Documents** from the left pane.
3. Select StyleGuide.doc from the right pane, right-click and select **Open** from the shortcut menu.
4. Highlight the text of the first bulleted item under the heading “How can we be better writers?”.

How can we be better writers?¶

We believe that the most important goals are these—¶


- → Engage Your Readers¶
- → Write Clearly¶
- → Write in a Visually Appealing Style¶

5. Click the **Insert Cross Reference** button  on the Doc-To-Help toolbar to open the **Cross-reference** dialog box.
7. In the **Cross Reference** dialog box, set the **Reference Type** to **Heading**.
8. Set the **Insert Reference To** drop-down to **Heading Text**.
9. In the heading list, select the **Engage Your Readers** heading and click **Insert**, then click **Close**.



Note: If you did not set the Word preference to display field codes, the text may appear the same. Doc-To-Help has actually replaced the text with a field code.

Completing Cross-references

1. If you have moved the cursor, place it immediately following the “Engage Your Readers” text.
2. Click the **Complete Cross Reference** button  on the Doc-To-Help toolbar.

Doc-To-Help inserts a page reference after the text and encloses the text in quotes. The “on page” text, added automatically by Doc-To-Help, can be modified by changing the OnPage property setting. To locate the OnPage property, click the **Project** icon, then select **Project Settings** from the left pane.

Note: You can optionally type a period at the conclusion of a completed cross-reference.

Creating More Cross-references

1. Repeat the procedures above to create and complete cross-references for the remaining bullet list items. Note that you need to match the bulleted text with the similar heading title in the **Cross-reference** dialog box.
2. Save and close the Word document.
3. In the project editor, click the **Make Target** button.
4. After the target is compiled, click the **View Target** button.
5. View the “Introduction” topic in the Help file to see how Doc-To-Help automatically converts cross-references to hyperlinks for online outputs.
6. When you’re finished, close the Help file.

For more information, see *Creating Cross References* (page 204).

Working With Margin Notes


The **Margin Note** drop down contains a variety of commands that allow you to create and format your margin notes. For this tutorial we will use the **Create Margin Note** and **Set Link** commands to produce a margin note with hyperlink.

The **Create Margin Note** command allows you to place text or graphics in the left margin, next to the main body of the text. Margin notes do not appear in the Help unless you explicitly link them to the text, in which case they become pop-ups. Text is automatically formatted to appear in Margin Note style, smaller than Body Text and italicized.

The **Set Link** command creates a hyperlink between the text or graphic in the right cell and the margin note in the left cell. If you're linking the margin note to text, that text

appears underlined and blue in Help. When the blue, underlined text or the selected graphic is clicked, the margin note displays in a popup window.

Creating a Margin Note

1. Click the **Project** icon.
2. Select **Documents** from the left pane.
3. Select StyleGuide.doc from the right pane, right-click and select **Open** from the shortcut menu.
4. Place your cursor anywhere in the first paragraph after the “Introduction” heading.
5. Click the **Margin Note** button  on the Doc-To-Help toolbar.
6. Select **Create** from the drop-down menu.

Doc-To-Help inserts a two-column table with the existing text in the right table cell.

Note: If Gridlines is selected in the **Table** menu, you'll see a dotted outline of the table, but these gridlines do not print.

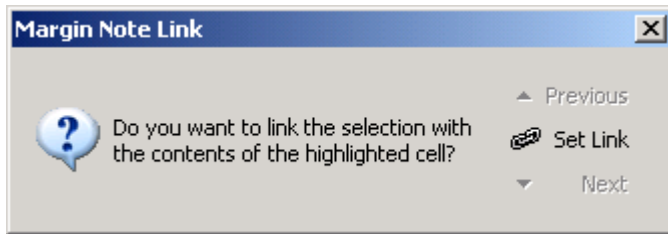
7. In the left cell, type the margin note “Heading 1 topics can now have text!”.

<i>Heading 1 topics can now have text!</i>	The traditional way of writing government documents has not worked well. Too often, complicated and jargon-filled documents have resulted in frustration, lawsuits, and a lack of trust between citizens and their government. To overcome this legacy, we have a great responsibility to communicate clearly.
--	--

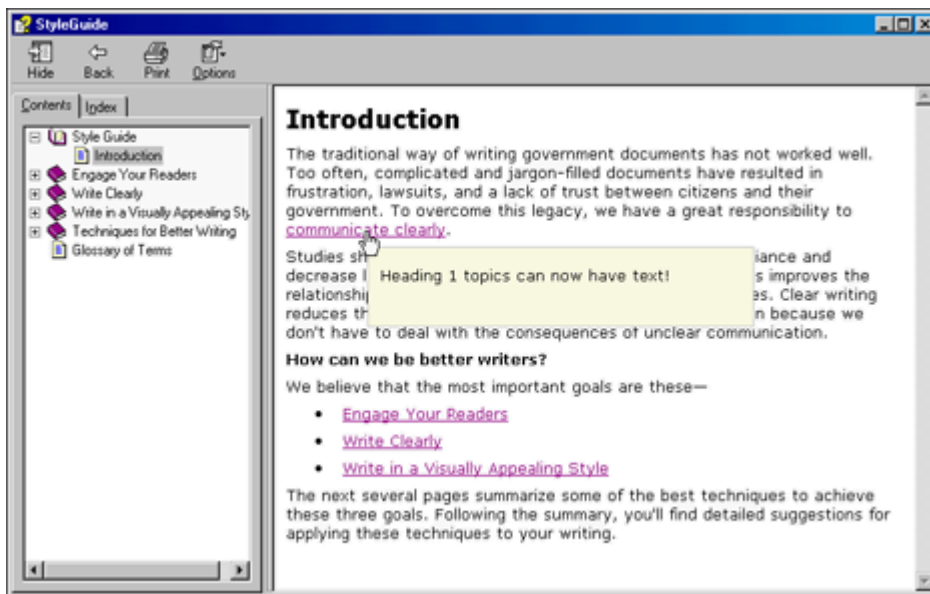
Linking a Margin Note

1. In the right cell, highlight the “communicate clearly” body text at the end of the paragraph.
2. Click the **Margin Note** button on the Doc-To-Help toolbar.
3. Select **Set Link** from the drop-down menu.

The Margin Note Link box opens. If you have more than one margin note, you can select the note you want by using the **Previous** and **Next** icons.



4. Since we only have only one margin note, click the **Set Link** icon.
Doc-To-Help places a Word comment next to the link text.
5. Save and close the Word document.
6. Click the **Make Target** button.
7. After the target is compiled, click the **View Target** button.
8. Open the “Introduction” topic, then click the communicate clearly hyperlink to see the popup.



9. When you’re finished, close the Help file.
For more information, see Using Margin Notes (page 207).

Adding Glossary Terms

To assist in developing an online and printed glossary, Doc-To-Help automatically creates an empty glossary document whenever a new project is created. During compilation, Doc-To-Help examines the project documents for text that matches the glossary entries, converting these matches into hyperlinks. Whenever the user clicks on one of the hyperlinks, a popup opens containing the corresponding glossary entry description.

By opening and adding entries to the glossary document within Doc-To-Help, you can quickly create and update your glossary as needed.

1. If necessary, click the **Project** icon.
2. Select **Documents** from the left pane.
3. Select Glossary.doc from the right pane, right-click and select **Open** from the shortcut menu.
4. Add the following text in the glossary document beneath the Glossary heading:

active·voice¶

A·style·of·writing·in·which·the·subject·of·a·sentence·performs·or·causes·the·action·expressed·by·the·verb.¶

jargon¶

A·special·vocabulary·for·a·particular·group.¶

5. Format “active voice” and “jargon” with the **Heading 5** style.

Note: The **Heading 5** style is reserved for glossary entries.

6. Format the other text with the **Definition** style.

When steps 4 – 6 are complete, the glossary text should look like the following example.

Active·voice¶

A·style·of·writing·in·which·the·subject·of·a·sentence·performs·or·causes·the·action·expressed·by·the·verb.¶

Jargon¶

A·special·vocabulary·for·a·particular·group.¶

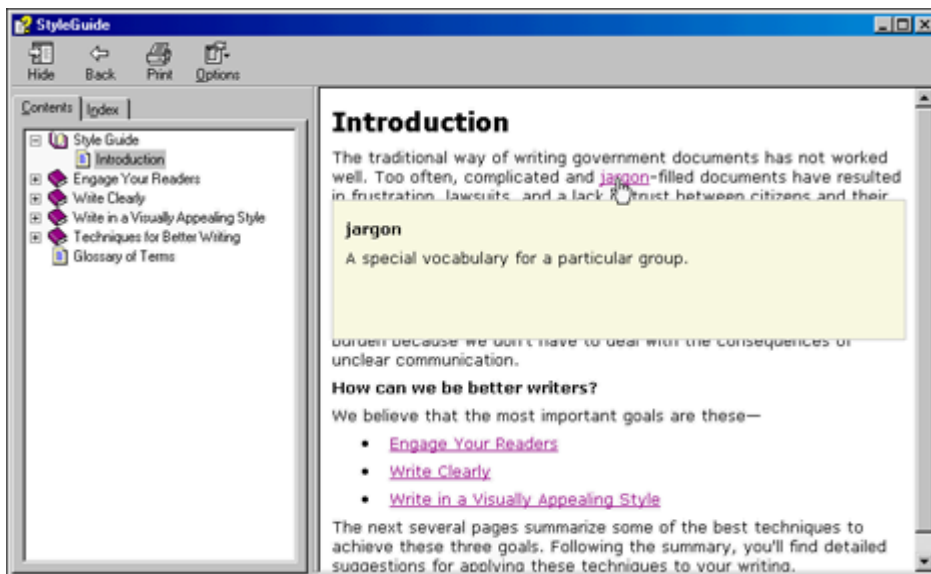
7. Save and close the Word document.

8. Restore the project editor.
9. Select **Build | Rebuild Target**.

Note: The **Make Target** command compiles only the source documents that have been modified since the last build. The **Rebuild Target** compiles all source documents. By selecting **Rebuild Target**, Doc-To-Help scans all source documents for the new glossary terms.

10. After the target is compiled, click the **View Target** button.

Wherever the Help text matches your glossary entries (Heading 5), Doc-To-Help has formatted the text as a popup.



11. When you're finished, close the Help file.

For more information, see *Modifying the Glossary* (page 266).

Adding Conditional Text

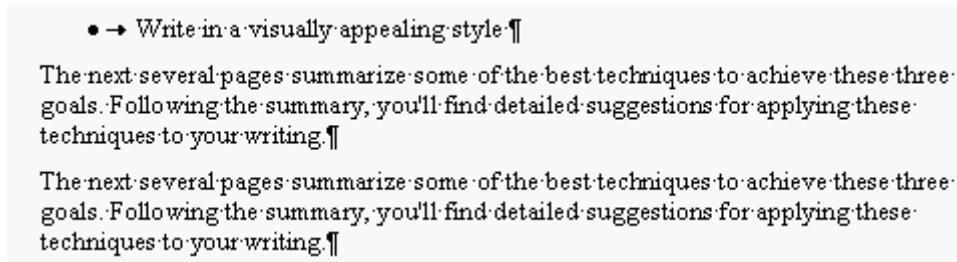
The conditional text feature allows you to determine the conditions under which certain text or graphics are incorporated into your Help targets.


When defining conditional text, Doc-To-Help provides three target options; **Platforms**, **Help Targets** and **Attributes**.

Platforms	This option allows you to incorporate conditional text into any Help file defined with a specific platform.
Help Targets	This option allows you to incorporate conditional text into a specific default or user defined Help target.
Attributes	This option allows you to incorporate conditional text into a specific version of a Help file.

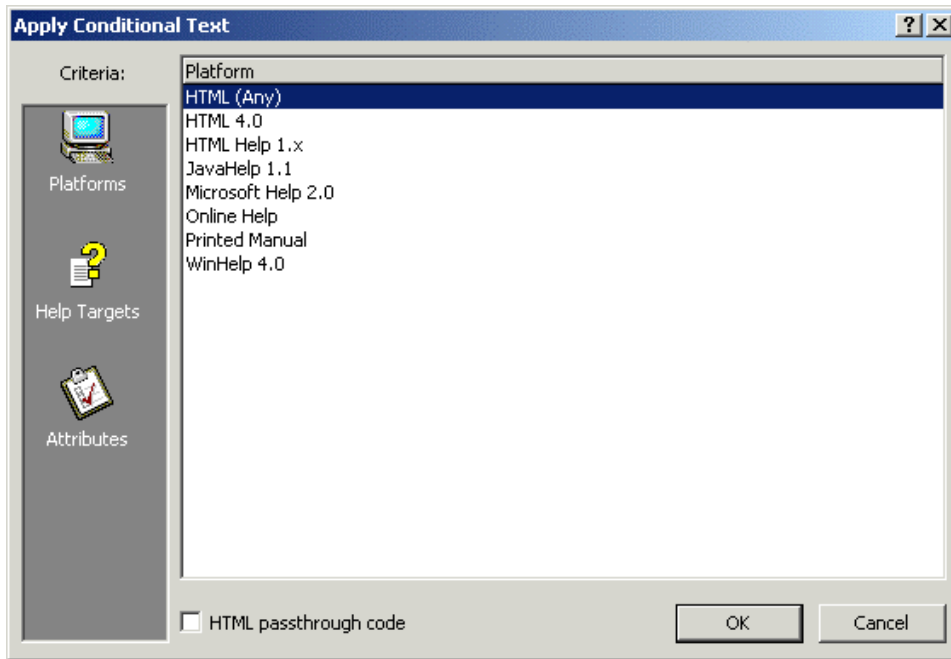
Applying Conditional Text to Printed Manuals

1. Select StyleGuide.doc from the right pane, right-click and select **Open** from the shortcut menu.
2. Copy the last paragraph of the Introduction topic and insert it beneath the existing last paragraph so that you have two duplicate paragraphs as shown below.



3. Highlight the first duplicate paragraph.
4. Click the **Apply Conditional Text** button .

The **Apply Conditional Text** dialog box opens.



5. From the **Platform** pane, select **Printed Manual** then click **OK**.

Doc-To-Help places a conditional text comment note after the first paragraph.

Applying Conditional Text to Online Help

Next, you'll change the second duplicate paragraph so that it will "fit" better as part of the online Help.

1. Change the second paragraph from:

The next several pages summarize some of the best techniques to achieve these three goals. Following the summary, you'll find detailed suggestions for applying these techniques to your writing.

To:

The following three topics summarize some of the best techniques to achieve these three goals. Following the summary topics, you'll find detailed suggestions for applying these techniques to your writing.

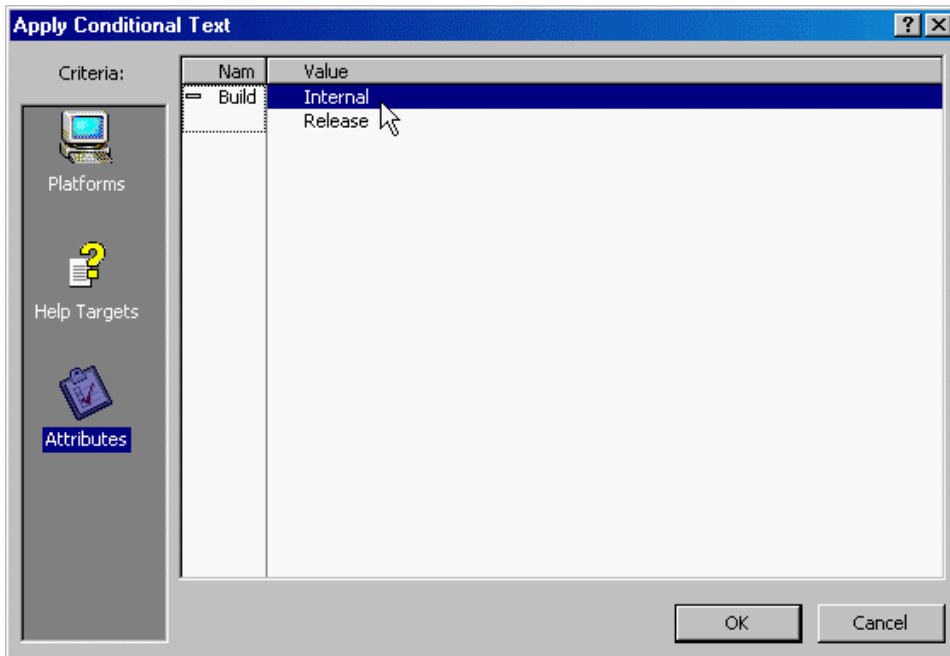
2. Highlight the modified paragraph.
3. Click the **Apply Conditional Text** button.
4. From the **Platform** pane, choose **Online Help** then click **OK**.

Doc-To-Help places another conditional text comment after the second paragraph.

Applying Conditional Text with Attributes

In some cases you may have two Help files using the same Help target, but with somewhat different text. To do this, you can set the conditional text by attribute.

1. Create an empty paragraph immediately above the “Introduction” topic.
2. Place your cursor in the empty paragraph, and type “Version History”.
3. Apply the Heading 2 style to the “Version History” text.
4. Highlight the “Version History” heading.
5. Click the Apply Conditional Text button.
6. In the Criteria: pane, click the Attributes icon.
7. From the Value pane, select Internal.

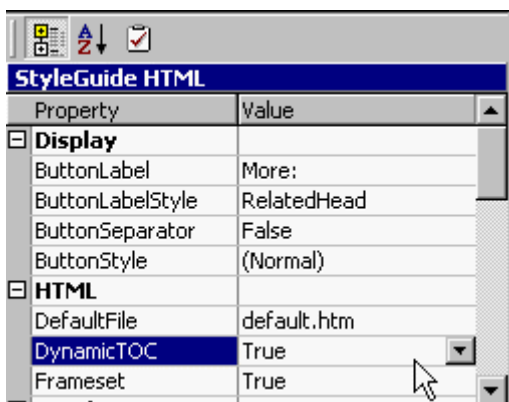


8. Click **OK**.
9. Save and close the Word document.


Building the Conditional Help

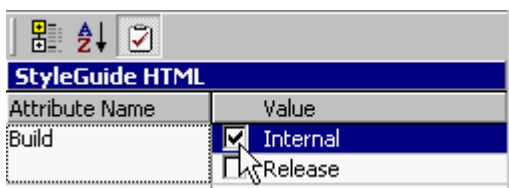
In the following steps, you will set the Help project properties and attributes, and compile your source documents for both HTML and HTML Help.

1. If necessary, click the **Project** icon.
2. Select the **Help Targets** item in the left pane.
3. Choose the StyleGuide HTML target from the right pane.
4. In the StyleGuide HTML properties pane, make sure that the DynamicTOC is set to True.



Though this property is not necessary to create the conditional text, it provides a Dynamic HTML table of contents for our HTML target. Note, this feature is only supported by Internet Explorer.

5. Click the **View Attribute** icon  in the StyleGuide HTML properties pane.
6. In the Value column, select **Internal**.



7. Click the **Make Target** button to build the HTML target.
8. After the HTML target is compiled, select StyleGuide HTML Help from the Project Toolbar drop-down.
9. Click the **Make Target** button again.

In this way, you have made sure that both the HTML and HTML Help targets have been compiled.

10. After both targets are compiled, click the **View Target** button to open the HTML Help file.

Notice that the online Help no longer contains the paragraph marked as manual or the Version History topic. If we were to build the printed manual Help target, the paragraph marked manual would be used and the paragraph marked online and the Version History topic would be omitted.

11. Close the HTML Help.
12. Select StyleGuide HTML from the Project Toolbar drop-down.
13. Click the **View Target** button. The Help file opens in your browser.

By examining the HTML Help file you will notice that the Version History topic and the paragraph marked online are included. The left pane contains a Dynamic HTML table of contents with books containing subtopics.

For more information, see Conditional Text and Attributes (page 299).


Applying Lists in Doc-To-Help

To assist in developing lists, Doc-To-Help provides six list styles to allow you to create nested lists in your documentation.

Creating a Bulleted List

1. On the Icon bar, click **Project**.
2. In the left pane, select **Documents**.
3. Select and right-click StyleGuide.doc from the right pane, and select **Open** from the shortcut menu.
4. Scroll down the document until you locate the “Use vertical lists” section.
5. Highlight the text as show below.

```
lists--||
Highlight levels of importance||
Help the reader understand the order in which things happen||
Make it easy for the reader to identify all necessary steps in a process||
Add blank space for easy reading||
```

- On the Doc-To-Help toolbar, click the **C1H Bullet Style** button .

Doc-To-Help applies a bulleted list as show below.


```
lists--¶
    •→ Highlight levels of importance¶
    •→ Help the reader understand the order in which things happen¶
    •→ Make it easy for the reader to identify all necessary steps in a process¶
    •→ Add blank space for easy reading¶
```

Nesting a Bulleted List

- Place your cursor at the end of the second bulleted line and press **Enter**.
- On the new line, type **Step one is first** and press **Enter**.
- On the second new line, type **Step two is second** and press **Enter**.
- On the third new line, type **This is how you get “nested” lists**.


You should have a bulleted list that looks like the figure below.

```
lists--¶
    •→ Highlight levels of importance¶
    •→ Help the reader understand the order in which things happen¶
    •→ Step One is first¶
    •→ Step Two is second¶
    •→ This is how you get “nested” lists¶
    •→ Make it easy for the reader to identify all necessary steps in a process¶
    •→ Add blank space for easy reading¶
```

- Highlight the lines that read “Step one is first” and “Step two is second”.
- Click the **C1H Number 2 Style** button .

Doc-To-Help applies a nested numbered list as show below.

```
lists--||  
  •→ Highlight levels of importance||  
  •→ Help the reader understand the order in which things happen||  
    1.→ Step One is first||  
    2.→ Step Two is second||  
  •→ This is how you get “nested” lists||  
  •→ Make it easy for the reader to identify all necessary steps in a process||  
  •→ Add blank space for easy reading||
```

7. Highlight the line that reads “This is how you get nested lists”.
8. Click the **Continue List** button .

Doc-To-Help applies a unnumbered list at the same level as the previous numbered list.

```
lists--||  
  •→ Highlight levels of importance||  
  •→ Help the reader understand the order in which things happen||  
    1.→ Step One is first||  
    2.→ Step Two is second||  
    This is how you get “nested” lists||  
  •→ Make it easy for the reader to identify all necessary steps in a process||  
  •→ Add blank space for easy reading||
```

9. Save and close the StyleGuide document.
10. Click the **Make Target** button.
11. After the target is compiled, click the **View Target** button.
12. Open the “Use vertical lists” topic to see your finished nested list.
13. When you’re finished, close the Help file.


For more information, see Using List Styles (page 161).

Indexing in Doc-To-Help

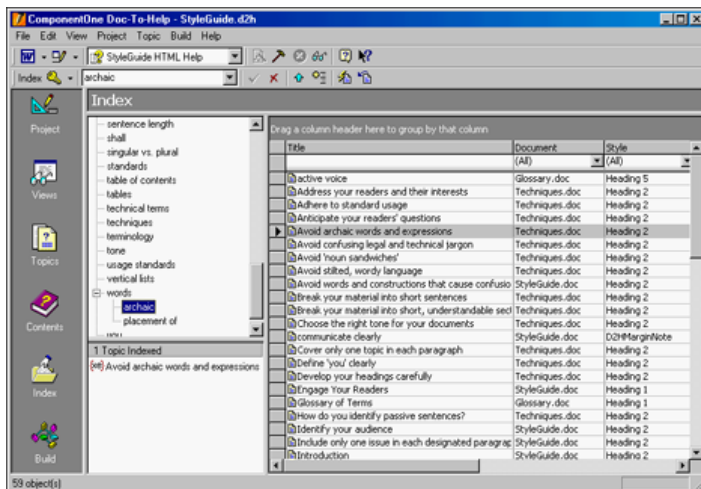
The Doc-To-Help Project Editor provides an additional interface for associating index keywords (KLinks) or named groups (ALinks) with specific Help topics. By creating

keywords and associating topics with them, **or** by choosing a topic and associating a keyword with that topic, you can develop an index for your Help file in the Project Editor. Of course, you can still use Word's {XE} fields to create index entries for print and keywords for Help targets if you prefer.

Examining the Index View

- From the Icon Bar, select the **Index** icon.
 - The upper left pane displays the index keywords and secondary keywords.
 - The lower left pane displays the topics indexed to a selected keyword.
 - The right pane shows the list of topics generated by your styles.
- In the left upper index keyword pane, scroll down to the keyword entry **words**.
- To see the secondary keywords, expand the tree view by clicking the  next to **words**.
- Select the secondary keyword, **archaic**.

The lower left hand pane contains the topics associated with the secondary keyword.



- Place your cursor between the top of the Title column and the Document column and drag the **Document** column to the **right** until it is wide enough to read the Topic Titles.

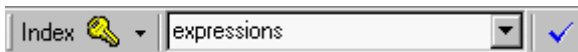
Drag a column header here to group by that column


	Title	Document	Style
	(Filter)	(All)	(All)
▶	✗ active voice	Glossary.doc	Heading 5
	📄 Address your readers and their interests	Techniques.doc	Heading 2
	📄 Adhere to standard usage	Techniques.doc	Heading 2
	📄 Anticipate your readers' questions	Techniques.doc	Heading 2

Adding Primary and Secondary Keywords

Now you will add the index keyword **expressions** and the secondary keyword **archaic** to the index keyword list.

1. Select any primary keyword in the upper left pane.
2. Select **Project | New | Keyword**.
3. Replace <new> with **expressions**.



4. Press **Enter** to insert “expressions” into the list of keywords alphabetically.
5. While the keyword “expressions” is still highlighted in the upper left pane, click the **Show Secondary** button .

Note: Clicking the **Show Secondary** button changes the options on the Index toolbar, disabling some options and enabling others. It also activates the Index Elements text box.

6. Click anywhere in the **Index Elements** drop-down.
7. Type **archaic** and press **Enter**.
Doc-To-Help adds the secondary keyword to the keyword list.

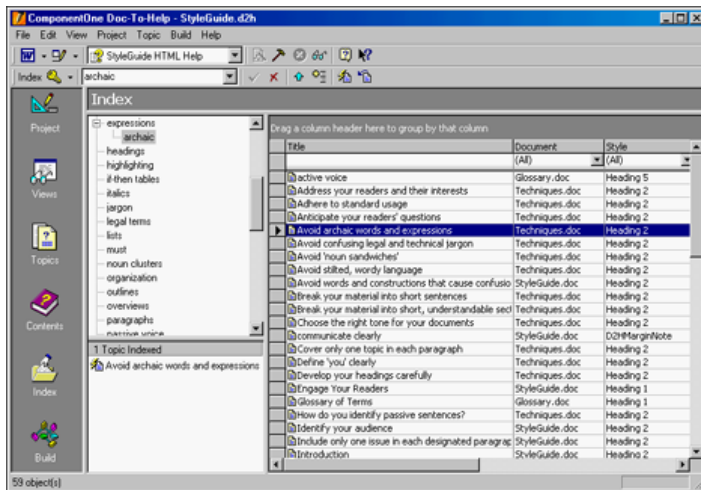


Associating Topics with a Keywords

1. Select the “expressions” keyword.
2. In the right pane, select the topic title **Avoid archaic words and expressions**.


3. Right-click the topic and select **Index** from the shortcut menu.
4. Select the secondary keyword **archaic**.
5. In the right pane, select the topic title **Avoid archaic words and expressions**.
6. Right-click the topic and select **Index** from the shortcut menu.

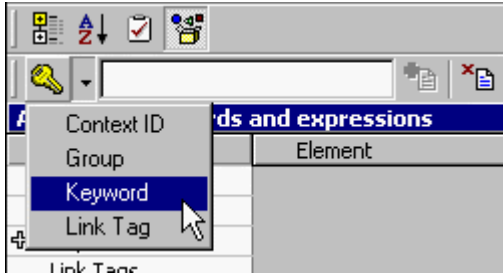
The topic has been associated with the both the primary and secondary keywords.



Associating Keywords with Topics

In the previous steps, you learned how to create an index keyword and associate it with a topic. Another method of indexing is to select a topic and associate keywords with that topic. This “topic to keyword” workflow makes it easier to create index keywords based on the existing topic titles.

1. Click the **Topics** icon.
2. Make the **Title** column as wide as necessary to view the list.
3. In the right pane, highlight **Avoid archaic words and expressions**.
4. In the property pane, click the **Collections** button .
5. Choose **Keyword** from the collections drop-down.



6. Type **archaic** into the keyword textbox and press **Enter**.
Doc-To-Help adds archaic to the list of keywords for this topic.
7. Type **archaic expressions** into the keyword textbox and press **Enter**.
8. Select StyleGuide HTML Help from the Help Target drop-down on the Project Toolbar.
9. Click the **Make Target** button.
10. After the target is compiled, click the **View Target** button.
11. Choose the **Index** tab to view your new index entries.

The secondary keyword archaic and the primary keywords expressions and archaic expressions have been added to the index and are associated with the correct topic.

12. When you're finished, close the Help file.

For more information, see Building an Index (page 219).

Creating Printed Manuals

One of the features included in Doc-To-Help is the ability to produce a "Printed Manual" in the same manner as you produce any other Help target. The compiled Printed Manual Help target includes all of the source document content, a table of contents, an index and customizable "front page". By setting the ByLine, SuperTitle and Title properties you can customize the front page of your manual with any text you like.

Examining the Printed Manual

1. Select **StyleGuide Manual** from the Project Toolbar drop-down.
2. Click the **Make Target** button.
3. Click the **View Target** button.

The front page of your printed manual contains the default ByLine, SuperTitle and Title.

4. Scroll down to see that Doc-To-Help has automatically included a table of contents.
5. Scroll down to the “Introduction” heading.

The graphic and the margin note are included in the printed manual. Note that the bulleted items under the heading “How can we be better writers?” have page numbers in place of the hyperlinks in the online Help file and the “printed manual” conditional text is included.

How can we be better writers?

We believe that the most important goals are these--

- → “Engage Your Readers” on page 3.
- → “Write Clearly” on page 5.
- → “Write in a Visually Appealing Style” on page 7.

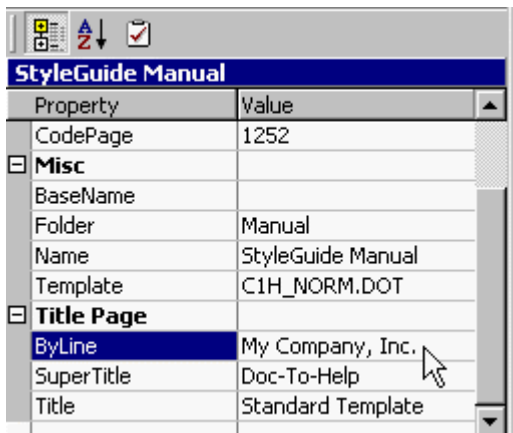
The next several pages summarize some of the best techniques to achieve these three goals. Following the summary, you'll find detailed suggestions for applying these techniques to your writing.

6. Scroll to the end of the printed manual.
Doc-To-Help has automatically included an index at the end of the printed manual.
7. When you're finished, close the printed manual.

Modifying the Front Page of the Printed Manual

The next step is to change the defaults of the ByLine, SuperTitle and Title properties to suit the StyleGuide document. By default the ByLine property is set to “By ComponentOne LLC”, the SuperTitle property is set to “Doc-To-Help”, and the Title property is set to “Standard Template”.

1. Select the **Project** Icon.
2. In the left pane, click **Help Targets**.
3. Select **StyleGuide Manual** from the right pane.
4. In the properties pane, select the ByLine Property.
5. Highlight the “By ComponentOne LLC” text and type “My Company, Inc.” (without the quotes).



6. Select the SuperTitle property.
7. Highlight the "Doc-To-Help" text and type "Becoming a Better Writer" (without the quotes).
8. Select the Title property.
9. Highlight the "Standard Template" text and type "Style Guide" (without the quotes).
10. Click the **Make Target** button.
11. Click the **View** target button.

Doc-To-Help has converted your title page to match the changes you made to the properties.

12. When you're finished, close the printed manual.

Congratulations, you have completed the Doc-To-Help Guided Tour!

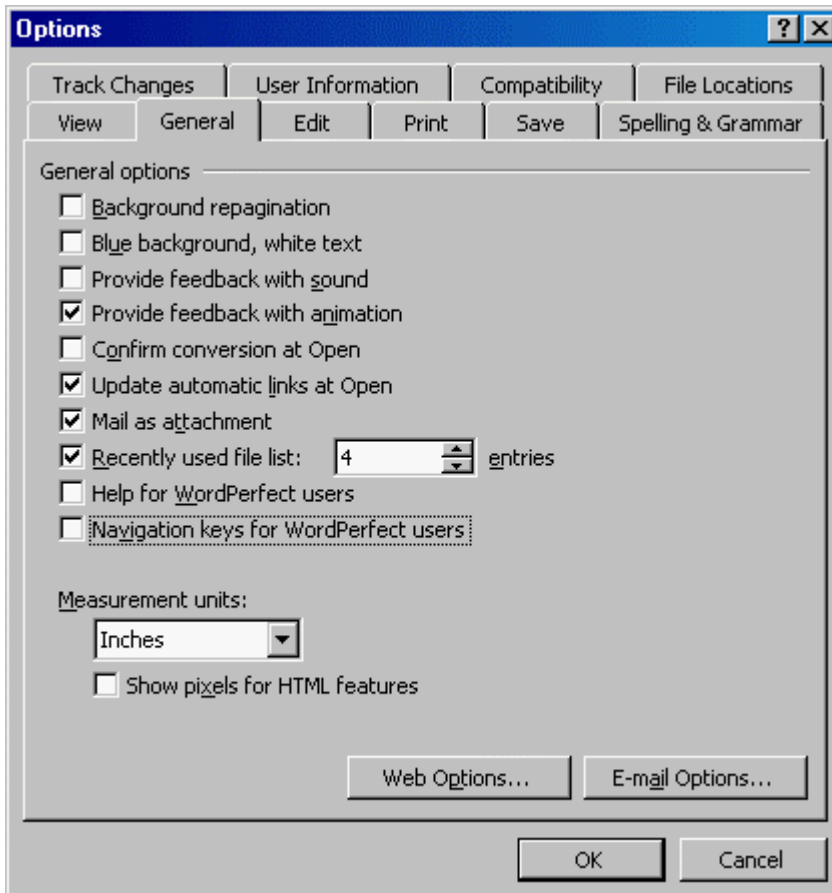
Working with Microsoft Word

One of the keys to creating well defined Help documentation with ComponentOne Doc-To-Help 7.5 is a working knowledge of Microsoft Word. In this section we will review some of the important settings in Word as well as how to work with templates and paragraph and character styles.

Let's start by reviewing some Word for Windows options to make sure that your computer is correctly configured to work on a Doc-To-Help project.

The Options Dialog Box

Most of the Word options and settings that you should check are found in the Options dialog box. Open this dialog box by clicking the **Tools** menu, and selecting **Options**. The Options dialog box contains a series of tabs that control the default options for View, Print, Edit, Save and many other settings. The General Tab is shown below.



General Tab

Please ensure that the Help For WordPerfect Users and Navigation Keys For WordPerfect Users checkboxes are clear (not selected). They interfere with many Word command keystrokes and can also make it difficult to properly apply styles. Background Repagination should also be clear because repagination can slow down Word's performance while editing large documents.

View Tab

Please select the Screen Tips option to visibly highlight text areas that have been modified by the Doc-To-Help link features.

Save Tab

Please select the **Save** tab and make sure that the following check box is clear (not selected):

Allow Fast Saves

Microsoft added this option in order to make it possible to save your documents quickly. Fast Save accomplishes this by adding your latest changes to the end of the .DOC file and referencing them at the appropriate location. When you decide to delete text or graphics, the reference is dropped, but the material is left in the file. In large documents, this means that the size of your file can keep increasing even if you delete material. Many Word users have reported that file corruption occurs much more easily when this feature is selected.

Word 2000 and Infrequently Used Menu Items

To streamline the appearance of your menus, Word 2000 displays only the most frequently used menu items, by default. If you're using Doc-To-Help and Word 2000, you will find that certain menu items referred to in this chapter are not immediately available. Please follow the two steps below to make additional menu items available to you.

Note: The streamlining discussed above also extends to the display of toolbar icons. If certain toolbar icons do not appear at first, click the More Buttons chevron (>>) on the toolbar to display hidden, yet available icons.

Modifying Word 2000 Default Settings to Display All Menu Items

- Open the **Tools** menu.
- Choose **Customize**.

The **Customize** dialog box opens.

- Clear the **Menus show recently used command first** checkbox under the Options tab, then click **Close**.

Note: Should you choose not to disable this Word 2000 feature, you'll find it is still possible to display infrequently used menu items by allowing your cursor to "hover" over the menu list.

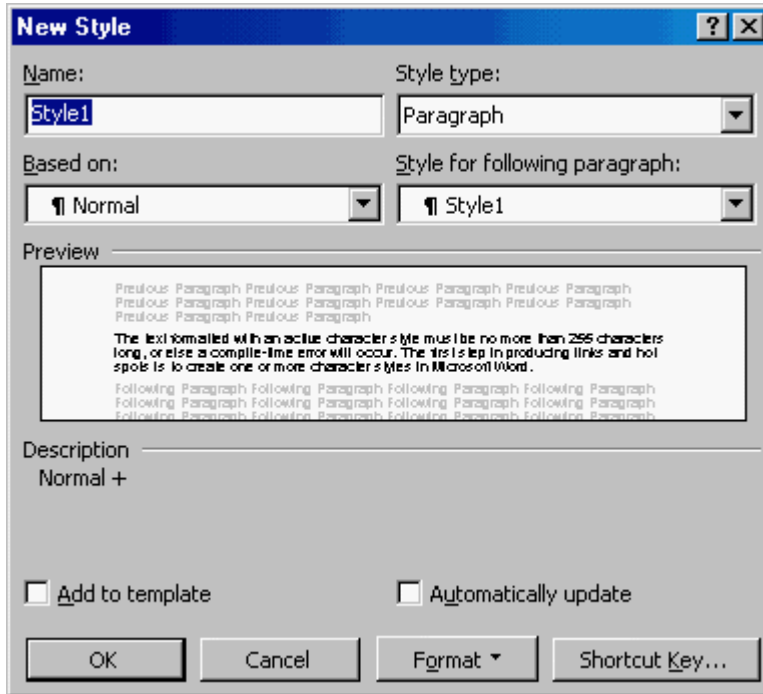
Working With Styles in Word

In Microsoft Word, you can use styles to apply paragraph formats (such as indentation and spacing) and character formats (such as font styles and colors) to give your documents a uniform appearance. If you apply styles consistently, any change you make to a style definition is propagated throughout the document, and the affected regions are updated accordingly. Styles are also used in Word's outline view to specify the hierarchy of paragraphs within a document.

Doc-to-Help extends Word's concept of styles to include Help authoring behavior. For each paragraph style that begins a topic, you define a like-named paragraph style in the project file. Similarly, for each character style that references a topic, you define a like-named character style in the project file. By customizing the properties of Doc-To-Help styles, you define the behavior of the compiled Help system.

Creating Styles in Word

- From the **Format** menu, choose **Style**.
The **Style** dialog box opens.
- Click **New**.
The **New Style** dialog box opens.

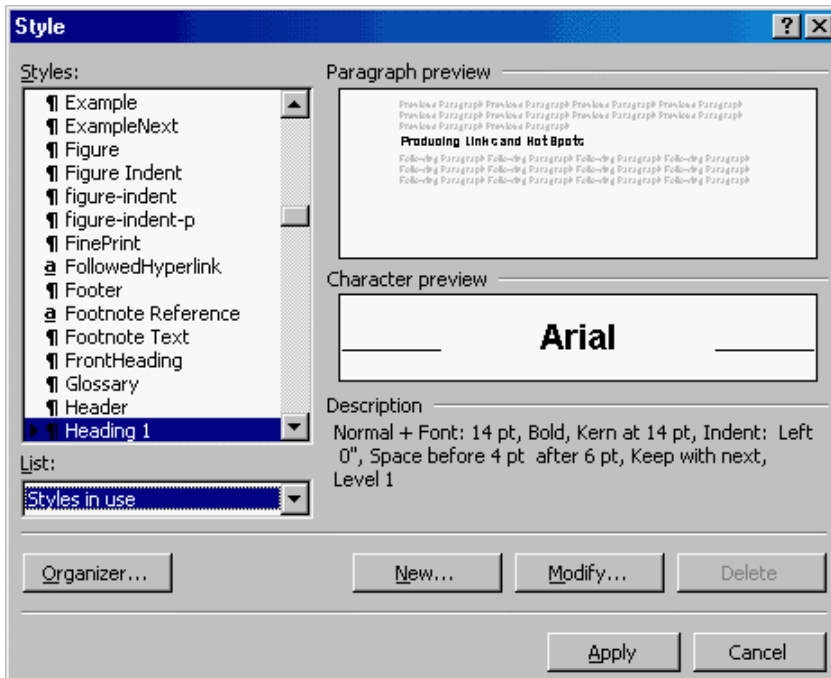


- In the **Name** box, type the name of the new style.
- In the **Style type** box, select **Paragraph** or **Character**, depending upon the type of style you want to create.
- If desired, change the other options on this dialog box, then click **Format** to change the visual attributes of the new style.
- When you are finished, click **OK**.

Applying Paragraph Styles in Microsoft Word

- Click anywhere within the paragraph to be formatted. You do not need to select the entire paragraph, although doing so will also work.
- From the **Format** menu, choose **Style**.

The **Style** dialog box opens.



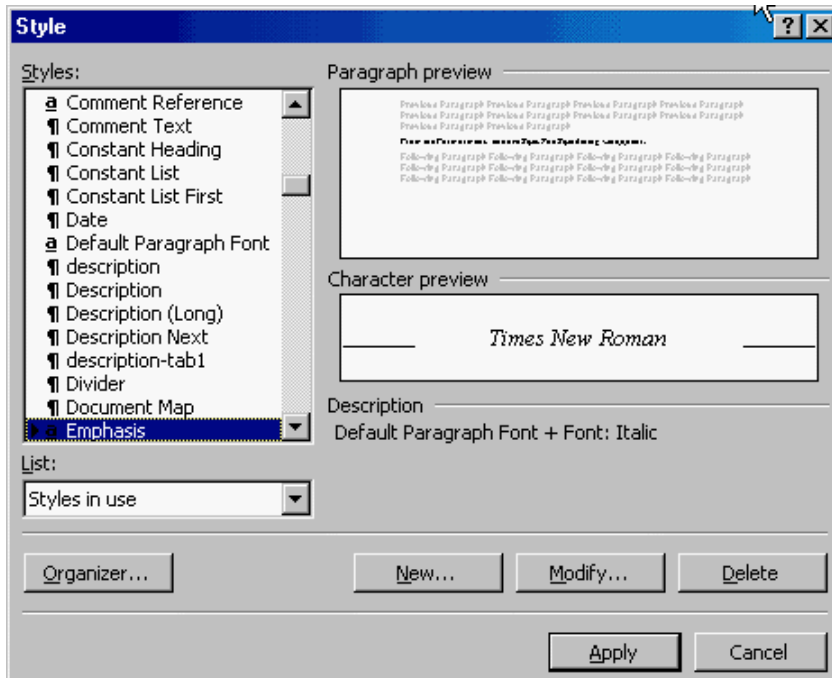
- In the **Styles** box, select the name of the paragraph style to be applied, then click **Apply**. If the desired style is not listed, select another group of styles in the List box. Paragraph styles are marked with the ¶ symbol.

Alternatively, you can apply the desired style directly by selecting it in the Style list of the Formatting toolbar.



Applying Character Styles in Microsoft Word

- Select the text to be formatted. If only a single word is to be formatted, you can click anywhere within the word.
- From the Format menu, choose Style. The Style dialog box opens.












- In the **Styles** box, select the name of the character style to be applied, then click **Apply**. If the desired style is not listed, select another group of styles in the List box. Character styles are marked with the **a** symbol.

Alternatively, you can apply the desired style directly by selecting it in the Style list of the Formatting toolbar.



Working with Projects

A ComponentOne Doc-To-Help 7.5 project file is a database that stores information about a Help project. Doc-To-Help maintains everything except the actual content of the source documents in a project file. Specifically, Doc-To-Help project files contain the following elements:

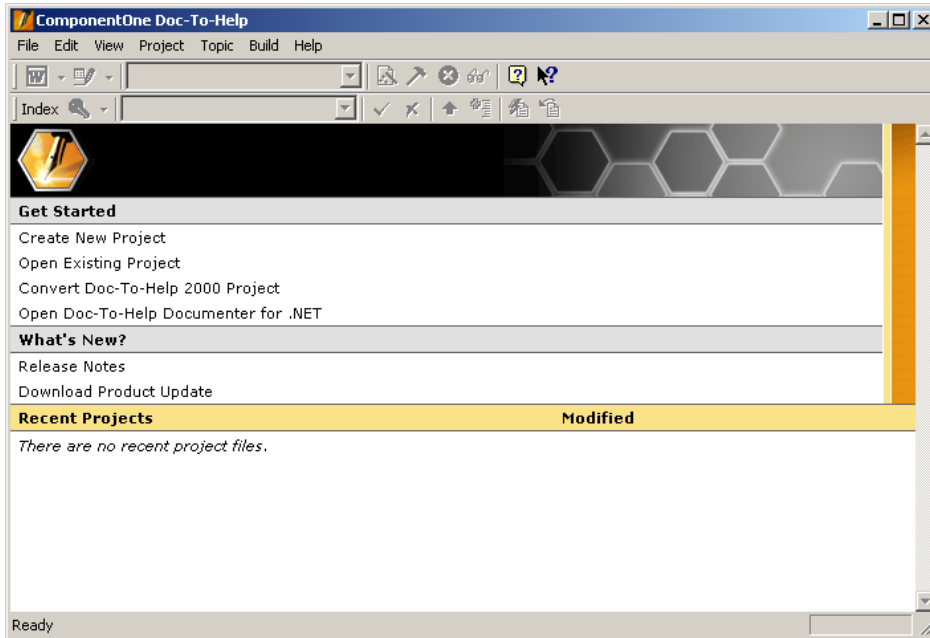
	Source document paths
	Document template paths
	Topic titles derived during compilation
	Paragraph style behaviors
	Character style behaviors
	Script code modules
	Help window definitions
	Index keywords visible to the reader
	Group names hidden from the reader

The icons shown here are used throughout Doc-To-Help to identify elements of a specific type and to indicate relationships between disparate elements.

Since a Doc-To-Help project file is a database, it must exist on your hard disk or a network drive; it cannot reside in memory like a new document in Microsoft Word. By default, Doc-To-Help project files have a *.d2h* extension.

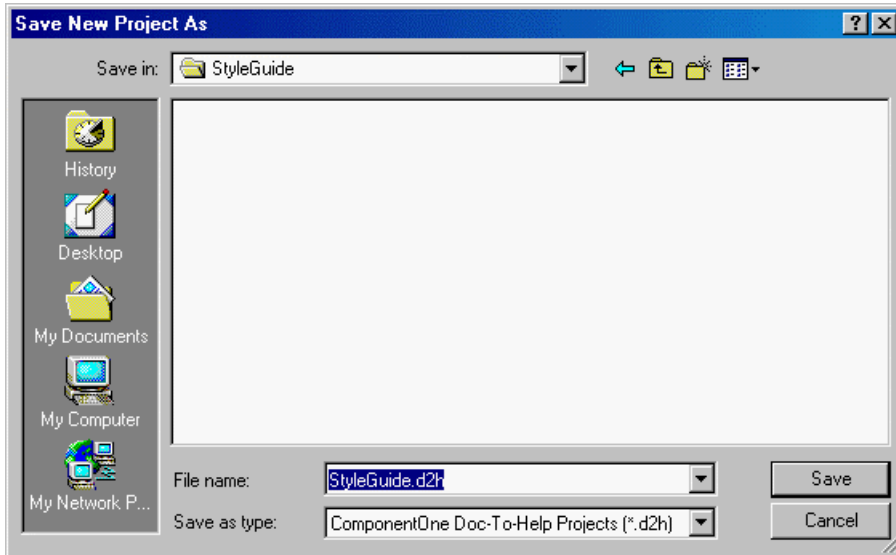
Creating or Opening a Project Upon Startup

When you run Doc-To-Help from the Start menu, or by double-clicking its icon in Windows Desktop, you are presented with the Doc-To-Help Start Page. From there you can choose some of your most recent projects, open an existing project, convert an older Doc-To-Help project or create a new project.

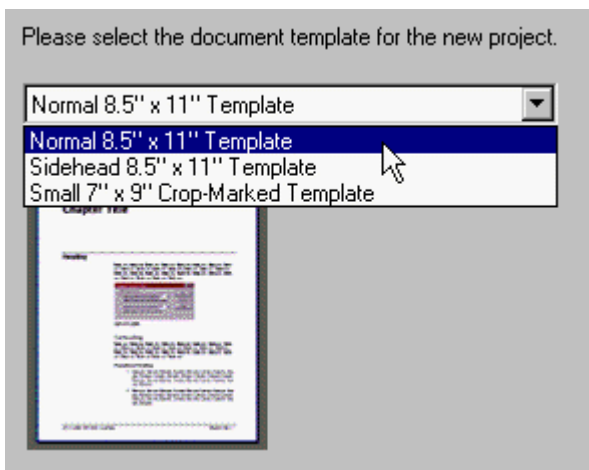


To Create a New Project

- Open the Doc-To-Help program.
You are presented with the Doc-To-Help Start Page.
- Click the **Create New Project** hyperlink. The first screen of the **New Project Wizard** opens.
- Using the **Browse** button to open the **Save New Project As** dialog box, create or choose a directory and enter a project name in the **File name** textbox.

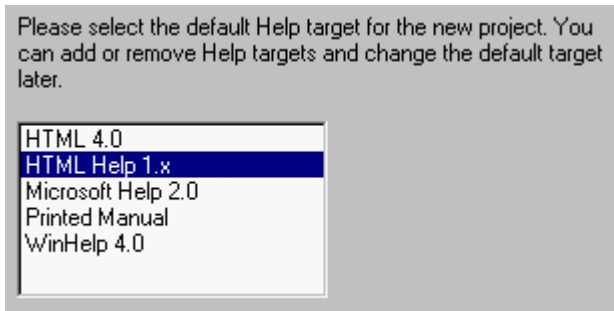


- After you have created the new directory and entered the project name click **Save**, then **Next**.
- From the drop-down list, choose the template that you wish to attach to your new project and click **Next**.



- From the list of Help targets, choose your default Help target and click **Next**.

Though you can build any Help target for your project, this determines the target your project compiles initially.



- Review your directory location and file name and click Finish.

The Doc-To-Help Wizard produces a new Help file, opening the project editor and a starting Microsoft Word document.

To Open a Recent Project

Select any of the most recent project links located in the Doc-To-Help Start Page.

To Open an Existing Project

- Click the **Open an Existing Project** link in the Doc-To-Help Start Page.
- The **Open** dialog box opens.
- Specify a folder and the name of an existing project file, then click Open.

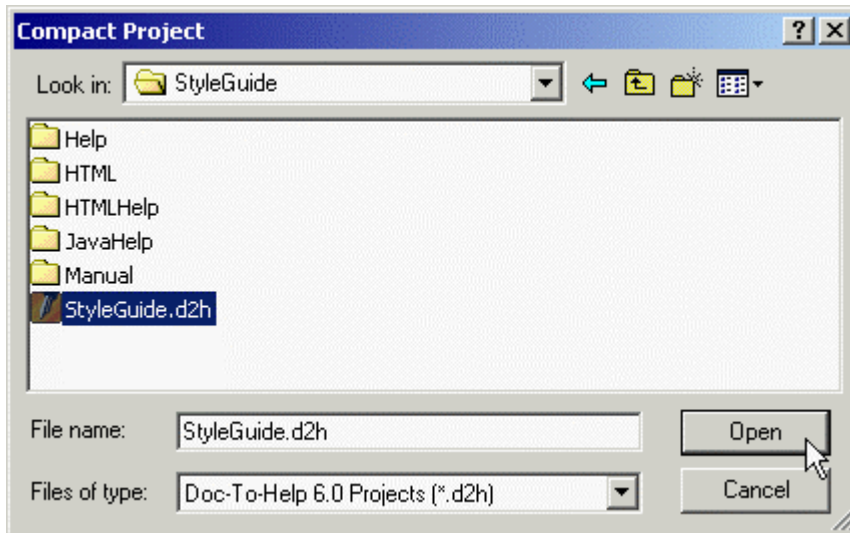
Compacting a Project File

Doc-To-Help includes a project compacting utility that can reduce the size of your project file. This utility checks the project file and, if it is fragmented, rearranges how the project file is stored on disk. When completed, the compacted project has reclaimed wasted space, and is usually smaller than the original.

To run the compacting utility:

- Open the Doc-To-Help program.
- You are presented with the Doc-To-Help Start Page.
- Click Project | Compact Project.

The Compact Project dialog opens.



- From the dialog, select the Doc-To-Help project you would like to compact and click **Open**.
- The utility performs the compacting process automatically.

Note: The compact utility will not work if the Doc-To-Help project you wish to compact is open.

Converting Projects with Doc-To-Help

If you have projects that were produced in Doc-To-Help 2000, you can convert them to Doc-To-Help 7.5 by using the conversion wizard provided.

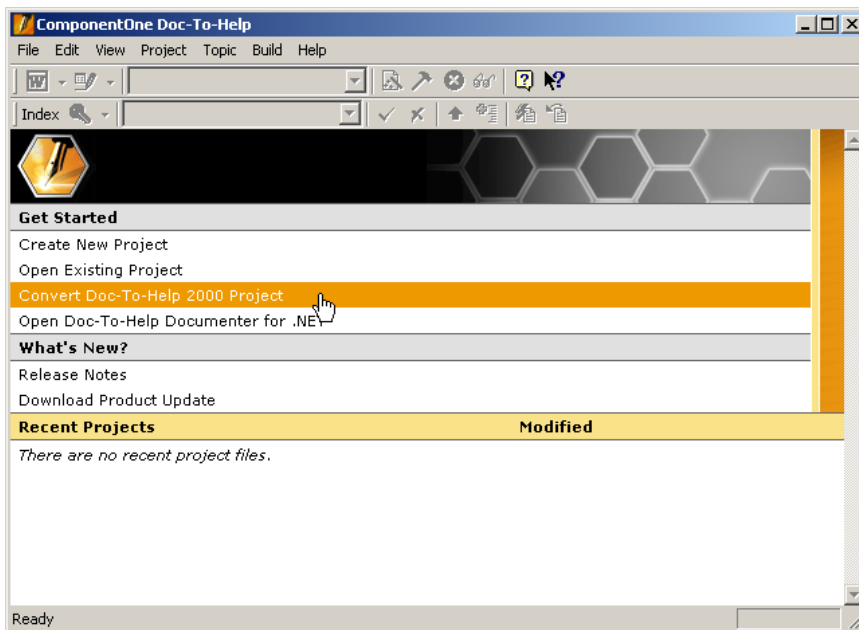
If you are converting a TrueHelp 1.0 project to Doc-To-Help 7.5, see **Converting a TrueHelp Project** for detailed instructions.

Before Converting a Doc-To-Help 2000 Project

The project must be in Doc-To-Help 2000 format and the project must have been “built” (output directories must exist). You should also run the Doc-To-Help 2000 Diagnostics, particularly those options that deal with bookmarks and cross-references.

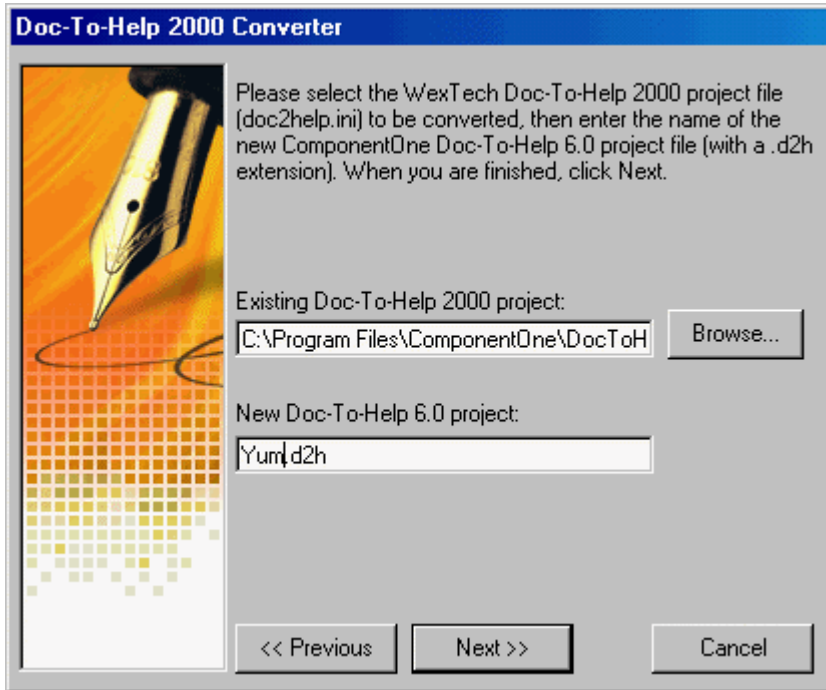
Converting a Doc-To-Help 2000 Project

- Open Doc-To-Help 7.5.
- Click **Convert Doc-To-Help 2000 Project**.



The Doc-To-Help conversion dialog box opens.

- Click **Next** to continue.
- Using the browse button, locate the project (.INI file) you wish to convert and click **Open**.



- In the **New Doc-To-Help project:** textbox, type the name of the new file and click **Next**.
- Review your selections and click **Finish**.

The Doc-To-Help Converter status dialog box opens to display the conversion progress.

- Once the conversion is complete you are prompted with “Project converted successfully. Click **OK** to close this window, then press **F5** to build the current Help target”.

Note: A back-up of your existing Doc-To-Help 2000 project is saved in the D2HconverterBackup directory.

Accessing the Converted Project

By default, Doc-to-Help 7.5 places your converted Help project files and folders in the parent directory where your Doc-To-Help 2000 files resided.

- The source documents are placed directly in the parent directory.

- The related folders are also placed in the parent directory. The Doc-To-Help conversion duplicates the original folder names used in your Doc-To-Help 2000 project.

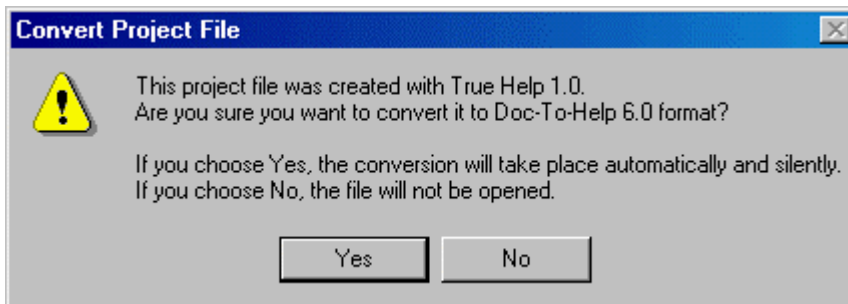
Converting a TrueHelp Project

You can easily convert any True Help project to Doc-to-Help by performing the following steps.

Note: Once you have converted your True Help 1.0 project to a Doc-To-Help 7.5 project, the converted files will no longer work with True Help 1.0. We suggest you make a back-up copy of any projects that you are converting.

- The first step is to Convert your True Help project file to the latest version of Microsoft Access. Open Access without a file (select **Cancel** when the File Selection dialogue appears).
- Select **Tools | Database Utilities | Convert Database | To Current Access Database Version**.
- When the **Database to Convert From** dialogue opens, change the **Files of Type** to **All Files**.
- Select the TrueHelp project you wish to convert and press the **Convert** button.
- Once the conversion is complete, close Access.
- Open the directory containing the True Help project you converted.
- Change the **.thp** extension of the project file to **.d2h**.
- Double-click the project file.

You are asked if you want to convert the file to Doc-To-Help format.



- Click Yes.

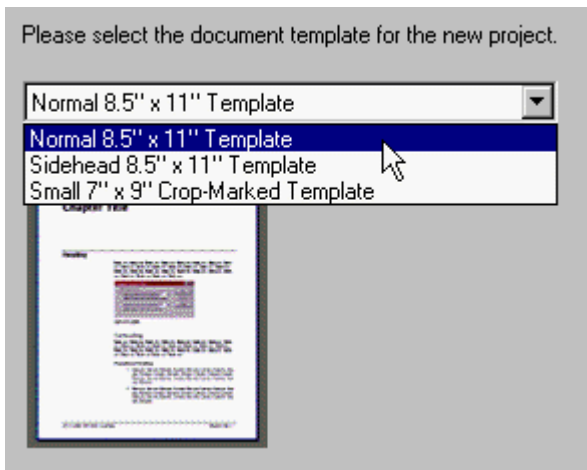
The Doc-To-Help project editor is opened with your converted file.

Creating a New Project from the Project Editor

- On the **File** menu, click **New**.

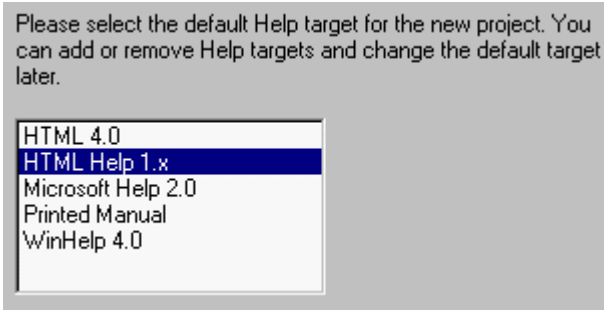
You are presented with the Doc-To-Help Start Page.

- Click the **Create New Project** hyperlink. The first screen of the **New Project Wizard** opens.
- Using the **Browse** button to open the **Save New Project As** dialog box, create or choose a directory and enter a project name in the **File name** textbox.
- After you have created the new directory and entered the project name click **Save**, then **Next**.
- From the drop-down list, choose the template that you wish to attach to your new project and click **Next**.



- From the list of Help targets, choose your default Help target and click **Next**.

Though you can build any Help target for your project, this determines the target your project compiles initially.



- Review your directory location and file name and click **Finish**.

The Doc-To-Help Wizard produces a new Help file, opening the project editor and a starting Microsoft Word document.

Note: When a new project file is created, Doc-To-Help creates a matching start-up document file with the same name as the project file.

Opening an Existing Project from the Project Editor

- On the File menu, click **Open**.
- The **Open** dialog box opens.
- Specify a folder and the name of an existing project file, then click **Open**.

Note: Doc-To-Help keeps track of the last nine project files that you opened. You can open one of these project files directly by clicking one of the numbered commands on the File menu.

Project Settings Properties

By selecting the **Project Settings** item on the project editor tree list, you are presented with the Make and View Project commands as well as links to all document files contained within your project. The lower right property pane contains the default project properties. These properties are generally set globally as defaults for the entire project. You can modify most properties by simply clicking in the box next to the property.

Document Default

The document properties are set globally for all documents, but can be overridden on a document-by-document basis in the document properties.

AdjustLeftIndent	Controls whether paragraph indentation is adjusted to account for wide margins when building online help.
KeepOutlineNumbers	When enabled, outline numbers are included as part of topic titles and are present in help targets.
KeepPageBreaks	When true, this property retains the page break characters in the source documents during compilation of a Printed Manual Help target. Set this property to false to discard page break characters.
SplitFile	Controls whether a document is split into separate topic files for HTML-based platforms.
SplitFileWithHHCtrl	When true, a single HTML file is generated using intrinsic HTML Help commands to give the appearance of separate HTML files for each topic in the document. When false, individual HTML files are generated for each topic in the document.
TextPopups	If true, generates a plain text only version of the Help file for context sensitive Help topics.

Localization

OnPage	Specifies the default text used when completing a cross-reference.
--------	--

Miscellaneous Properties

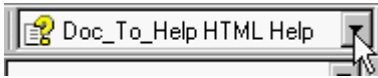
DefaultTemplate	Specifies the document template that will be attached to new documents created in the Project Editor.
Glossary	Specifies the document that contains glossary terms for the project.
MapNumberOffset	The value of this property is added to the automatically generated map numbers to prevent numbering conflicts in modular Help systems.
ModularHub	If true, the generated Help file can dynamically load the contents of other Help files, if present.

URLMode Specifies the rules used for naming the .htm files generated for topics.

Changing the Current Help Target

A Doc-To-Help project always has a current Help target that specifies the kind of Help system to be built or viewed. To change the current Help target, choose the **Help Targets** item in the left pane of the project editor, then do either one of the following:

- On the **Project** toolbar, select one of the available items in the **Help Target** list.



- In the right pane of the project editor, select one of the available items in the Help target list.

Doc_To_Help Help 2. Microsoft Help 2.0	MSHelp
Doc_To_Help HTML HTML 4.0	HTML
Doc_To_Help HTML HTML Help 1.x	HTMLHelp
Doc_To_Help Manua Printed Manual	Manual
Doc_To_Help WinHel WinHelp 4.0	Help

Building the Current Help Target

When you update the current Help target, Doc-To-Help selectively incorporates any substantive changes you have made to the project since it was last built, and performs only those actions that are absolutely necessary. When you rebuild the current Help target, Doc-To-Help empties its output directory and recompiles the entire project, even if you haven't changed anything.

To update the current Help target:

- On the **Build** menu, click **Make Target**.

Alternatively, you can click the **Make Target** button on the **Project** toolbar.

To rebuild the current Help target in its entirety:

- On the **Build** menu, click **Rebuild Target**.

Viewing the Current Help Target

- On the **Build** menu, click **View Target**.

Alternatively, you can click the **View Target** button on the **Project** toolbar.

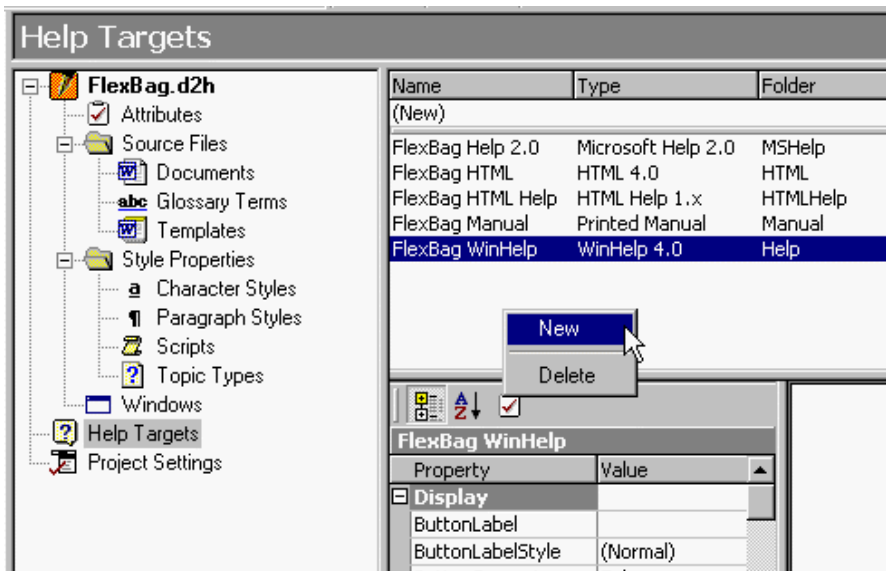
Note: If the current Help target has not been built or needs to be updated, you are asked if you want to build it. If you choose Yes, Doc-To-Help updates the Help target first by executing the Make Target command.

Adding a New Help Target

Doc-To-Help provides five default Help targets. In some situations, you may wish to have multiple Help targets of the same type, utilizing different file names. To add a new Help target to the default list:

- On the Icon bar, click **Project**.
- Select the **Help Targets** item from the left pane.
- Right-click anywhere in the Help target list in the right pane and choose **New** from the shortcut menu.

Note that the cursor has moved to the top of the Help target list window.



- Type the name of the new Help target.
- Choose the type of Help target from the **Type** drop-down menu and press **Enter**.

Note that the new Help target has been added to the list. This new Help target contains the default properties of the **Type** you chose.

Alternatively, you can add a new Help target by clicking the **Project** menu, choosing **New**, then clicking **Help Target**.

Changing the Help Target Directory

By default, Doc-To-Help stores the Help target files in an output directory whose name corresponds to the type of the Help target, or, in the case of a newly created Help target, the default output directory matches the name of the Help target. You can easily change the name of your output directory as needed:

- On the Icon bar, click **Project**.
- Select the **Help Targets** item from the left pane.
- Choose the Help target you wish to edit from the right pane of the project editor.

Note that the property pane changes to reflect the Help target chosen.

- In the Help target property pane, locate the Folder property. In the text box to the right of the property, highlight the existing name and change it as desired.

Help Target Properties

By selecting the **Help Targets** item on the project editor tree list, you are presented with a list of the currently defined Help targets in the right pane. By choosing any one of the Help targets, the lower left pane displays the property settings for that target.

Microsoft Help 2.0

This target produces the standard Microsoft Help 2.0.

Appearance Properties

BackColor	Displays the current background color setting for the Help pane.
BackgroundImage	Displays the image file to be used as a background in the Help pane.
BackgroundRepeat	Defines how the background image is arranged in the Help pane.
Theme	Determines the appearance of the Help pane and framesets for HTML-based targets.

Build Properties

GenerateCSS	When set to false, the existing cascading stylesheet will not be overwritten when building the Help target.
-------------	---

Display Properties

ButtonLabel	Specifies the text that precedes the subtopic buttons.
ButtonLabelStyle	Lists the name of the style used to format the label that precedes the subtopic buttons.
ButtonSeparator	When enabled, a dividing line separates the topic text from the subtopic buttons.
ButtonStyle	Lists the name of the style used to format automatically generated subtopic buttons.
ColorReduction	Determines the color threshold for images that are saved with source documents.
Value	Description
None	Images are not converted.
16 Colors	Images with more than 16 colors are reduced to 16.
256 Colors	Images with more than 256 colors are reduced to 256.
ScaleGraphics	When set to false, graphics contained with the source documents will not be scaled.

Localization Properties

Charset	Specifies one of the character set aliases listed in HTML Character Sets and Named Entities.
CodePage	Defines the windows codepage number to be used by the Help file. This property is most commonly used when producing Help files in foreign languages, providing special character sets for translation on the screen or in print.

Miscellaneous Properties

Basename	Base file name to be used for generating Help target files. By default, this name will be the same as the file name for the Help project itself.
----------	--

CopyFolder	Defines the folder name where ancillary build files will reside.
Folder	Defines the folder name where the Help files will reside.
Name	Returns a string containing the name of a specified Help target (read-only).
Template	The document template to apply to the intermediate files during compilation.

Strings Properties

Namespace	Specifies the Help 2.0 namespace that is used to register the Help file after a successful build.
ParentNamespace	Specifies the namespace for the built help collection to plug into. Usually, that will be the namespace of the Visual Studio .NET Combined Collection. Set this property if you want the built Help to be automatically registered as a plug-in for Visual Studio help. For example, for Visual Studio .NET 2003, set ParentNamesace to "MS.VSCC.2003".
StringNext	Allows modification of the Next string value.
StringPrevious	Allows modification of the Previous string value.

HTML 4.0

This target produces browser neutral HTML in accordance with version 4.0 of the HTML specification published by W3C.

Appearance Properties

BackgroundColor	Displays the current background color setting for the Help pane.
BackgroundImage	Displays the image file to be used as a background in the Help pane.
BackgroundRepeat	Defines how the background image is arranged in the Help pane.
Theme	Determines the appearance of the Help pane and framesets for HTML-based targets.

Build Properties

GenerateCSS	When set to false, the existing cascading stylesheet will not be overwritten when building the Help target.
-------------	---

Display Properties

ButtonLabel	Specifies the text that precedes the subtopic buttons.
ButtonLabelStyle	Lists the name of the style used to format the label that precedes the subtopic buttons.
ButtonSeparator	When enabled, a dividing line separates the topic text from the subtopic buttons.
ButtonStyle	Lists the name of the style used to format automatically generated subtopic buttons.
ColorReduction	Determines the color threshold for images that are saved with source documents.
Value	Description
None	Images are not converted.
16 Colors	Images with more than 16 colors are reduced to 16.
256 Colors	Images with more than 256 colors are reduced to 256.
ScaleGraphics	When set to false, graphics contained with the source documents will not be scaled.

HTML Properties

DefaultFile	File name to be used as part of the base URL for the HTML target. Typically, the filename will be <i>default.htm</i> or <i>index.html</i> .
DynamicTOC	Generates a Dynamic HTML version of the contents page with an expandable outline.
Frameset	Generates a frameset version of the HTML Help project with the left frame displaying the contents and the right frame displaying the Help topics.

Localization Properties

Charset	Specifies one of the character set aliases listed in HTML Character Sets and Named Entities.
---------	--

CodePage	Defines the windows codepage number to be used by the Help file. This property is most commonly used when producing Help files in foreign languages, providing special character sets for translation on the screen or in print.
----------	--

Miscellaneous Properties

Basename	Base file name to be used for generating HTML target files. By default, this name will be the same as the file name for the Help project itself.
CopyFolder	Defines the folder name where ancillary build files will reside.
Folder	Defines the folder name where the Help files will reside.
Name	Returns a string containing the name of a specified Help target (read-only).
Template	The document template to apply to the intermediate files during compilation.

Strings Properties

PrefixIndexListFile	Allows modification of the default prefix for topic list files corresponding to the main index.
PrefixTopicListFile	Allows modification of the default prefix for topic list files corresponding to keyword hot spots.
StringContents	Allows modification of the contents string value.
StringFoundMany	Allows modification of the %d topics found: string value.
StringFoundOne	Allows modification of the 1 topic found: string value.
StringFoundZero	Allows modification of the No topics found. string value.
StringIndex	Allows modification of the index string value.
StringNext	Allows modification of the next string value.
StringPrevious	Allows modification of the previous string value.
StringReturnToIndex	Allows modification of the return to index string value.

HTML Help

This target produces Microsoft HTML Help, which requires Internet Explorer to be installed on the client machine. Internet Explorer version 3.0 is acceptable, but 4.0 or greater is preferred.

Appearance Properties

BackColor	Displays the current background color setting for the Help pane.
BackgroundImage	Displays the image file to be used as a background in the Help pane.
BackgroundRepeat	Defines how the background image is arranged in the Help pane.
Theme	Determines the appearance of the Help pane and framesets for HTML-based targets.

Build Properties

BinaryIndex	When true, generates a binary index instead of a site map index.
BinaryTOC	When true, generates a binary table of contents instead of a site map table of contents.
GenerateCSS	When set to false, the existing cascading stylesheet will not be overwritten when building the Help target.
GenerateProjectFile	When set to false, the existing .hjp or .hhp files will not be overwritten when building the Help target.

Display Properties

ButtonLabel	Specifies the text that precedes the subtopic buttons.
ButtonLabelStyle	Lists the name of the style used to format the label that precedes the subtopic buttons.
ButtonSeparator	When enabled, a dividing line separates the topic text from the subtopic buttons.
ButtonStyle	Lists the name of the style used to format automatically generated subtopic buttons.
ColorReduction	Determines the color threshold for images that are saved with source documents.

Value	Description
None	Images are not converted.
16 Colors	Images with more than 16 colors are reduced to 16.
256 Colors	Images with more than 256 colors are reduced to 256.
DefaultWindow	Specifies the default window definition for the compiled help file.
ScaleGraphics	When set to false, graphics contained with the source documents will not be scaled.
TopicsFound	Determines whether Alink/Klink topic lists are displayed in a dialog box (default) or a pop-up menu.
Localization Properties	
Charset	Specifies one of the character set aliases listed in HTML Character Sets and Named Entities.
CodePage	Defines the windows codepage number to be used by the Help file. This property is most commonly used when producing Help files in foreign languages, providing special character sets for translation on the screen or in print.
Miscellaneous Properties	
Basename	Base file name to be used for generating HTML Help target files. By default, this name will be the same as the file name for the Help project itself.
CopyFolder	Defines the folder name where ancillary build files will reside.
Folder	Defines the folder name where the Help files will reside.
MapFileForVB	Creates Visual Basic header files for context sensitive Help.
MapNumbers	Determines whether a unique Context ID (map number) is generated for each topic.
Name	Returns a string containing the name of a specified Help target (read-only).

SkipGlossary	When true, omits the Glossary topic from the generated Help contents.
Template	The document template to apply to the intermediate files during compilation.

Printed Manual

This target produces a Microsoft Word document that includes a table of contents, an index and the individual subdocuments generated from your source documents.

Build Properties

Concatenate	When true, subdocuments are concatenated into a single master document in printed manual builds.
FixLists	Controls how numbered lists are corrected in the printed manual output when the Concatenate property is set to true.

Value	Description
Never	Never corrects the numbered lists.
Always	Always corrects the numbered lists.
Word 2000 Only (Default)	Corrects lists with Word 2000, but not with Word 2002 (XP).

Localization Properties

CodePage	Defines the windows codepage number to be used by the Help file. This property is most commonly used when producing Help files in foreign languages, providing special character sets for translation on the screen or in print.
----------	--

Miscellaneous Properties

Basename	Base file name to be used for generating the target file. By default, this name will be the same as the file name for the Help project itself.
CopyFolder	Defines the folder name where ancillary build files will reside.
Folder	Defines the folder name where the Help files will reside.

Name	Returns a string containing the name of a specified Help target (read-only).
Template	The document template to apply to the intermediate files during compilation.

Title Page Properties

ByLine	Specifies the byline text used on the title page of printed manuals.
SuperTitle	Specifies the text that appears above the title string on the title page of printed manuals.
Title	Represents the title string of an object.

WinHelp 4.0

This target produces the standard 32-bit Windows Help 4.0.

Build

GenerateProjectFile	When set to false, the existing .hbj or .hhp files will not be overwritten when building the Help target.
---------------------	---

Display Properties

ButtonLabel	Specifies the text that precedes the subtopic buttons.
ButtonLabelStyle	Lists the name of the style used to format the label that precedes the subtopic buttons.
ButtonSeparator	When enabled, a dividing line separates the topic text from the subtopic buttons.
ButtonStyle	Lists the name of the style used to format automatically generated subtopic buttons.
ColorReduction	Determines the color threshold for images that are saved with source documents. This property has no effect on linked images that are not saved with the document.

Value

Description

None	Images are not converted.
16 Colors	Images with more than 16 colors are reduced to 16.
256 Colors	Images with more than 256 colors are reduced to 256.

ScaleGraphics When set to false, graphics contained with the source documents will not be scaled.

Localization Properties

CodePage Defines the windows codepage number to be used by the Help file. This property is most commonly used when producing Help files in foreign languages, providing special character sets for translation on the screen or in print.

Miscellaneous Properties

Basename Base file name to be used for generating WinHelp target files. By default, this name will be the same as the file name for the Help project itself.

Citation Appends a citation to the end of any information (except in a context-sensitive pop-up window) that is copied from the Help file.

Compression Determines the level of file compression used by the WinHelp compiler.

Value	Description
None	No compression.
Chose best	The WinHelp compiler determines the best algorithm to use.
Phrase	For Help files under 100K.
Hall	For Help files that will be compressed again by another utility.
Zeck	Minimal compression for quick builds.
Zeck + Phrase	Minimal compression for Help files under 100K.
Zeck + Hall	Maximum compression.
CopyFolder	Defines the folder name where ancillary build files will reside.
Copyright	Places a custom copyright notice in the Version dialog box of WinHelp.

Folder	Defines the folder name where the Help files will reside.
HubContents	Displays the name of the contents (.cnt) file to associate with the Help file.
MapFileForC	Creates C or C++ header files for context sensitive Help.
MapFileForVB	Creates Visual Basic header files for context sensitive Help.
MapNumbers	Determines whether a unique Context ID (map number) is generated for each topic.
Name	Returns a string containing the name of a specified Help target (read-only).
SkipGlossary	When true, omits the Glossary topic from the generated Help contents.
Template	The document template to apply to the intermediate files during compilation.
Title	Represents the title string of an object.
WinHelpMacro	Specifies a macro to run when a Help target, a topic, or secondary window is opened.

Navigation Properties

OneBrowseSequence	When true, this property provides a continues browse sequence that spans multiple source documents.
-------------------	---

JavaHelp

This target produces Java Help 1.1.3. For more information, see [JavaHelp Target](#) (page 34).

Appearance Properties

BackColor	Displays the current background color setting for the Help pane.
BackgroundImage	Displays the image file to be used as a background in the Help pane.
BackgroundRepeat	Defines how the background image is arranged in the Help pane.

Build Properties

GenerateCSS When set to false, the existing cascading stylesheet will not be overwritten when building the Help target.

Display Properties

ButtonLabel Specifies the text that precedes the subtopic buttons.

ButtonLabelStyle Lists the name of the style used to format the label that precedes the subtopic buttons.

ButtonSeparator When enabled, a dividing line separates the topic text from the subtopic buttons.

ButtonStyle Lists the name of the style used to format automatically generated subtopic buttons.

ScaleGraphics When set to false, graphics contained with the source documents will not be scaled.

Localization Properties

Charset Specifies one of the character set aliases listed in HTML Character Sets and Named Entities.

CodePage Defines the windows codepage number to be used by the Help file. This property is most commonly used when producing Help files in foreign languages, providing special character sets for translation on the screen or in print.

Miscellaneous Properties

Basename Base file name to be used for generating HTML Help target files. By default, this name will be the same as the file name for the Help project itself.

CopyFolder Defines the folder name where ancillary build files will reside.

Folder Defines the folder name where the Help files will reside.

Name Returns a string containing the name of a specified Help target (read-only).

Template The document template to apply to the intermediate files during compilation.

Strings Properties

PrefixIndexListFile	Allows modification of the default prefix for topic list files corresponding to the main index.
PrefixTopicListFile	Allows modification of the default prefix for topic list files corresponding to keyword hot spots.
StringContents	Allows modification of the contents string value.
StringFoundMany	Allows modification of the %d topics found: string value.
StringFoundOne	Allows modification of the 1 topic found: string value.
StringFoundZero	Allows modification of the No topics found. string value.
StringIndex	Allows modification of the index string value.
StringNext	Allows modification of the next string value.
StringPrevious	Allows modification of the previous string value.
StringReturnToIndex	Allows modification of the return to index string value.

Documents and Templates

When working with ComponentOne Doc-To-Help 7.5, you supply the source text of a Help system by adding one or more Microsoft Word documents to a Doc-To-Help project file. Only the pathname is stored, not the document contents.

You can also supply optional Microsoft Word templates to apply target-specific formatting during compilation. As with documents, only the pathname is stored.

Once a document or template has been added to a project, you can use Doc-To-Help commands to open it, either in the default mode, using a separate Word application window, or in-place within the right pane of the Doc-To-Help project editor.

Select the **Documents** item on project editor tree list to get a list of the source documents currently used by the Help project in the right pane. Select any one of the source documents and the property pane displays the property settings for that document. You can modify most properties by simply clicking the box next to the property.

Organizing Your Project Files

When you first build a Help target, Doc-To-Help creates a subdirectory (within the directory containing the project file) for all intermediate and final output. This keeps the generated files separate from your source files. Note that you can change the name of any Help target directory by setting the Help target Folder directory.

The following table shows the mapping between the directory name and the default Help target:

MSHelp	Microsoft Help 2.0
HTML	HTML 4.0
HTMLHelp	HTML Help 1.2
Manual	Printed Manual
Help	WinHelp 4.0
Help Target Name	User Created Help Target

IMPORTANT: Do not place any of your own files in these subdirectories, as Doc-To-Help may overwrite or delete them during the build process. For this reason, placing multiple Doc-To-Help projects (*.d2h* files) in a single directory is also not recommended.

Doc-To-Help stores document and template pathnames relative to the directory that contains the project file. Therefore, it is a good idea to keep your Microsoft Word

documents (.doc files) and Doc-To-Help projects (.d2h files) in the same directory. That way, if you copy the entire directory to another location, Doc-To-Help will still be able to find your source documents.

If a Doc-To-Help project contains one or more Microsoft Word templates (.dot files), you can keep them in the same directory as the project file. However, if you create many Help projects, you may not want to have a duplicate set of templates for each one. In that case, you can keep one copy of the templates in a common directory, and each project can refer to them by their absolute pathnames. If you follow this approach, remember to copy the templates when transferring your source files to another machine.

Adding a New Document to a Project

Many times it is advantageous to have your Help file separated into several smaller source documents. One of the advantages of using separate source documents for each topic is that it provides you with the flexibility of having several authors working on the project simultaneously. Additionally, when changes are made to topics, Doc-To-Help only needs to compile the source documents that have been modified, greatly reducing the compile time.

Document Properties

By selecting the **Documents** item on the project editor tree list, you are presented with a list of the project source documents in the right pane. By choosing any one of the documents, the lower left pane displays the property settings for that document.

Miscellaneous Properties

AdjustLeftIndent	Controls whether paragraph indentation is adjusted to account for wide margins when building online help.
KeepOutlineNumbers	When enabled, outline numbers are included as part of topic titles and are present in help targets.
Name	Returns a string containing the name of a specified object (read-only).
Order	Determines the order of the documents in the Help contents.

For more information, see Documents and Templates (page 133.)

Glossary Terms

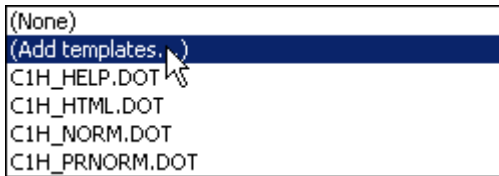
To assist in developing an online and printed glossary, Doc-To-Help automatically creates an empty glossary document whenever a new project is created. This document remains as your final document in the Help project by default. A listing of the glossary

entries can be viewed by clicking the **Glossary** item in the project editor tree. For more information on adding and modifying glossary terms, see *Modifying the Glossary* (page 266).

Templates

With Doc-To-Help you can supply optional Microsoft Word templates to apply target-specific formatting during compilation. Adding additional templates is simply a matter of right-clicking in the templates list pane on the right side of the project editor.

Alternatively, you can select the Template property for any Help target and from the template selection dropdown, pick **Add templates**. By using this method, the template is added into the project and attached as the template for the current Help target.



By selecting the **Templates** item on the project editor tree list, you are presented with a list of the default templates in the right pane. By choosing any one of the templates, the lower left pane displays the property settings for that template.

Miscellaneous Properties

Name	Returns a string containing the name of a specified object (read-only).
------	---

For more information, see *Documents and Templates* (page 133.)

Character Styles

In Microsoft Word, you can use styles to apply character formats (such as font styles and colors) to give your documents a uniform appearance.

Doc-To-Help extends Word's concept of styles to include Help authoring behavior. For each character style that references a topic, you define a like-named character style in the project file. By customizing the properties of Doc-To-Help styles, you define the behavior of the compiled Help system.

Display Properties

AffectsAppearance	Determines whether the style defines the appearance of the text (font, color, etc.) in the target help file, or is only used to create a hotspot or keyword and does not affect target appearance.
-------------------	--

PageRef	When producing a manual, this property places the page number of the referenced topic next to the text formatted with this style.
Replacement	This property sets or returns a string that overrides the default text generated by Doc-To-Help for a character style with PageRef set to True, the Replacement string specifies the format of page number references in printed manual targets. The string is ignored unless it contains a pound sign, which is replaced with a PAGEREF field.
Window	Shows the name of the window in which topics formatted in this style are displayed.

Indexing Properties

AutoIndex	When true, this property automatically creates index keywords from text formatted with that character style and associates them with the topic that contains the text.
-----------	--

Miscellaneous Properties

MultiLink	When false, only the first occurrence of a "jump" character style (in a given topic) generates a link. All other formatted occurrences are skipped.
Name	Returns a string containing the name of a specified object (read-only).
Script	Displays the script to be run during compilation whenever text formatted with this style is encountered.
Type	Returns a constant that represents the style type of the specified object (read-only).

Value	Description
None	No type.
Jump	Topic jump hot spot.
Popup	Topic popup hot spot.
Group	Group link hot spot.
Keyword	Keyword link hot spot.

Adding a New Source Document

- On the Icon bar, click **Project**.
- Select the **Documents** item from the left pane.
- Right-click anywhere in the document list in the right pane and choose **Add Documents** from the shortcut menu.

The **Add Documents** dialog box opens.

- Select the document you wish to add and click **Open**.

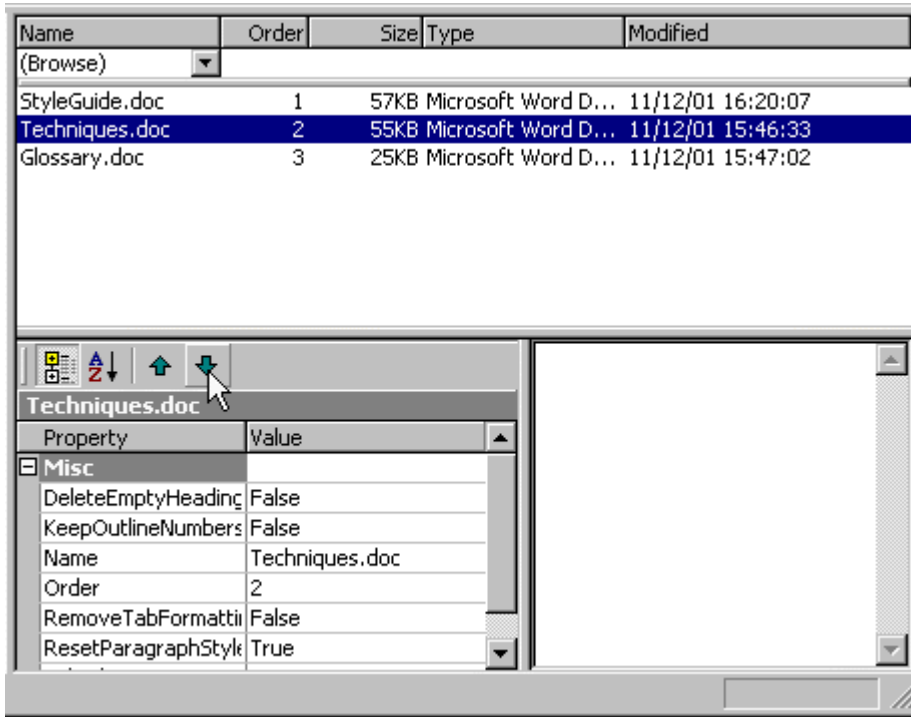
Note that the new document has been added to the list. This new document contains the default properties and can be modified as necessary.

Alternatively, you can add a new document by clicking the **Project** menu, choosing **New Document**, then choosing the document.

Modifying the Order of the Source Documents

In a project that contains multiple source documents, the order in which they were added to the project determines the Order property for each document. Since this may not be the order you intended, you can rearrange it.

- On the Icon bar, click **Project**.
- Select the **Documents** item from the left pane.
- Select the source document you wish to edit.
- Change the Order property in the property pane or press the up or down arrow located in at the top of the documents property pane.



Adding Templates to a Project

You can easily supply optional Microsoft Word templates to apply target-specific formatting during compilation.

Select the **Templates** item on the project editor tree list to get a list of the default templates in the right pane. Select any one of the templates and the property pane displays the property settings for that template.

To add a new template to the default list:

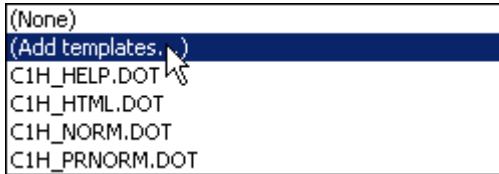
- On the Icon bar, click **Project**.
- Select the **Templates** item from the left pane.
- Right-click anywhere in the templates list in the right pane and choose **Add Templates** from the shortcut menu.

The **Add Templates** dialog box opens.

- Select the template you wish to add and click **Open**.

The new template is added to the list. Note that you'll need to add the template to the project in order to attach it to any of the Help targets.

Alternatively, you can select the Template property for any Help target and from the template selection dropdown, pick **Add templates**. By using this method, the template is added into the project and attached as the template for the current Help target.



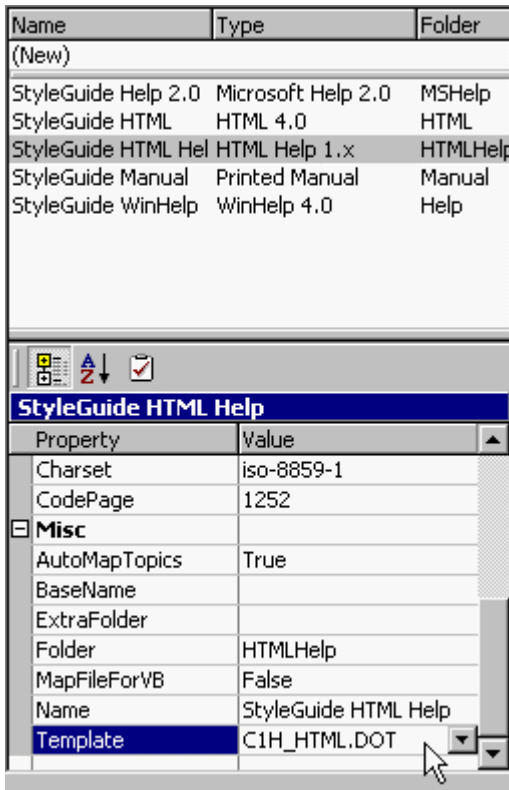
Assigning a template to a specific Help target

Once you've added a template to a project, assign it to a specific Help target so that it will be applied to the source document copies during compilation. To assign a template to a Help target:

- On the Icon bar, click **Project**.
- Select the **Help Targets** item from the left pane.
- Select the Help target you wish to edit from the right pane.

Note that the property pane changes to reflect the Help target chosen.

- In the Help target property pane, locate the Template property. In the text box to the right of the property, select the name of the desired template.



Using Custom Templates

One of the strengths of Doc-To-Help is the ability to use custom templates. For most authors, the default templates supplied with Doc-To-Help should be sufficient for most applications, but in some cases it may be necessary to use a customized template. Generally, it is a good practice to store these custom templates with the associated project. In the past, storing a template with the project could be an issue when moving the project because the template files were stored with an absolute path (full path to template directory). Now all template files are stored using a relative path, no matter where you move the project, the templates associations remain intact.

Note: If the template being used by a target is stored in both the project directory and the Word template directory, the template in the project directory will be used by default.

Opening a File From a Project

To open a document or template in Microsoft Word:

- On the Icon bar, click **Project**.

- Select the **Documents** item from the left pane.
- Right-click the document you wish to open and choose **Open** from the shortcut menu.

The document you have chosen opens within Microsoft Word.

To edit a document or template in-place within Doc-To-Help:

- On the Icon bar, click **Project**.
- Select the **Documents** item from the left pane.
- Right-click the document you wish to open and choose **Edit** from the shortcut menu.

The document you have chosen opens within the right pane of the project editor.

Note: Some anti-virus software options can limit the functionality of Doc-To-Help's "in place" editing features. For a list of known issues, contact Technical Support (page 15).

To close an in-place document or template within Doc-To-Help:

- On the **File** menu, click **Close and Return to Doc-To-Help**.

Removing a File From a Project

- Based on what file you wish to remove, click **Project** on the Icon bar and choose the **Documents** or **Templates** item from the left pane.
- Select and right-click the file you wish to remove and choose **Remove** from the shortcut menu.


Note: When you remove a document from a project, all of its topic records are deleted. However, no index keywords or groups are deleted, even if they were created during compilation of the file that was removed.

Compiling Source Documents

When you build the current Help target, Doc-To-Help compiles your source documents and combines the resulting files into a single entity. However, there are times when you may want to compile a source document individually without updating the entire target. For example, if a build fails because a topic title exceeds the 255-character limit, you can edit the topic title, and then compile that document alone to verify that the problem is fixed before proceeding with a full build.

To compile a single document:

- On the Icon bar, click **Project**.
- Select the **Documents** item from the left pane.
- Select and right-click the document that you wish to compile and select **Compile** from the shortcut menu.

Alternatively, you can compile a document by highlighting the document and clicking the **Compile File** button  on the **Project** toolbar.

To compile all documents in the project:

- From the **Build** menu, select **Compile All**.

To interrupt compilation of one or more documents:

- Click the **Stop Build** button  on the **Project** toolbar

Using Styles in Doc-To-Help

In Microsoft Word, you can use styles to apply paragraph formats (such as indentation and spacing) and character formats (such as font styles and colors) to give your documents a uniform appearance. If you apply styles consistently, any change you make to a style definition will be propagated throughout the document, and the affected regions will be updated accordingly. Styles are also used in Word's outline view to specify the hierarchy of paragraphs within a document.

After creating paragraph or character styles in Microsoft Word, define a matching style within the ComponentOne Doc-To-Help 7.5 project. Through those matching Doc-To-Help styles, you define the behavior of the compiled Help system.

For more information on defining styles in Microsoft Word, see *Working With Styles in Word* (page 102).

Using Paragraph Styles

For each Word paragraph style that begins a topic, define a like-named paragraph style in the project file so that Doc-To-Help compiles the Help system properly. For example, if you have a word paragraph style named SubHeading, you should also have a paragraph style in the project file called SubHeading. In this way, you can simply select which properties you want to apply to the SubHeading style in the property pane of the Doc-To-Help project editor.

Note that Topic Types and Paragraph Styles have many duplicate properties. When a paragraph style has a TopicType associated with it, the duplicate Topic Type properties override the Paragraph Style properties. To make it easy to determine which properties are overridden, Doc-To-Help dims the Paragraph Style properties when unavailable.

Paragraph Style Properties

You can edit the Paragraph Style properties by accessing property pane for the specific Paragraph Style. For more information, see *Editing Paragraph Style Properties* (page 148).

Display Properties

ContentsOnly

When true, omits topics defined with this paragraph style from the Help target, but uses them as book titles in the Help contents. This property is duplicated in the Topic Types.

Nonscrolling	When the NonScrolling property of a paragraph style is set to true, any topic headings formatted with that paragraph style appear in the non-scrolling region of the Help window. This property is duplicated in the Topic Types.
OnlineOnly	When true, omits paragraphs formatted with this style from the printed manual target only.
Popup	When the popup property of a paragraph style is set to true, any topic formatted with that paragraph style appear in a popup window on top of the Help window. This property is duplicated in the Topic Types.
Replacement	Sets or returns a string that overrides the default text generated by Doc-To-Help. For a paragraph style with AutoNext set to True, the Replacement string is used instead of the topic title next to the generated button. This property is duplicated in the Topic Types.
Untitled	When true, the topic title is not displayed in the Help window. This property is duplicated in the Topic Types.
Window	Shows the name of the window in which topics formatted in this style are displayed. This property is duplicated in the Topic Types.

Indexing Properties

AutoKeyword	When true, this property automatically creates index keywords from topic titles formatted with that paragraph style and associates them with the appropriate topic. This property is duplicated in the Topic Types.
-------------	---

Miscellaneous Properties

AutoContextID	When true, automatically creates Context IDs for all topics using this paragraph style. This property is duplicated in the Topic Types.
Name	Returns a string containing the name of a specified object (read-only). This property is duplicated in the Topic Types.
Script	Displays the script to be run during compilation whenever text formatted with this style is encountered. This property is duplicated in the Topic Types.

TopicType	Specifies a named set of display, navigation, and indexing characteristics to be associated with an object (such as what window the Help topic appears in, how the Help topic is accessed, and whether it gets a map number).
Type	Determines the help authoring behavior associated with a style at compile time.

Value	Description
--------------	--------------------

Body Text	Topic body.
Heading Text	Topic with no outline level.
Level 1	Topic at outline level 1.
Level 2	Topic at outline level 2.
Level 3	Topic at outline level 3.
Level 4	Topic at outline level 4.
Level 5	Topic at outline level 5.
Level 6	Topic at outline level 6.
Level 7	Topic at outline level 7.
Level 8	Topic at outline level 8.
Level 9	Topic at outline level 9.

Navigation Properties

AutoButtons	When true, displays subtopic buttons for paragraph styles with numeric outline levels greater than the parent level. This property is duplicated in the Topic Types.
AutoLink	Creates unique link tags for topics formatted with the paragraph style by enabling the auto link tag feature. This property is duplicated in the Topic Types.
AutoNavigate	Determines which paragraph styles are included in the navigation sequence. This property is duplicated in the Topic Types.
AutoNext	Enables the next topic button for any active paragraph style. This property is duplicated in the Topic Types.

ExplicitAccess	When true, defines a specific paragraph style as accessible only through a hyperlink. Paragraph styles defined as explicit are not accessible from the Help contents, the index list, or the text search. This property is duplicated in the Topic Types.
MidTopic	Defines a paragraph style as an “in topic” jump location similar to an HTML anchor. By defining a paragraph style as MidTopic and defining a character style as a jump, end users have jump functionality within a lengthy topic. This property is duplicated in the Topic Types.

Default Paragraph Styles

	Display Properties	Indexing Properties	Miscellaneous Properties	Navigation Properties
Heading 1	ContentsOnly=False Nonscrolling=True OnlineOnly=False Popup=False Replacement= Untitled=False Window=(Default)	AutoKeyword=False	AutoContextID=True Name=Heading 1 Script=(None) TopicType=Contents Type=Level 1	AutoButtons=False AutoLink=True AutoNavigate=True AutoNext=False ExplicitAccess=False MidTopic=False
Heading 2	ContentsOnly=False Nonscrolling=True OnlineOnly=False Popup=False Replacement= Untitled=False Window=(Default)	AutoKeyword=False	AutoContextID=True Name=Heading 2 Script=(None) TopicType=Conceptual Type=Level 2	AutoButtons=True AutoLink=True AutoNavigate=True AutoNext=False ExplicitAccess=False MidTopic=False
Heading 3	ContentsOnly=False Nonscrolling=True OnlineOnly=False Popup=False Replacement= Untitled=False Window=(Default)	AutoKeyword=False	AutoContextID=True Name=Heading 3 Script=(None) TopicType=Conceptual Type=Level 3	AutoButtons=True AutoLink=True AutoNavigate=True AutoNext=False ExplicitAccess=False MidTopic=False
Heading 4	ContentsOnly=False Nonscrolling=True OnlineOnly=False Popup=False Replacement= Untitled=False Window=(Default)	AutoKeyword=False	AutoContextID=True Name=Heading 4 Script=(None) TopicType=Procedural Type=Level 4	AutoButtons=True AutoLink=True AutoNavigate=True AutoNext=False ExplicitAccess=False MidTopic=False

Heading 5	ContentsOnly=False Nonscrolling=True OnlineOnly=False Popup=False Replacement=Untitled=False Window=(Default)	AutoKeyword=False	AutoContextID=True Name=Heading 5 Script=(None) TopicType=Glossary Term Type=Level 5	AutoButtons=False AutoLink=True AutoNavigate=True AutoNext=False ExplicitAccess=False MidTopic=False
Jump	ContentsOnly=False Nonscrolling=False OnlineOnly=True Popup=False Replacement=Untitled=False Window=(Default)	AutoKeyword=False	AutoContextID=False Name=Jump Script=(None) TopicType=(Inherit from Style) Type=Body text	AutoButtons=False AutoLink=False AutoNavigate=False AutoNext=False ExplicitAccess=False MidTopic=False
MidTopic	ContentsOnly=False Nonscrolling=False OnlineOnly=False Popup=False Replacement=Untitled=False Window=(Default)	AutoKeyword=False	AutoContextID=False Name=MidTopic Script=(None) TopicType=(Inherit from Style) Type=Heading text	AutoButtons=False AutoLink=False AutoNavigate=False AutoNext=False ExplicitAccess=False MidTopic=True
RelatedHead	ContentsOnly=False Nonscrolling=False OnlineOnly=True Popup=False Replacement=Untitled=False Window=(Default)	AutoKeyword=False	AutoContextID=False Name=RelatedHead Script=(None) TopicType=(Inherit from Style) Type=Body text	AutoButtons=False AutoLink=False AutoNavigate=False AutoNext=False ExplicitAccess=False MidTopic=False
WhatsThis	ContentsOnly=False Nonscrolling=False OnlineOnly=False Popup=True Replacement=Untitled=True Window=(Default)	AutoKeyword=False	AutoContextID=True Name=WhatsThis Script=(None) TopicType=(Inherit from Style) Type=Heading text	AutoButtons=False AutoLink=True AutoNavigate=False AutoNext=False ExplicitAccess=True MidTopic=False

Creating New Paragraph Styles

- On the Icon bar, click **Project**.
- Select the **Paragraph Styles** item from the left pane.
- Right-click anywhere in the paragraph style list pane on the right and select **New** from the shortcut menu.

Note that the cursor has moved to the top of the paragraph style list window.

- Type the name of the new paragraph style and press Enter.

Note that the new paragraph style is added to the list.

Alternatively, you can add a new paragraph style by clicking the **Project** menu, choosing **New**, then clicking **Paragraph Style**.

Editing Paragraph Style Properties

- On the Icon bar, click **Project**.
- Select the **Paragraph Styles** item from the left pane.
- In the right pane, select the name of the style to be edited.

Note that the property pane changes to show the properties of the selected style.

- Change the properties as desired.

Note that your changes affect the selected style instantly and need not be saved explicitly. However, the full effect of your changes is not realized until you build the Help target.

Heading 2	
Property	Value
Display	
ContentsOnly	False
Nonscrolling	True
Popup	False
Replacement	
Untitled	False
Window	(Default)
Indexing	
AutoKeyword	False
Misc	
AutoContextID	True
Name	Heading 2
Script	(None)
TopicType	(Inherit from Style)
Type	Level 2
Navigation	
AutoButtons	True
AutoLink	False
AutoNavigate	True
AutoNext	False
ExplicitAccess	False
MidTopic	False

Note: If your Paragraph style has a “parent” Topic Type associated with it, the duplicate Topic Type properties supersede the Paragraph style properties. See Topic Type Properties (page 155).

Using Character Styles

Doc-to-Help extends Word's concept of styles to include Help authoring behavior. For each character style that references a topic, you define a like-named character style in the project file. For example, if you have a Word character style named *BoldText*, you should also have a character style in the project file called *BoldText*. In this way, you can simply select which properties you want to apply to the *BoldText* style in the property pane of the Doc-To-Help project editor.

Character Style Properties

You can edit the Character Style properties by accessing property pane for the specific Character Style. For more information, see [Editing Character Style Properties](#) (page 151).

Display Properties

AffectsAppearance	Determines whether the style defines the appearance of the text (font, color, etc.) in the target help file, or is only used to create a hotspot or keyword and does not affect target appearance.
PageRef	When producing a manual, this property places the page number of the referenced topic next to the text formatted with this style.
Replacement	Sets or returns a string that overrides the default text generated by Doc-To-Help. For a paragraph style with AutoNext set to True, the Replacement string is used instead of the topic title next to the generated button.
Window	Shows the name of the window in which topics formatted in this style are displayed.

Indexing Properties

AutoIndex	When true, this property automatically creates index keywords from text formatted with that character style and associates them with the topic that contains the text.
-----------	--

Miscellaneous Properties

MultiLink	When false, only the first occurrence of a "jump" character style (in a given topic) generates a link. All other formatted occurrences are skipped.
Name	Returns a string containing the name of a specified object (read-only).

Script Displays the script to be run during compilation whenever text formatted with this style is encountered.

Type Determines the help authoring behavior associated with a style at compile time.

Value	Description
None	No hot spot.
Jump	Topic jump hot spot.
Popup	Topic popup hot spot.
Group	Group link hot spot.
Keyword	Keyword link hot spot.

Default Character Styles

	Display Properties	Indexing Properties	Miscellaneous Properties
C1HIndex	NoFormat=False PageRef=False Replacement= Window=(Default)	AutoIndex=True	Name=C1HIndex Script=(None) Type=None
C1HJump	NoFormat=False PageRef=False Replacement= Window=(Default)	AutoIndex=False	Name=C1HJump Script=(None) Type=Jump
C1HPopup	NoFormat=False PageRef=False Replacement= Window=(Default)	AutoIndex=False	Name=C1HPopup Script=(None) Type=Popup

Creating New Character Styles

- On the Icon bar, click **Project**.
- Select the **Character Styles** item from the left pane.
- Right-click anywhere in the character style list in the right pane and select **New** from the shortcut menu.

Note that the cursor has moved to the top of the character style list window.

- Type the name of the new character style and press **Enter**.

Note that the new character style is added to the list.

Alternatively, you can add a new character style by clicking the **Project** menu, choosing **New**, then clicking **Character Style**.

Editing Character Style Properties

- On the Icon bar, click **Project**.
- Select the **Character Styles** item from the left pane.
- In the right pane, select the name of the style to be edited.

Note that the property pane changes to show the properties of the selected style.

- Change the properties as desired.

Note that your changes affect the selected style instantly and need not be saved explicitly. However, the full effect of your changes is not realized until you build the Help target.

Reference	
Property	Value
Display	
NoFormat	False
PageRef	False
Replacement	
Window	(Default)
Indexing	
AutoIndex	False
Misc	
Name	Reference
Script	D2HGlossaryRef
Type	Jump

For more information, see Using Character Styles (page 149.)

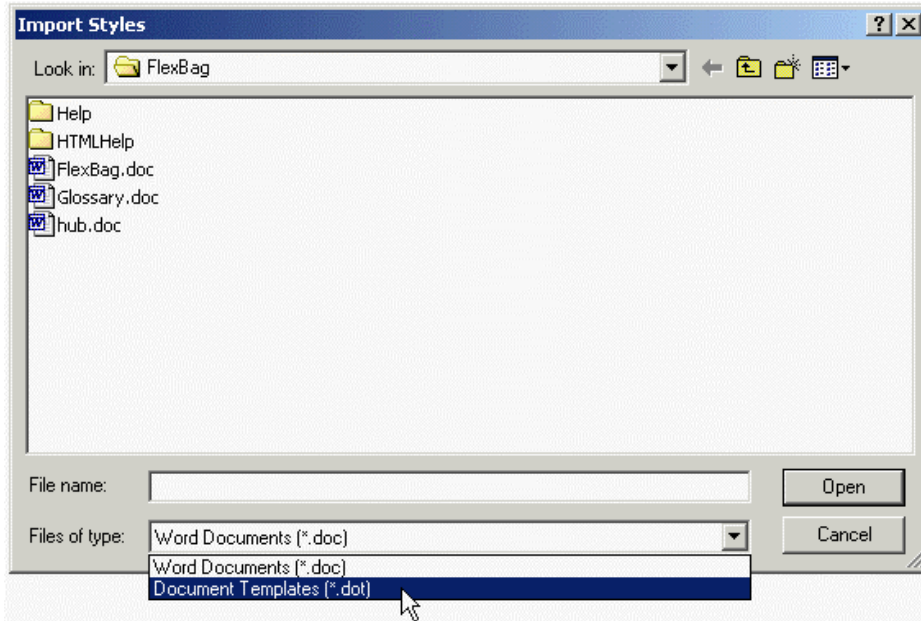
Importing Styles From a Microsoft Word Document or Template

- On the **Project** menu, click **Import Styles**.

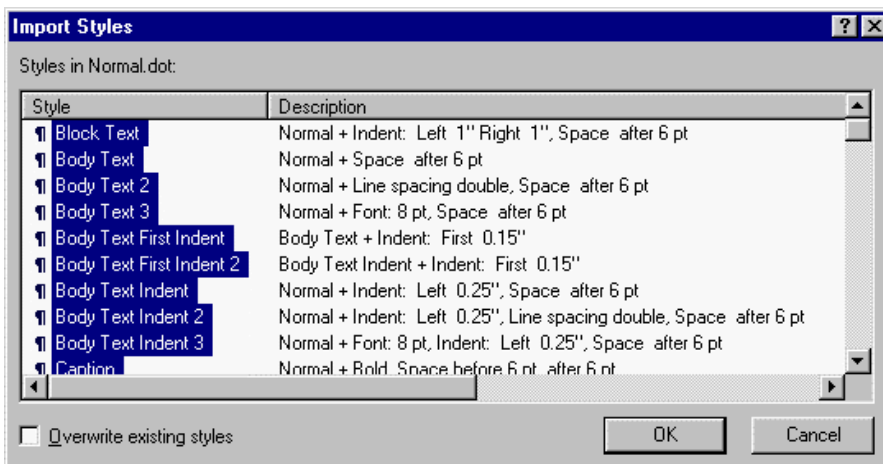
A file selection dialog box opens.

- Specify a folder and file name, then click Open.

You can specify any Microsoft Word document or template; you are not restricted to those that you have added to the project. Simply change the Files of type: drop-down to specify Word Documents or Document Templates.



Doc-To-Help reads all style definitions from the specified file, then displays them in a list as shown in the following figure. Initially, all styles are selected.



By default, a selected style is not imported if a like-named counterpart already exists in the active project file. If desired, check the **Overwrite existing styles** box to force Doc-To-Help to re-import existing styles.

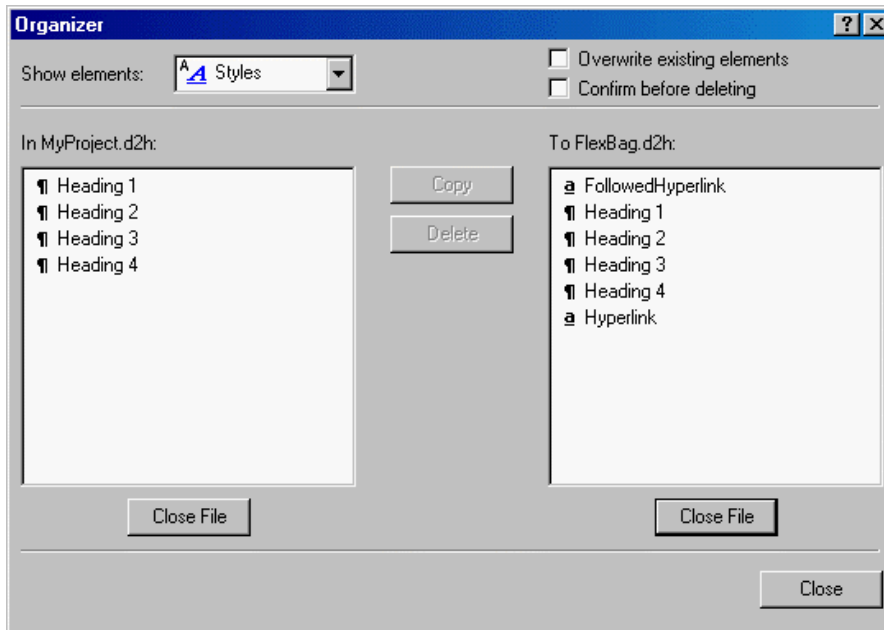
- Click **OK** to import all styles into the active Doc-To-Help project. Alternatively, you can select one or more styles as you would files in Windows Explorer, then click **OK**.

When a paragraph style is imported, it receives the same **Outline level** it had in Microsoft Word. When a character style is imported, its **Link type** is set to *None (index only)*. No other settings are affected.

Importing Styles From Another Project

- On the **Project** menu, click **Organizer**.
The **Organizer** dialog box opens.
- In the **Show elements** list, select **Styles**.
- Click **Open File** on the right side of the Organizer to display a file selection dialog box.
- Specify a folder and file name for the existing project, then click **Open**.

The list on the right side of the Organizer dialog box now contains all styles in the specified project. Initially, none of the styles are selected, and the **Copy** button is unavailable.



By default, like-named styles are not copied from one project to another. If desired, check the **Overwrite existing elements** box to force the Organizer dialog box to re-import existing styles.

- Select one or more styles in the list on the right, then click Copy. When you are finished, click the Close button at the lower right corner of the dialog box.

When a style is copied from one Doc-To-Help project to another, it retains all of its attributes, including any script or window associations. If necessary, the Organizer dialog box also copies the appropriate script module and/or window definition. The **Overwrite existing elements** setting also applies to scripts and windows associated with a style.

Note that you can also use the Organizer dialog box to export styles in the active project to another Doc-To-Help project. When you select one or more styles in the list on the left, the direction of the copy operation is reversed and the list captions change accordingly.

Using Topic Types

A Topic Type is a named collection of topic attributes: what window the Help topic appears in, how it's accessed, whether it gets a map number, where the heading paragraph appears, or if it appears at all. Topics are mapped to Topic Types by the style of their headings. Using Topic Types streamlines the customizing process for you, while Helping maintain consistency. There are two uses for topic types. First, you can use a

Topic Type to customize an individual topic, overriding the style properties. Second, you can use a Topic Type to customize the default behavior of a paragraph style.

Note that Topic Types and Paragraph Styles have many duplicate properties. When a paragraph style has a TopicType associated with it, the duplicate Topic Type properties supercede the Paragraph Style properties.

Topic Type Properties

Edit the Topic Type properties by accessing the property pane for the specific Topic Type. For more information, see [Editing Topic Type Properties](#) (page 158).

Display Properties

ContentsOnly	When true, omits topics defined with this Topic Type from the Help target, but uses them as book titles in the Help contents. This property is duplicated in the Paragraph Styles.
Nonscrolling	When the NonScrolling property of a Topic Type is set to true, any topic headings formatted with that topic type appear in the non-scrolling region of the Help window. This property is duplicated in the Paragraph Styles.
Popup	When the popup property of a Topic Type is set to true, any topic formatted with that topic type appears in a popup window on top of the Help window. This property is duplicated in the Paragraph Styles.
Replacement	Sets or returns a string that overrides the default text generated by Doc-To-Help. For a Topic Type with AutoNext set to True, the Replacement string is used instead of the topic title next to the generated button. This property is duplicated in the Paragraph Styles.
Untitled	When true, the topic title is not displayed in the Help window. This property is duplicated in the Paragraph Styles.
Window	Shows the name of the window in which topics formatted in this style are displayed. This property is duplicated in the Paragraph Styles.

Indexing Properties

AutoKeyword When true, this property automatically creates index keywords from topic titles formatted with that topic type and associates them with the appropriate topic. This property is duplicated in the Paragraph Styles.

Miscellaneous Properties

AutoContextID When true, automatically creates Context IDs for all topics using this paragraph style. This property is duplicated in the Paragraph Styles.

Name Returns a string containing the name of a specified object (read-only). This property is duplicated in the Paragraph Styles.

Script Displays the script to be run during compilation whenever text formatted with this style is encountered. This property is duplicated in the Paragraph Styles.

Navigation Properties

AutoButtons When true, displays subtopic buttons for the topic type with numeric outline levels greater than the parent level. This property is duplicated in the Paragraph Styles.

AutoLink Creates unique link tags for topics formatted with the topic type by enabling the auto link tag feature. This property is duplicated in the Paragraph Styles.

AutoNavigate Determines which topic types are included in the navigation sequence. This property is duplicated in the Paragraph Styles.

AutoNext Enables the next topic button for any active topic type.

ExplicitAccess When true, defines a specific topic type as accessible only through a hyperlink. Topic types defined as explicit are not accessible from the Help contents, the index list, or the text search. This property is duplicated in the Paragraph Styles.

MidTopic

Defines a topic type as an “in topic” jump location similar to an HTML anchor. By defining a topic type as MidTopic and defining a character style as a jump, end users have jump functionality within a lengthy topic. This property is duplicated in the Paragraph Styles.

Default Topic Types

Doc-To-Help comes with eight default topic types. You can modify the properties of these default topic types or create your own as needed. Adding a topic type can be achieved by right-clicking anywhere in the topic type list pane on the right side of the project editor.

	Display Properties	Indexing Properties	Miscellaneous Properties	Navigation Properties
Conceptual	ContentsOnly=False Nonscrolling=True Popup=False Replacement=Untitled=False Window=main	AutoKeyword=False	AutoContextID=True Name=Conceptual Script=(None)	AutoButtons=True AutoLink=True AutoNavigate=True AutoNext=False ExplicitAccess=False MidTopic=False
Contents	ContentsOnly=True Nonscrolling=True Popup=False Replacement=Untitled=False Window=main	AutoKeyword=False	AutoContextID=False Name=Contents Script=(None)	AutoButtons=False AutoLink=True AutoNavigate=True AutoNext=False ExplicitAccess=False MidTopic=False
Glossary of Terms	ContentsOnly=False Nonscrolling=True Popup=False Replacement=Untitled=False Window=main	AutoKeyword=False	AutoContextID=True Name=Glossary of Terms Script=(None)	AutoButtons=False AutoLink=True AutoNavigate=False AutoNext=False ExplicitAccess=False MidTopic=False
Glossary Term Definition	ContentsOnly=False Nonscrolling=False Popup=True Replacement=Untitled=False Window=(Default)	AutoKeyword=False	AutoContextID=False Name=Glossary Term Definition Script=D2HGlossaryTerm	AutoButtons=True AutoLink=True AutoNavigate=False AutoNext=False ExplicitAccess=True MidTopic=False
Margin Note	ContentsOnly=False Nonscrolling=False Popup=True Replacement=Untitled=True Window=(Default)	AutoKeyword=False	AutoContextID=False Name=Margin Note Script=(None)	AutoButtons=False AutoLink=False AutoNavigate=False AutoNext=False ExplicitAccess=True MidTopic=False

Procedural	ContentsOnly=False Nonscrolling=False Popup=False Replacement= Untitled=False Window=proc	AutoKeyword=False	AutoContextID=True Name=Procedural Script=(None)	AutoButtons=True AutoLink=True AutoNavigate=False AutoNext=False ExplicitAccess=False MidTopic=False
Sub-Contents	ContentsOnly=True Nonscrolling=True Popup=False Replacement= Untitled=False Window=main	AutoKeyword=False	AutoContextID=False Name=Sub-Contents Script=(None)	AutoButtons=False AutoLink=True AutoNavigate=True AutoNext=False ExplicitAccess=False MidTopic=False
What's This	ContentsOnly=False Nonscrolling=False Popup=True Replacement= Untitled=True Window=(Default)	AutoKeyword=False	AutoContextID=True Name=What's This Script=(None)	AutoButtons=False AutoLink=True AutoNavigate=False AutoNext=False ExplicitAccess=True MidTopic=False

Adding a New Topic Type

To add a new topic type to the default list:

- On the Icon bar, click **Project**.
- Select the **Topic Types** item from the left pane.
- Right-click anywhere in the Topic Type list in the right pane and select **New** from the shortcut menu.

Note that the cursor has moved to the top of the Topic Type list window.

- Type the name of the new Topic Type and press Enter.

Note that the new topic type is added to the list.

Alternatively, you can add a new topic type by clicking the **Project** menu, choosing **New**, then clicking **Topic Type**.

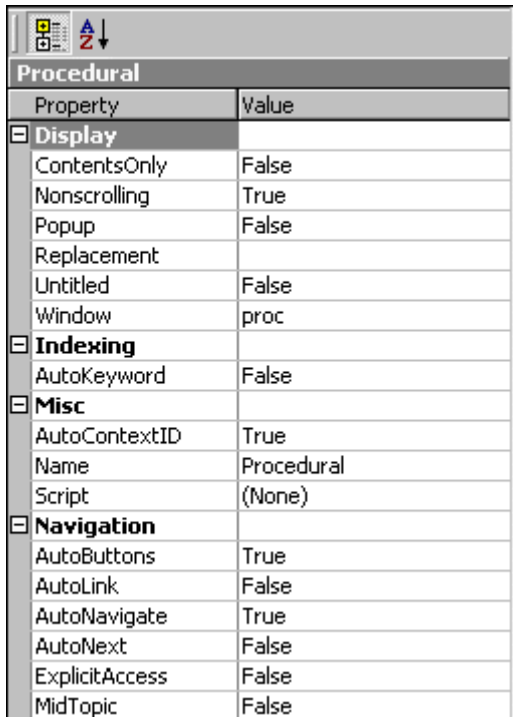
Editing Topic Type Properties

- On the Icon bar, click **Project**.
- Select the **Topic Types** item from the left pane.
- In the right pane, select the name of the Topic Type to be edited.

Note that the property pane changes to show the properties of the selected style.

- Change the properties as desired.

Note that your changes affect the selected Topic Type instantly and need not be saved explicitly. However, the full effect of your changes is not realized until you build the Help target.

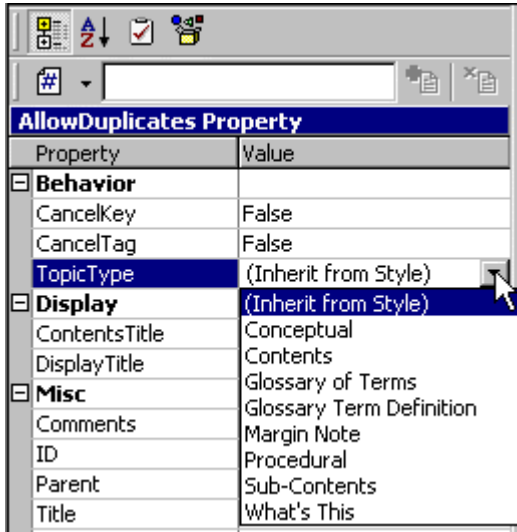


Procedural	
Property	Value
Display	
ContentsOnly	False
Nonscrolling	True
Popup	False
Replacement	
Untitled	False
Window	proc
Indexing	
AutoKeyword	False
Misc	
AutoContextID	True
Name	Procedural
Script	(None)
Navigation	
AutoButtons	True
AutoLink	False
AutoNavigate	True
AutoNext	False
ExplicitAccess	False
MidTopic	False

Applying Topic Types to Topics

- On the Icon bar, click **Project**.
- Select the **Topic Types** item from the left pane.
- In the right pane, select the name of the Topic Type to be edited.
- Note that the property pane changes to show the properties of the selected style.
- In the property pane, use the TopicType property drop-down menu to select the Topic Type you wish to associate with the specific topic.

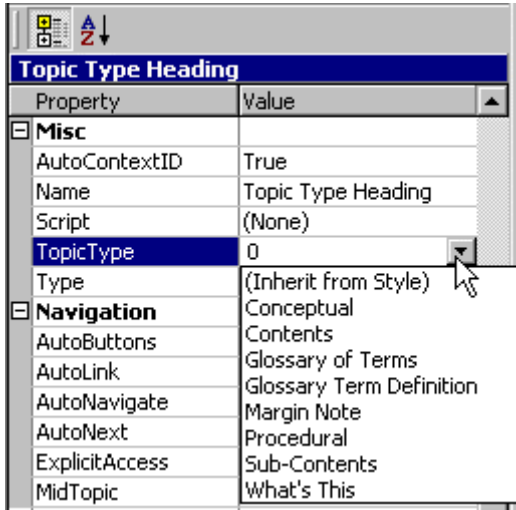
Note that your changes affect the selected Topic Type instantly and need not be saved explicitly. However, the full effect of your changes is not realized until you build the Help target.



Applying Topic Types to Paragraph Styles

- On the Icon bar, click **Project**.
- Select the **Paragraph Styles** item from the left pane.
- In the right pane, select the heading to be edited.
- Note that the property pane changes to show the properties of the selected style.
- Use the TopicType property drop-down menu to select the Topic Type you wish to associate with the specific Paragraph Style.

Note that your changes affect the selected topic type instantly and need not be saved explicitly. However, the full effect of your changes is not realized until you build the Help target.



Using List Styles

Doc-To-Help provides two ways to add lists to you Help targets: Microsoft Word lists and the list style buttons.

For those of you familiar with making lists in Microsoft Word, you will be happy to know that Doc-To-Help accepts all numbered and bulleted lists. Because Doc-To-Help will accept any number of levels or styles, using the Word list functionality is the recommended method when working with documents that require multiple bullet or number styles or more than a few levels.

Alternately, when working with simple lists, there are six list styles available on the Doc-To-Help toolbar for Microsoft Word. These list style buttons allow you to create numbered and bulleted nested lists quickly and easily.



C1H Number Style

Numbers the specified text.



C1H Number 2 Style

Numbers and indents the specified text to nest it within a list.



C1H Bullet Style

Formats the specified text with a bullet.



C1H Bullet 2 Style

Bullets and indents the specified text to nest it within a list.

**C1H Bullet 2A Style**

Bullets and indents the specified text to nest it within a list.

**Continue List**

Formats the specified text as an unnumbered list at the same level as the previous numbered list.


Creating Lists with Doc-To-Help List Styles

- On the Icon bar, click Project.
- In the left pane, select **Documents**.
- Select and right-click a document from the right pane, and select **Open** from the shortcut menu.
- Highlight the text in your document to which you want to apply a list style.
- On the Doc-To-Help toolbar, click one of the style buttons.

Alternatively, you can apply the desired style directly by selecting it in the Style list of the Formatting toolbar or from the **Format | ComponentOne Doc-To-Help Styles** menu command.

For more information, see *Applying Lists in Doc-To-Help* (page 90) and *Working With Styles in Word* (page 102).

Nesting Lists with Doc-To-Help List Styles

- On the Icon bar, click Project.
- In the left pane, select **Documents**.
- Select and right-click a document from the right pane, and select **Open** from the shortcut menu.
- Highlight the text in your document to which you want to apply the bulleted list style.
- Click the **C1H Bullet 2 Style** button .

Doc-To-Help applies a nested bulleted list.

Alternatively, you can apply the desired style directly by selecting it in the Style list of the Formatting toolbar or from the **Format | ComponentOne Doc-To-Help Styles** menu command.

For more information, see *Applying Lists in Doc-To-Help* (page 90) and *Working With Styles in Word* (page 102).

Using Lists in the Printed Manual Target

Microsoft Word has a number of known issues concerning the preservation of numbered list formatting when copy/paste and other text manipulations are performed. These problems are generally restricted to Word 2000 and are largely solved in Word 2002 (XP). They originate in the sophisticated manner in which Word tries to make numbered list behavior as smart and intuitive as possible. As a result, numbered lists are sometimes re-numbered and re-formatted incorrectly when you perform copy/paste and other modifications.

For this same reason, the numbering and formatting of some lists can be broken when compiling the master document (print target) with the Concatenate property is set to **True**. This problem is impossible to predict and may be difficult to find, so when the FixLists property is set to default (Word 2000 Only), Doc-To-Help automatically fixes, re-numbers and re-formats lists where it detects this problem in the resulting document.

However, on very rare occasions, when Doc-To-Help attempts to fix this problem, it can cause Word to crash, freeze or corrupt the resulting document. In these rare cases, if you encounter such problems while the Doc-To-Help progress indicator reads "Restoring lists formatting...", you may want to make Doc-To-Help skip this action.

Set the **FixLists** property to **Never** if you want to skip the fix for Word 2000.

Set the **FixLists** property to **Always** if you want to force the fix for Word 2002.

Working with Cascading Style Sheets

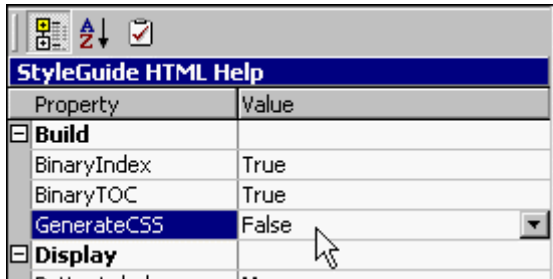
When compiling any HTML based Help target, Doc-To-Help produces a Cascading Style Sheet file (.css) based on the template that is attached to your Help target. This file governs the appearance of your HTML Help targets in much the same way as a Microsoft Work template governs the appearance of a Work document. In most cases, there is no need to do anything to the CSS file, Doc-To-Help modifies it as needed during each compilation of the Help target.

Modifying the Cascading Style Sheet

In some cases the Help author may wish to modify the CSS file or even replace it with customized CSS file of their own. To make sure that Doc-To-Help does not overwrite the modified CSS file, simply set the GenerateCSS property to **False**.

- On the Icon bar, click **Project**.
- Select the **Help Targets** item from the left pane.

- In the right pane, select the Help target to be edited.
- In the Property Pane, change the GenerateCSS property **False**.



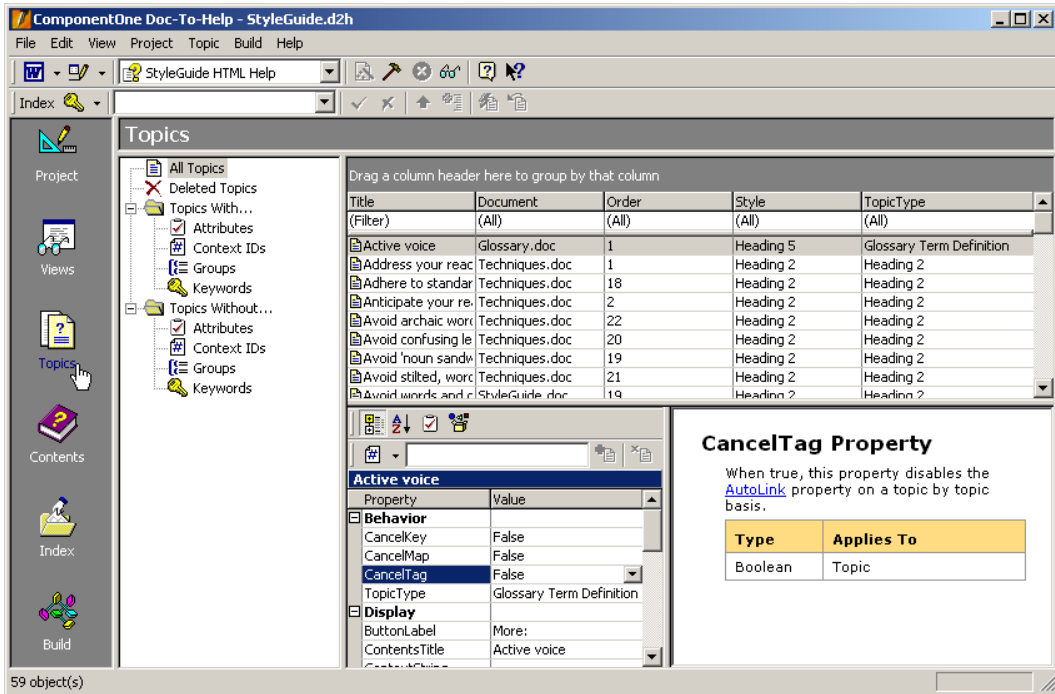
Note: Cascading Style Sheets are only used when creating HTML output.

Defining and Organizing Topics

The Topics Icon

The **Topics** Icon displays a grid view of the Help topics derived from the source documents during compilation. It also provides an interface for setting properties of individual topics, such as index keywords and context-sensitive Help IDs. The left hand pane of the project editor contains a tree list allowing the author to view specific types of topics. You can view all topics, deleted topics or topics with and without Attributes, Context IDs, Groups and Keywords. Additionally, you can sort the topics in the right project editor pane by using the convenient outlook style grouping and filtering features provided with ComponentOne Doc-To-Help 7.5.

Since Doc-To-Help derives topics from your source documents, it does not provide an explicit way for you to create a new topic from scratch. Instead, you create topics indirectly by applying a paragraph style to one or more source document paragraphs in Microsoft Word, defining a like-named active paragraph style in your Doc-To-Help project if one does not already exist, then compiling the affected documents either individually or by building the Help target. After compilation, you can view the newly created topics and edit their properties in the **Topics** window.



The term *active paragraph style* denotes a style that Doc-To-Help uses to define topics during compilation. Any style with a numeric outline level in Microsoft Word is active. For example, the predefined Doc-To-Help styles *Heading 1*, *Heading 2*, and *Heading 3* are active, while the built-in Word styles *Normal*, *Body Text*, and *List Bullet* are not. If you have your own document templates that use other styles to format topic titles, you can import them into a Doc-To-Help project so that they are recognized during compilation.

Topics do not formally exist in a Doc-To-Help project file until you build a Help system, at which time each source document is compiled as follows:

- Doc-To-Help saves a copy of the document as an RTF (Rich Text Format) file, which is placed in a folder created by Doc-To-Help. Your original document is never modified. Once created, the RTF copy is not modified, although Doc-To-Help may delete it during subsequent builds.
- For each paragraph in the RTF file, Doc-To-Help compares the name of the associated style against the names of the active paragraph styles defined in the Doc-To-Help project.
- For each paragraph that matches an active style, Doc-To-Help adds a topic record to the project file using the text of the entire paragraph as the topic title. (If the topic was already encountered during a prior build, Doc-To-Help does not add a new record.)

- Doc-To-Help generates one or more additional intermediate files in a format that is appropriate for the current Help target. These files include suitable mark-up symbols derived from the topic records and other elements of the project file.

Once all source documents have been compiled, Doc-To-Help combines all marked-up intermediate files into a single entity. For WinHelp and HTML Help, this step involves running a Microsoft compiler. For other Help targets, Doc-To-Help generates ancillary files (such as HTML framesets) as needed.

Notes: Only the topic title should be formatted with the active paragraph style. In the topic you are now reading, the paragraph above containing the words *Defining and Organizing Topics* is formatted with the *Heading 1* style, but the remaining paragraphs are formatted with the *Normal* style.

Topic titles must be no more than 255 characters long, or else a compile-time error occurs.

Topic titles defined by a paragraph style with a *Level 1* type must be unique within a document. That is, a single document can contain multiple topics named *Example* if each is formatted with a different level 1 paragraph style, but it cannot contain multiple topics named *Example* that are formatted with the same style. The only way to have multiple like-named topics that are formatted with the same style is to place each in a separate source document. These restrictions are enforced at compile time.

Topic Properties

By selecting any of the Topics in the right pane, the property pane displays the property settings for that topic.

Behavior Properties

CancelKey	When true, this property disables the AutoKeyword property on a topic by topic basis.
CancelMap	When true, this property disables the AutoContextID property on a topic by topic basis.
CancelTag	When true, this property disables the AutoLink property on a topic by topic basis.
TopicType	Defines a pre-determined collection of attributes to a topic (i.e. what window the Help topic appears in, how the Help topic is accessed, whether it gets a map number).

Display Properties

ButtonLabel	Specifies the text that precedes the subtopic buttons on a topic by topic basis. Set this property to an empty string to suppress the default More: label.
ContentsTitle	Allows for modification of the topic titles in the Help window contents area without changing the actual topic title in the source document.
ContextString	When using context-sensitive and dynamic help with Microsoft Help 2.0, specifies a context string for the topic.
DisplayTitle	Allows for modification of the topic title in respect to Help file searches without changing the actual topic title in the source document.

Miscellaneous Properties

AsciiName	Specifies the ASCII-only string used to identify the topic in situations where non-ASCII characters are not allowed.
Comments	When activated, this property provides an editable textbox to be used for comments by the Help author. These comments are not accessible by the end user.
ID	Returns a unique numeric identifier assigned to the topic during compilation (read-only).
Parent	Shows the ID number of the related parent topic.
Title	Returns the topic title (read-only).
URL	Specifies the name of the generated .htm file for a topic.
WinHelpMacro	Specifies a macro to run when a Help target, a topic, or secondary window is opened.

Modular Properties

MergeContents	For a placeholder topic in a modular hub project, this property specifies the platform-dependent contents file to be loaded dynamically.
MergeFile	For a placeholder topic in a modular hub project, this property specifies the platform-dependent Help file to be loaded dynamically.

MergeTitle

For a placeholder topic in a modular hub project, this property specifies the text used to clarify like-named topics in keyword search lists.

Navigation

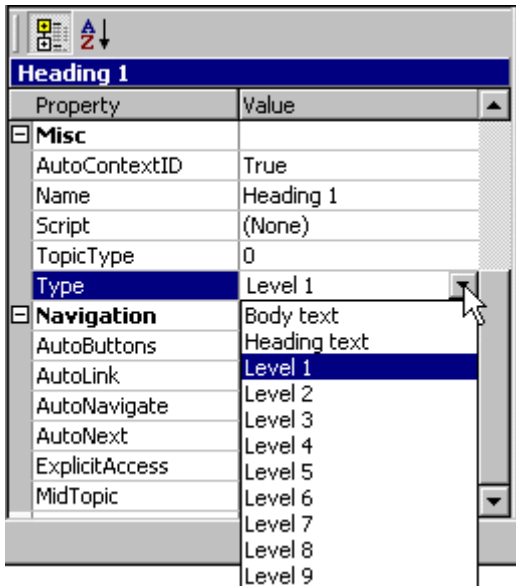
RelatedJumps

Controls the order of automatic and/or custom jumps on a topic by topic basis.

Making a paragraph style active

In order for a paragraph style used in a source document to define a topic during compilation, it must be marked as active within the Doc-To-Help project file as follows:

- On the Icon bar, click **Project**.
- Select the **Paragraph Styles** item from the left pane.
- In the right pane, select the name of the style to be edited.
- In the property pane, scroll down to the Type property and set the heading type to a value other than **Body text**.



When you create a new paragraph style, the default outline level is *Body text*. The *Body text* setting is the only one that does not mark a paragraph style as active. Like the numeric level settings, it is analogous to its counterpart in Microsoft Word, and is provided so that you can easily import styles from a Word document or template without turning every paragraph into a separate topic. You can also use this setting to

disable an active style. For example, if your Help system does not need three levels of topics, you can set the outline level of *Heading 3* to *Body text*.

The *Heading* text style makes the style active but excludes associated topics from the Help contents. This setting is typically used to implement ancillary popup topics such as figures, glossary terms, and code samples.

The settings *Level 1* through *Level 9* also make the style active. In addition, these settings determine the hierarchy of associated topics within the Help contents. The predefined Doc-To-Help styles *Heading 1*, *Heading 2*, and *Heading 3* are set to *Level 1*, *Level 2*, and *Level 3*, respectively.

For more information on paragraph styles, see *Using Paragraph Styles* (page 143).

Specifying the Help Contents

Initially, Doc-To-Help derives the Help contents automatically based upon the hierarchy of topics in the project. Therefore, in order for a topic to be included in the Help contents, the paragraph style that defines it must have a numeric outline level.

To assign a numeric outline level to a paragraph style:

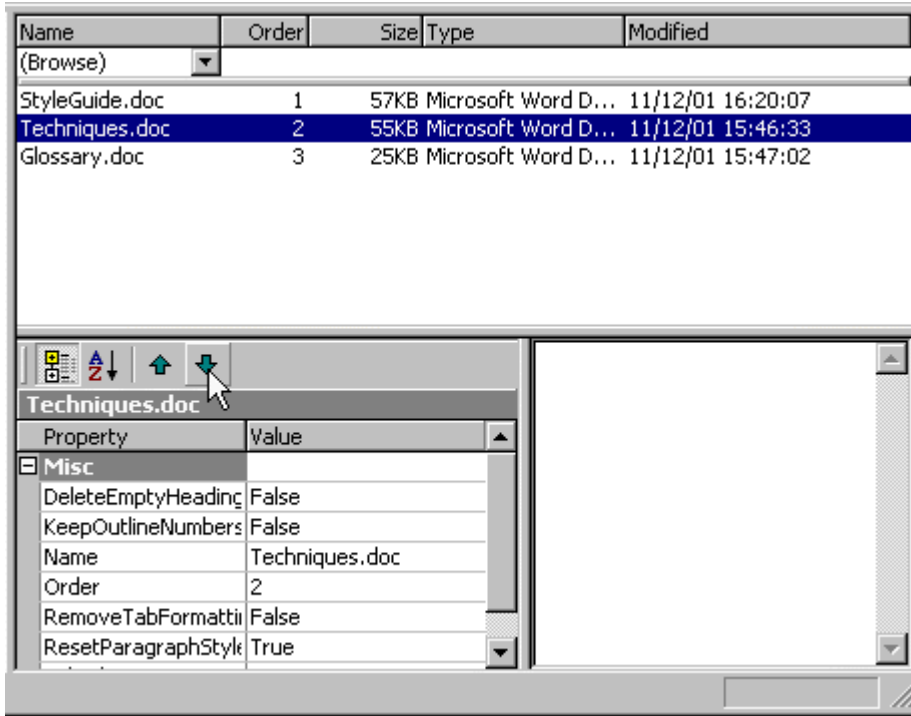
- On the Icon bar, click **Project**.
- Select the **Paragraph Styles** item from the left pane.
- Select the paragraph style to be edited from the right pane.
- In the property pane, change the Type property to a value between *Level 1* and *Level 9*.

To modify the Help contents using the Project Editor, see *Modifying the Help Contents* (page 180).

If a project contains multiple source documents, the order in which they were added to the project determines the order in which they are listed in the Help contents. Since this may not be the order you intended, you can rearrange it.

To rearrange the order of the documents and therefore the top level Help topics contents:

- On the Icon bar, click **Project**.
- Choose **Documents** from the left pane.
- Choose the document you wish to move from the right pane.
- Either change the Order property in the property pane or press the up or down arrow located in at the top of the documents property pane.



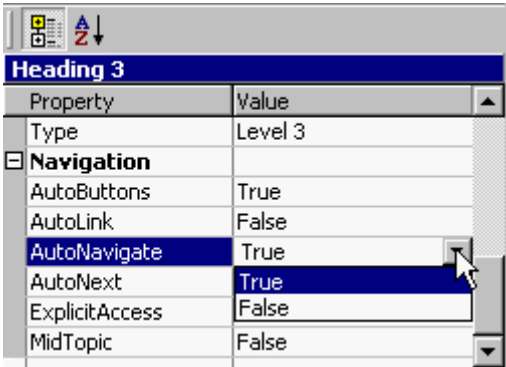
Note: This scheme works best if each document starts with a Level 1 heading.

Specifying the Navigation Sequence

Doc-To-Help does not provide an interface for defining arbitrary topic navigation sequences, but instead derives them automatically based upon the order of topics in each source document. By default, only paragraph styles with a numeric outline level participate in the navigation sequence.

To control whether topics created with an active style are navigable:

- On the Icon bar, click **Project**.
- Select the **Paragraph Styles** item from the left pane.
- Select the paragraph style to be edited from the right pane.
- From the property pane, modify the AutoNavigate property value as desired. Setting the value to True allows the topic with the selected style to be navigable.

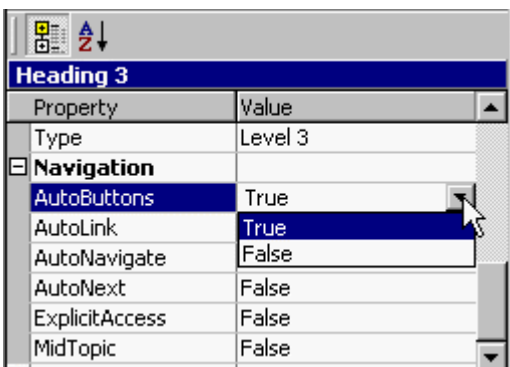


Providing Links to Subtopics

For paragraph styles with numeric outline levels greater than *Level 1*, Doc-To-Help can automatically display a list of subtopic buttons at the end of each parent topic.

To enable subtopic buttons for a paragraph style:

- On the Icon bar, click **Project**.
- From the left pane, select the **Paragraph Styles**.
- From the right pane, select the paragraph style to be edited.
- From the property pane, modify the AutoButtons property value as desired. Setting the value to True allows subtopic buttons to be displayed at the end of the parent topic.

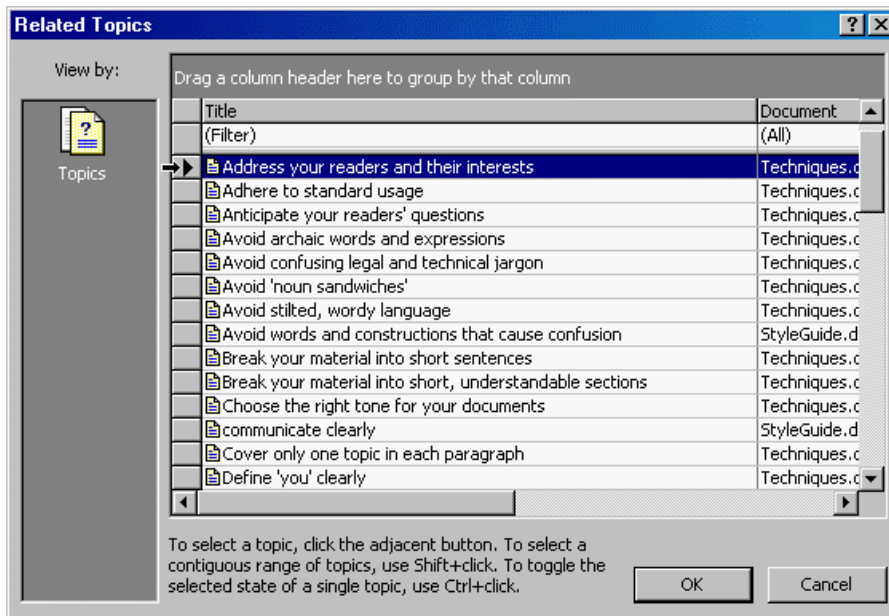


Providing Links to Specific Topics

Along with providing automatic links to subtopics at the end of a parent topic, Doc-To-Help allows you to customize the list of links by using the **Related Topics** command. This command adds additional, author selected topic links, to the end of a parent topic.

To add author selected topic links to the end of a topic:

- On the Icon bar, click **Topics**.
- From the right pane, select and right-click the parent topic where you want to add a topic link.
- From the shortcut pane, click **Related Topics**.
- From the Related Topics dialog box, select the topic you wish to jump to by clicking on one or more of the record selector buttons to the left of the topic titles.



Note: Doc-To-Help supports multiple topic selection by using the Shift and Ctrl keys. Simply press and hold the Shift key, for consecutive selections, or the Ctrl key, for nonconsecutive selections, and choose the record selector buttons to the left of the topic titles.

- Once you have made your selection, click **OK**.

Note: The OK button is disabled unless you select your topics by using the record selectors. Simply, selecting the topic in the title column will not enable the OK button.

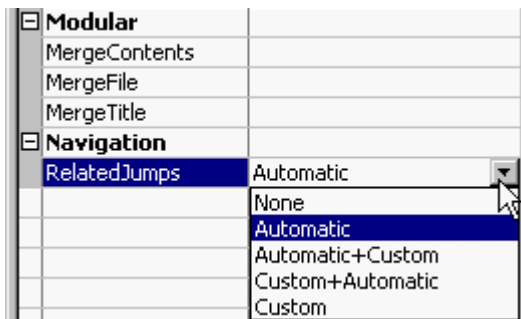
- Upon compilation, the selected topic link is added to the end of the parent topic.

Customizing the Related Topic Jumps

Doc-To-Help provides you with the option to customize which topic links are available at the end of a parent topic. By using the RelatedJumps property, you can determine whether your automatic and author specified topics are available and in what order they are presented.

To customize the topics:

- On the Icon bar, click Topics.
- From the right pane, select the parent topic you wish to edit.
- From the topics properties pane, select RelatedJumps and click the option you want from the drop-down menu.



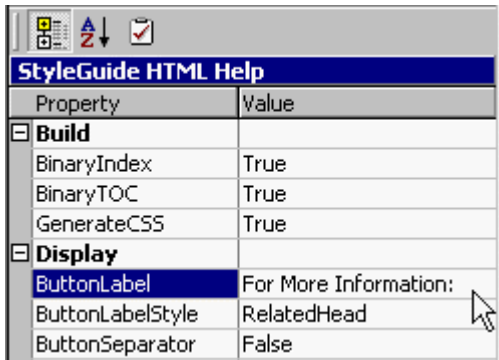
Value	Description
None	No Jumps are displayed
Automatic	Only Automatic jumps are displayed
Automatic + Custom	Automatic then Custom jumps are displayed
Custom + Automatic	Custom then Automatic jumps are displayed
Custom	Only Custom jumps are displayed

Labeling Subtopic Links

In cases where you want to insert a descriptive heading for the subtopic buttons, Doc-To-Help provides the ButtonLabel property. The default setting for this property is **More**; but you can modify the default easily. To modify the default topic:

- On the Icon bar, select **Project**.

- In the left pane, click **Help Targets**.
- In the right pane, select the Help Target you want to modify.
- In the property pane, locate the ButtonLabel property and enter the text you wish to use as your label to the right of the property name.



The changes you made to the ButtonLabel property produce results as shown in the illustration below.

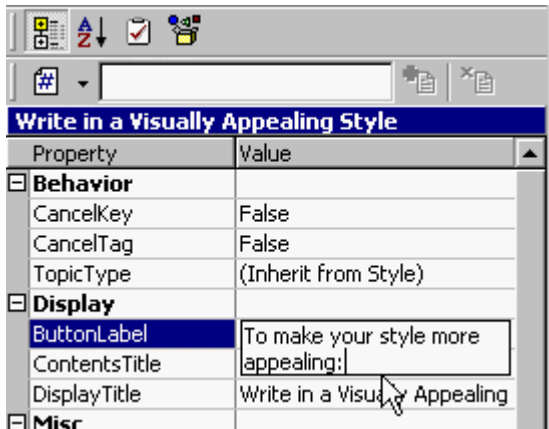
For More Information:

- Use short sentences
- Write to one person, not to a group
- Use the simplest tense you can
- Use 'must' to convey requirements
- Place words carefully
- Use 'if-then' tables
- Avoid words and constructions that cause confusion
- Use contractions when appropriate

Modifying the Subtopic Label for a Specific Topic

In some instances, you may wish to override the Help target's existing ButtonLabel property, for a single topic.

- On the Icon bar, select **Topics**.
- In the right pane, select the topic you want to modify.
- In the property pane, locate the ButtonLabel property and enter the text you wish to use as your label to the right of the property name.



The changes you made to the ButtonLabel property produce results as shown in the illustration below.

To make your style more appealing:

- Use lots of informative headings
- Write short sections
- Include only one issue in each designated paragraph
- Use vertical lists
- Use tested emphasis techniques to highlight important points

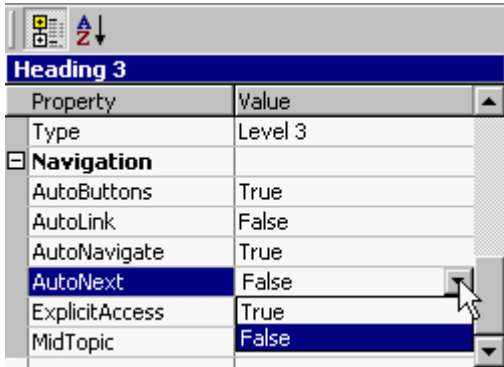
Note: When modifying the ButtonLabel property for a topic, the changes are limited to that topic only. The remaining Subtopic labels will contain the text specified in the Help target ButtonLabel Property.

Providing Links to the Next Topic

For any active paragraph style, Doc-To-Help can display a button at the end of each associated topic that links to the next topic in the source document, even if the next topic does not participate in the navigation sequence.

To enable the next topic button for a paragraph style:

- On the Icon bar, click **Project**.
- Select the **Paragraph Styles** item from the left pane.
- Select the paragraph style to be edited from the right pane.
- From the property pane, modify the AutoNext property value as desired. Setting the value to True allows a button that links to the next topic to be displayed.



You can use this setting to chain together a series of topics that are meant to be read in sequence. Although the reader can generally achieve the same effect with a built-in navigation button, providing an explicit link at the end of each topic makes your intentions clear.

You can also use this setting to link master topics with detail topics without affecting the navigation sequence. For example, suppose that a source document contains a set of function descriptions at the *Heading 2* level, each of which is immediately followed by a code example at the *Heading 3* level. If you enable *AutoNext* for *Heading 2*, to create links to the examples, disable *AutoNavigate* for *Heading 3*, to remove it from the navigation sequence, and assign a secondary window to *Heading 3*, then the reader can step through the function descriptions and click the generated button to view a particular code sample in a separate window.

Note: In HTML Help the BinaryTOC property must be set to True for topic to topic navigation.

In WinHelp the OneBrowseSequence property must be set to True for topic to topic navigation.

Editing Topic Properties

- On the Icon bar, click **Topics**.
- Select **All Topics** or any of the other topic choices available from the left pane.
- In the right pane, select the name of the topic to be edited.

Note that the property pane changes to show the properties of the selected topic.

- Change the properties as desired.

Note that your changes affect the selected topic instantly and need not be saved explicitly. However, the full effect of your changes is not realized until you build the Help target.

For more information, see [Topic Properties](#) (page 167.)

Adding a Link Tag to a Topic

In order for a topic to be a destination for a jump or popup link, it must have a unique link tag that identifies it. Link tags can be generated automatically by paragraph styles, programmatically by scripts, or manually by the author.

To enable automatic link tags for a paragraph style:

- On the Icon bar, click **Project**.
- Select the **Paragraph Styles** item from the left pane.
- In the right pane, choose the paragraph style to be edited.
- In the property pane, set the AutoLink property value to **True**.

Link tags created in this manner are identical to the topic title except that spaces, hyphens, and periods are converted to underscores; letters, numbers, and underscores are unchanged; and all other characters are removed. For example, the topic title:

`BackCol or Property (TextBox)`

yields the link tag:

`BackCol or_Property_TextBox`

Examining Link Tags

In order for a link to a topic to work, the link tag of the destination topic in Doc-to-Help must exactly match the text of the hot spot in Word. For example, if you created a hot spot in Word with the text **help topics**, then the link tag of the destination topic must be **help_topics**.

When using link tags, spaces, hyphens and periods must be converted to underscores; letters, numbers and underscores are unchanged; and all other characters must be removed. For example, the topic title:

`6th St. Exit`

yields the link tag:

`6th_St_Exit`

In this example the topic title contains a period next to a space, so you must provide an underscore for both the period and the space.

To determine the link tag of your destination topic:

- On the Icon bar, click **Topics**.
- Select the **All Topics** item from the left pane.
- In the right pane, choose the destination topic.
- Click the **Choose Collection** button on the Topics Property Toolbar and choose **Link Tag** from the **Collection Type** drop-down.

The link tags are listed next to the **Link Tags** collection.

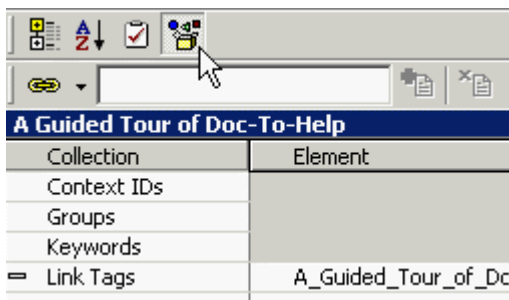
- Make sure that one of the link tags exactly matches the text of your hot spot. If none does, you must create a matching link tag.

For more information, see [Adding a Link Tag to a Topic \(page 178\)](#) and [Manually Inserting Link Tags \(page 179\)](#).

Manually Inserting Link Tags

To manually insert a link tag:

- On the Icon bar, click **Topics**.
- Select the **All Topics** item from the left pane and select the desired topic for editing.
- Click the **Choose Collection** button on the Topics Property Toolbar and choose **Link Tag** from the **Collection Type** drop-down.



- Type the Link Tag into the **Element Textbox** and press **Enter**.

Note that the Link Tag has been associated with the highlighted topic and is displayed to the right of the Link Tag collection.

When using link tags, spaces, hyphens and periods must be converted to underscores; letters, numbers and underscores are unchanged; and all other characters must be removed. For example, the topic title:

6th St. Exit

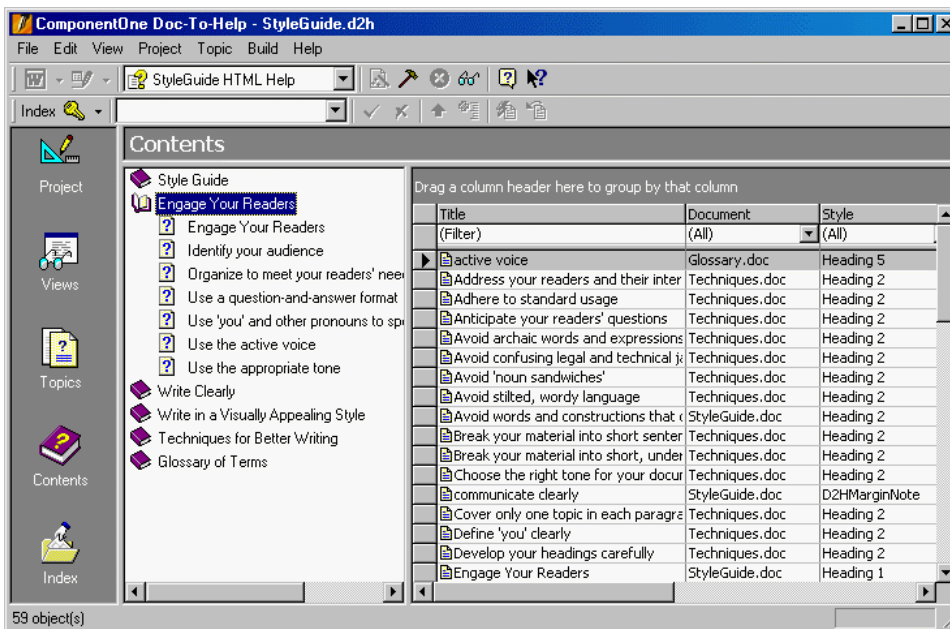
yields the link tag:

6th_St_Exit

In this example the topic title contains a period next to a space, so you must provide an underscore for both the period and the space.

Modifying the Help Contents

The **Contents** Icon displays the hierarchy of Help topics as determined by the outline level of the defining styles. This hierarchy is used to build the contents for all online Help targets. Note that the left pane contains the same books and topics as the Contents section in the compiled Help file, and the right pane displays a sorted list of topics as in the Topics window. The topics can be sorted and filtered in the same manner as in the Topic window. Doc-To-Help supports multiple topic selection by using the *Shift* and *Ctrl* keys. Simply press and hold the **Shift** key, for consecutive selections, or the **Ctrl** key, for nonconsecutive selections, and choose the record selector buttons to the left of the topic titles.



Doc-To-Help's Contents Window is an interface in which the final Help system's contents can be fully customized. The ability to redefine the default topic hierarchy, create new

books, insert and delete topics, and perform common editing functions is available through this view. The interface utilizes toolbars, drag-and drop editing, and right-click shortcut menus to make the editing of contents easier and you more productive.

Using the Contents Toolbar

On the **View** menu, point to **Toolbars**, then click **Contents**. A check mark next to the command indicates that the toolbar is displayed

After a Doc-To-Help project has been compiled, you can use the contents toolbar to cut, copy, paste, delete, and rearrange contents and the topics.

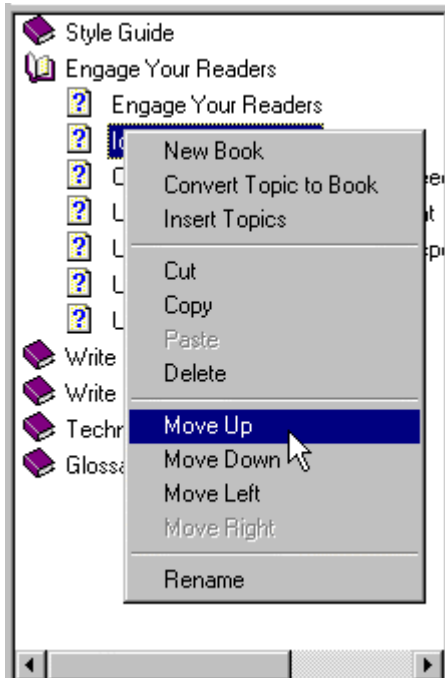


From left to right, the Contents toolbar contains the following commands:

New Book	Adds a new book to the Help contents
Convert Topic	Changes the selected topic into a book
Insert Topics	Inserts the selected topic(s) from the right pane into the Help contents
Cut	Removes the selection and places it on the clipboard
Copy	Copies the selection to the clipboard.
Paste	Inserts the contents of the clipboard at the insertion point, replacing any selection
Delete	Deletes the selection
Move Up	Moves the selection up within the outline
Move Down	Moves the selection down within the outline
Move Left	Moves the selection left within the outline
Move Right	Moves the selection right within the outline

Editing the Help Contents

Even though the Help contents can be fully edited in the Contents window through the use of the toolbar, it can also be edited through Doc-To-Help's right-click shortcut menu. Right-clicking a book or a topic brings up the shortcut menu. This menu conveniently opens and provides all of the functions of the toolbar. Using the right-click shortcut menu helps to speed up the time it takes to rearrange and edit your contents.



Rearranging the Help Contents

Upon compilation, the outline for the final Help system is displayed in the left pane of the Contents window. This outline is completely customizable in respect to its structure. The four directional commands on the toolbar or the “move” commands in the shortcut menu are used to alter the outline structure.

To move a topic or book up or down in the outline:

- Select the topic or book to be moved in the left pane.
- Use the move up or move down commands either from the contents toolbar or the shortcut menu to change the position of the topic or book.

To move a topic or book right or left in the outline:

- Select the topic or book to be moved in the left pane.
- Use the move left or move right commands either from the contents toolbar or the shortcut menu to change the position of the topic or book.

Notice that these commands cannot change a book into a topic or vice versa, but that they merely move the item around the outline.

Creating a New Book

- In the Contents window, right-click anywhere in the left pane.
- Choose **New Book** from the menu or select the new book command from the contents toolbar.

A blinking cursor appears and allows you to customize the name of the new contents item.

Notice that if you right-click the dead area, choose new book from the shortcut menu, and no book is selected in the left pane, then Doc-To-Help creates a new book after the last book in the outline.

Converting a Subtopic into a Book

Doc-To-Help also allows you to convert an existing subtopic into a main topic. The topic remains in the same position in the Help system's outline, but is converted from a topic to a book.

- In the Contents window, select a subtopic that you would like to be converted into a book.
- Click the **Convert Topic** command in the toolbar.

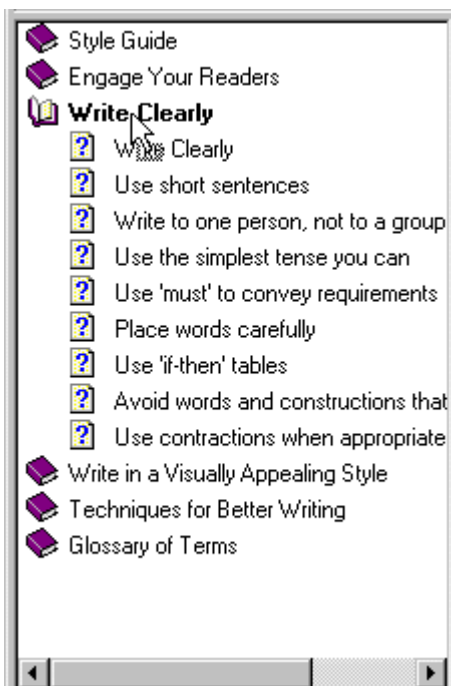
Alternatively, you can right-click the topic you want to convert into a book and select **Convert Topic to Book** from the shortcut menu.

Adding Topics to the Help Contents

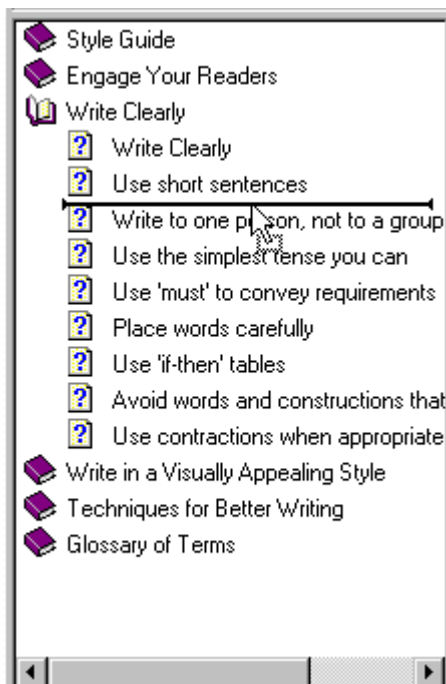
Upon compilation, all of the available topics that Doc-To-Help derives from the source documents are placed into the right pane of the Contents window. This is an inclusive list, so it includes topics already associated with books or main topics. The Contents window interface allows you to add these topics to different books in the left pane with ease.

Adding topics to the Help contents using the drag-and-drop method

- In the Contents window, click the topic record selector in the right pane that you would like to associate with a topic in the left pane.
- Drag the topic to the left pane.
- If you drag the topic over an existing book, the book becomes bold indicating that the topic will be associated with that book if you drop it there.



- If you drag it between books or topics, a bold line appears indicating the location where the new topic will be placed, if dropped.



- Drop the topic in the left pane, or over the book with which you would like it associated.

Note: Doc-To-Help supports multiple topic selection by using the Shift and Ctrl keys. Simply press and hold the Shift key, for consecutive selections, or the Ctrl key, for nonconsecutive selections, and choose the record selector buttons to the left of the topic titles.

Adding topics to the Help contents using the contents toolbar

- Select a book in the left pane.
- Select the topic in the right pane to be associated with that book.
- Click the Insert Topics command on the toolbar or select Insert from the shortcut menu in the right pane.

The new topic is now associated under the book that was selected in the left pane. Note that if you selected a topic in the left pane, the new topic is inserted above that topic.

Renaming Books and Topics

- Click the book or topic to be renamed in the left pane of the Contents window.
- Right-click in the left pane and bring up the shortcut menu.
- Select **Rename** from the shortcut menu.

A blinking cursor appears and allow you to customize the name of the new book or topic.

Saving the Help Contents

Doc-To-Help allows for the saving of the current contents without recompilation. The changes to the contents are stored within the active project file itself. The **Save Contents** menu item is available and can be used to save only while working in the Contents window.

- Open the **File** menu from the menu bar.
- Select the menu option to **Save Contents**.

In addition, while working in the Contents window, if you attempt to close Doc-To-Help without saving, you are asked if you would like to save the Contents before exiting. This ensures that any changes to your contents are deliberate and properly retained.

Context Sensitive Help

Context-sensitive Help is Help that is sensitive to the context from which it is called — that is, it “knows” what you’re doing when you ask for Help and offers assistance on performing that specific task.

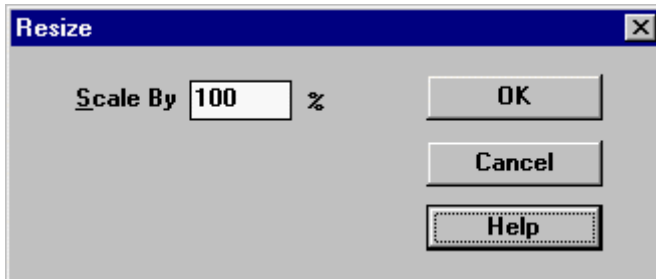
Before context-sensitive Help, integrating Help into an application was easy. The Help author wrote Help topics about an application, and when a user needed to access Help, she went to the Help menu, brought up the contents screen and navigated through the Help file until she found what she wanted.

For very simple applications, this model of Help authoring works well. Many users and programmers, however, prefer and demand a higher level of online Help. Help files, like the programs they document, have grown larger over the years. It is no longer convenient for a user to navigate through an entire Help file to find the section she needs. In a larger application, she may well never find it.

Enter context-sensitive Help. By reacting to the state of the application from which it is called, context-sensitive Help eliminates most of the navigation through the Help, searching for the relevant topics.

The Help Button

At its most basic level, context-sensitive Help can be a Help button on a dialog box.



To view the Help topic for the entire dialog box, the user can simply click the Help button. There's no longer any need to search through an entire Help file. The jump to the correct topic is automatic and painless.

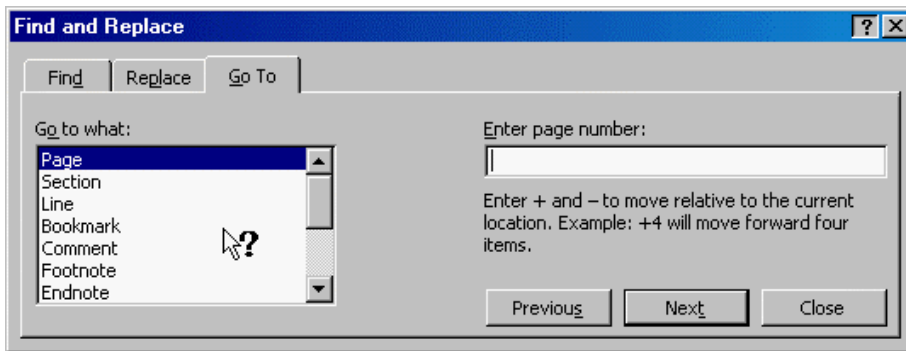
This method works well in most situations, but it has one drawback. Suppose the dialog box has a large number of text boxes, drop-down list boxes, command buttons, etc. The Help for that screen can rapidly exceed the space provided for by one Help screen (and Help topics that fill more than one Help screen are harder to read).

What's This Help

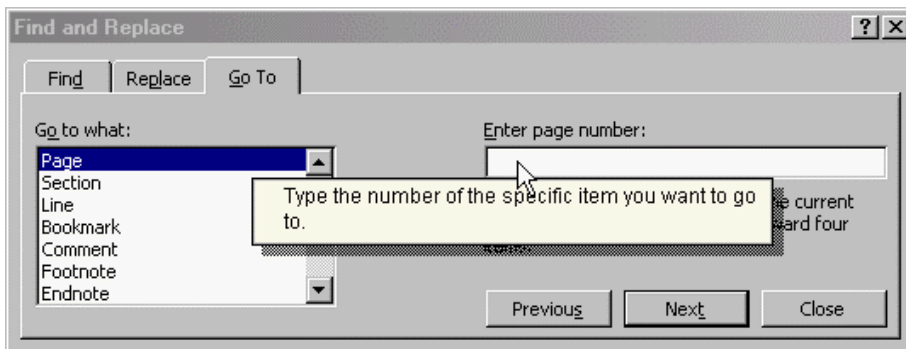
A more finely tuned approach to context-sensitive Help allows the user to access popup Help topics for individual text boxes, drop-down list boxes, command buttons, etc. This sort of field-level Help is known as "What's This?" help.

For most applications, there are three ways to access "What's This" help in Windows:

- The user can press **F1** to see popup Help for the currently selected item in the window.
- The user can right-click the window item and click the "What's This" menu item in the popup menu that opens.
- The user can click the "What's This" button in the title bar. The cursor becomes a special context-sensitive Help cursor: an arrow with a question mark attached to it.



When you click an item in the dialog box, Windows displays a popup Help topic for that item.



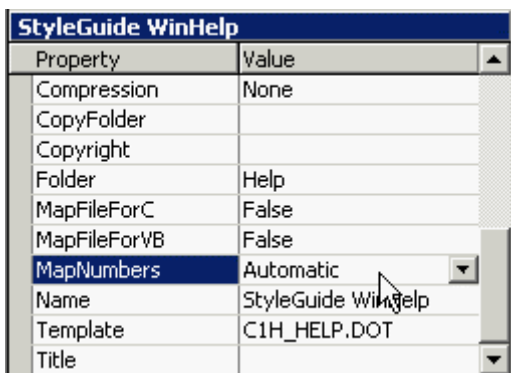
Creating Context-Sensitive Help

Context-sensitive Help involves associating topics in your Help project with specific items in a software application. The two most popular methods of making this association are to either have the Help author supply the application developer with topic ID numbers, which he uses as part of his application, or to have the developer supply the Help author with context ID numbers. In this second method, the Help author matches the context ID numbers with the topics in the Doc-To-Help project editor. For more information, see Adding Context IDs With the Topic Properties Toolbar (page 190).

Generating Context ID Numbers

To expedite the first method where the Help author needs to supply the applications developer with topic ID numbers, Doc-To-Help assigns a unique number to each Help topic. In this way, the developer can use these topic ID numbers as the handle through which an application calls the topic in the Help file. Please Note that for this method to be functional, the MapNumbers property of the current Help target must be set to Automatic or Automatic+Custom and the AutoContextID of the related paragraph headings must be set to True.

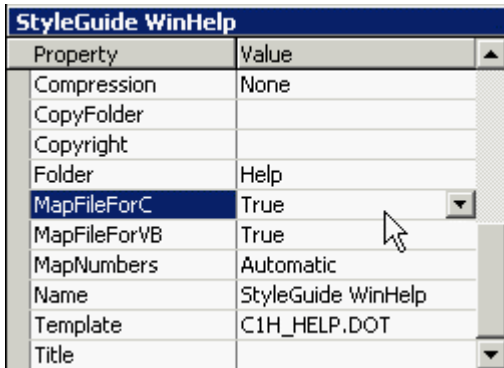
- On the Icon bar, click **Project**.
- Select the **Help Targets** item from the left pane.
- Select the name of the desired Help target.
- In the property pane, set the MapNumbers property value to **Automatic**.



To facilitate the transfer of topic ID numbers to developers, you can generate Visual Basic or C/C++ header files in Doc-To-Help by toggling the MapFileForVB and MapFileForC properties in the Help target's property pane to True.

- On the Icon bar, click **Project**.
- Select the **Help Targets** item from the left pane.

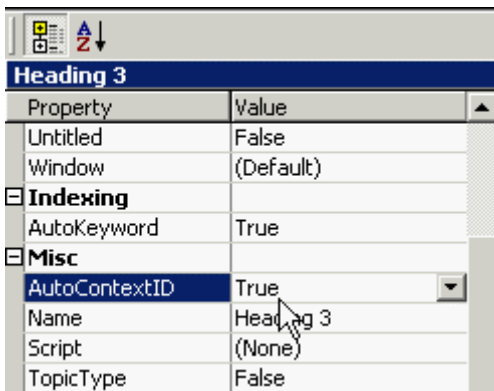
- Select the name of the desired Help target.
- In the property pane, set the MapFileForC and the MapFileForVB property values to **True**.



Note: Not all Help targets have the option to generate both VB and C header files.

To generate a context ID number:

- On the Icon bar, click **Project**.
- Select the **Paragraph Styles** item from the left pane.
- In the right pane, select the heading for which you want to generate context IDs.
- In the property pane, set the AutoContextID property value to **True**.

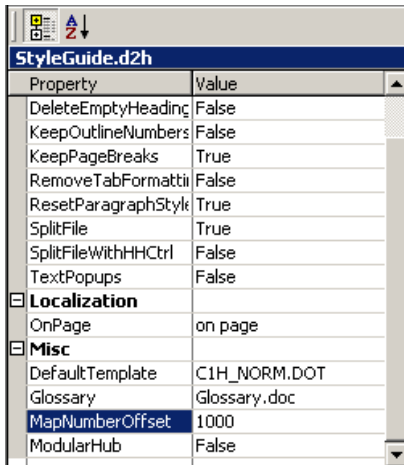


Customizing your Automatically Generated Context ID Numbers

In some cases, like modular Help applications, you may want to customize your Context Ids to start at a specific number. By using the MapNumberOffset property. When set,

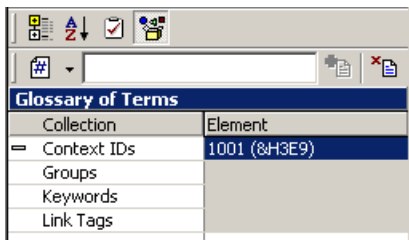
Doc-To-Help adds the value associated with this property with the automatically generated Context ID to create a customized number series.

- On the Icon bar, click **Project**.
- Select **Project Settings** from the left pane.
- From the property pane, set the MapNumberOffset property to the value you want to use for your automatic Context IDs.



- Rebuild your Help Project.

Note that your automatically generated context IDs have the MapNumberOffset value added to the context id number.

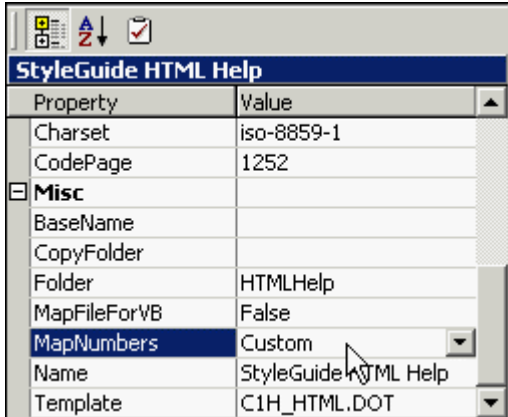


Adding Context IDs With the Topic Properties Toolbar

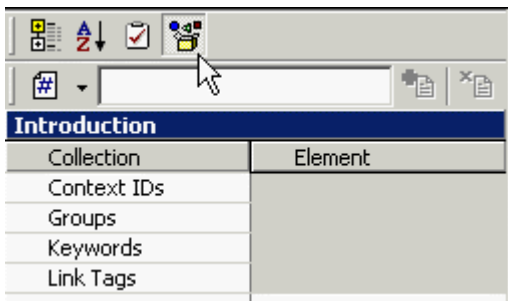
After a Doc-To-Help project has been compiled, you can use the topic properties toolbar to insert context IDs. To associate a developer-supplied Help context ID with a topic:

- On the Icon bar, click **Project**.
- Select the **Help Targets** item from the left pane.
- From the right pane, choose the Help target to be edited.

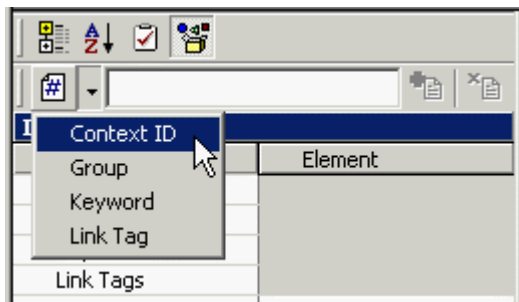
- From the property pane, set the MapNumbers property value to **Custom**.



- Click the **Topics** icon on the Icon toolbar, select the **All Topics** item and select the desired topic in the right pane.
- Click the **Choose Collections** button on the Topics Property Toolbar.



- Select Context ID from the Collection Type drop-down.



- Type the context ID, provided by the developer, into the Element Textbox and press Enter.

Note that the context ID has been associated with the highlighted topic and is displayed to the right of the **Context IDs** collection.

Note: You can assign multiple context IDs to a single topic, but you cannot assign a single context ID to multiple topics.

Sorting Topics For Easy Viewing With Outlook Style Grouping

Doc-To-Help provides the Help author with enhanced topic display and sorting capabilities in the topic list pane on the right by using a convenient outlook style grouping feature. By simply dragging any of the column headings (Title, Document or Style) into the grouping area located at the top of the topic list pane, Doc-To-Help sorts automatically by that heading. To sort by headings:

- On the Icon bar, click **Topics**.
- Select the **All Topics** item from the left pane.
- Select the desired topic from the right pane.
- In the right pane, select the "Style" column from the header bar, and while holding down the mouse button, drag its header to the column grouping area as shown below.

re to group by that column				
Title	Document	Order	Style	TopicType
(Filter)	(All)	(All)	(All)	(All)
Active voice	Glossary.doc	1	Heading 5	Glossary Term Def
Address your readers and their interests	Techniques.doc	1	Heading 2	Heading 2
Adhere to standard usage	Techniques.doc	18	Heading 2	Heading 2
Anticipate your readers' questions	Techniques.doc	2	Heading 2	Heading 2
Avoid archaic words and expressions	Techniques.doc	22	Heading 2	Heading 2
Avoid confusing legal and technical jargon	Techniques.doc	20	Heading 2	Heading 2
Avoid 'noun sandwiches'	Techniques.doc	19	Heading 2	Heading 2
Avoid stilted, wordy language	Techniques.doc	21	Heading 2	Heading 2
Avoid words and constructions that cause	StyleGuide.doc	19	Heading 2	Heading 2
Break your material into short sentences	Techniques.doc	10	Heading 2	Heading 2
Break your material into short, understand	Techniques.doc	25	Heading 2	Heading 2
Choose the right tone for your document:	Techniques.doc	9	Heading 2	Heading 2
communicate clearly	StyleGuide.doc	27	Margin Note	Margin Note
Cover only one topic in each paragraph	Techniques.doc	26	Heading 2	Heading 2
Define 'you' clearly	Techniques.doc	6	Heading 2	Heading 2
Develop your headings carefully	Techniques.doc	24	Heading 2	Heading 2

- Release the mouse button and note the change in the Doc-To-Help topic list pane. The rows are now sorted by the "Style" heading.

Style	Title	Document	Order	TopicType
(All)	(Filter)	(All)	(All)	(All)
Heading 1	Engage Your Readers	StyleGuide.doc	3	Heading 1
	Glossary of Terms	Glossary.doc	0	Glossary of Terms
	Style Guide	StyleGuide.doc	0	Heading 1
	Techniques for Better Writing	Techniques.doc	0	Heading 1
	Write Clearly	StyleGuide.doc	10	Heading 1
Heading 2	Write in a Visually Appealing Style	StyleGuide.doc	21	Heading 1
	Address your readers and their interests	Techniques.doc	1	Heading 2
	Adhere to standard usage	Techniques.doc	18	Heading 2
	Anticipate your readers' questions	Techniques.doc	2	Heading 2
	Avoid archaic words and expressions	Techniques.doc	22	Heading 2
	Avoid confusing legal and technical jargon	Techniques.doc	20	Heading 2
	Avoid 'noun sandwiches'	Techniques.doc	19	Heading 2
	Avoid stilted, wordy language	Techniques.doc	21	Heading 2
	Avoid words and constructions that cause	StyleGuide.doc	19	Heading 2
	Break your material into short sentences	Techniques.doc	10	Heading 2
Break your material into short, understand	Techniques.doc	25	Heading 2	

- Select the "Document" column and drag its header to the column group area. Note that the topics are sorted by Style, then Document as shown below.

Style	Document	Title	Order	TopicType
(All)	(All)	(Filter)	(All)	(All)
Heading 1	Glossary.doc	Glossary of Terms	0	Glossary of Terms
	StyleGuide.doc	Engage Your Readers	3	Heading 1
		Style Guide	0	Heading 1
		Write Clearly	10	Heading 1
		Write in a Visually Appealing Style	21	Heading 1
Heading 2	StyleGuide.doc	Techniques for Better Writing	0	Heading 1
		Avoid words and constructions that cause	19	Heading 2
		Identify your audience	4	Heading 2
		Include only one issue in each designated	24	Heading 2
		Introduction	2	Heading 2
		Organize to meet your readers' needs	5	Heading 2
		Place words carefully	17	Heading 2
		Test	16	Heading 2
		Use a question-and-answer format	6	Heading 2
		Use contractions when appropriate	20	Heading 2
Use the following	18	Heading 2		

- By simply dragging the headers back down into the header bar you can return to the default topic view.

Sorting to View Order by Document

In some cases, you may need to view the order of the topics within each document. Sorting by the document, then order columns, provides you with this view.

- To do this, select the document column from the header bar, and while holding down the mouse button, drag its header to the column grouping area.

- Next, select the order column from the header bar, and while holding down the mouse button, drag its header to the column grouping area.

Note that the topics are sorted by document, then order as shown below.

Document	Order	Title	Style	TopicType
(All)	(All)	(Filter)	(All)	(All)
Glossary.doc	0	Glossary of Terms	Heading 1	Glossary of Terms
	1	Active voice	Heading 5	Glossary Term Def
	2	Jargon	Heading 5	Glossary Term Def
StyleGuide.doc	0	Style Guide	Heading 1	Heading 1
	1	Version History	Heading 2	Heading 2
	2	Introduction	Heading 2	Heading 2
	3	Engage Your Readers	Heading 1	Heading 1
	4	Identify your audience	Heading 2	Heading 2
	5	Organize to meet your readers' needs	Heading 2	Heading 2
	6	Use a question-and-answer format	Heading 2	Heading 2
	7	Use 'you' and other pronouns to speak di	Heading 2	Heading 2
	8	Use the active voice	Heading 2	Heading 2
	9	Use the appropriate tone	Heading 2	Heading 2
	10	Write Clearly	Heading 1	Heading 1
	12	Use short sentences	Heading 2	Heading 2
12	Write to one person, not to a group	Heading 2	Heading 2	

Filtering Topics in Doc-To-Help


Doc-To-Help provides the author with topic filtering capabilities by using the filter boxes located beneath the headers in the topic list in the right pane. By simply choosing one or more of the drop-down filtering options, or by entering text in the topic textbox, you can filter the topics in any fashion you like. To filter the topics:

- On the Icon bar, click **Topics**.
- Select the **All Topics** item from the left pane.
- To filter by Topic, simply type the text you want to use as your filter into the textbox under the Title header and press **Enter**.

Drag a column header here to group by that column				
Title	Document	Order	Style	TopicType
Identify	(All)	(All)	(All)	(All)
How do you identify passive sentences?	Techniques.doc	8	Heading 2	Heading 2
Identify your audience	StyleGuide.doc	4	Heading 2	Heading 2

- To filter by Document or Style, click the arrow under the respective header and choose the item you want to use as your filter.

Drag a column header here to group by that column

Title	Document	Order	Style	TopicType
Identify	Techniques.doc	(All)	(All)	(All)
 How do you identify passive sentences?	Techniques.doc	8	Heading 2	Heading 2

Note that by combining the Title, Document and Style filtering options, you can sort by almost any combination you like as shown above.

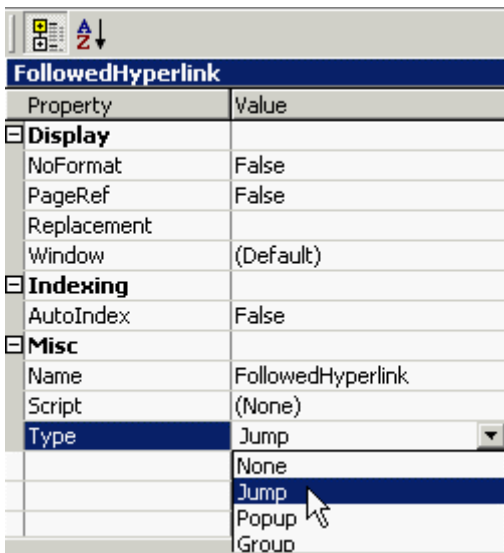
Links and Hot Spots

After Creating Styles in Microsoft Word and defining like-named styles within the ComponentOne Doc-To-Help 7.5 project, you are ready to define the type of links you wish to use. To do this, you need to define the active link type (simply the Type property) for each character style in your project.

Making a Character Style Active

In order for a character style used in a source document to implement a hot spot in the compiled Help target, it must be marked as active within the Doc-To-Help project file as follows:

- On the Icon bar, click **Project**.
- Select the **Character Styles** item from the left pane.
- In the right pane, select the name of the style to be edited.
- In the property pane, set the Type property to a value other than None (index only).



The active link Type settings are as follows:

Jump	Go to another topic
Popup	Display another topic in a popup window
Group	Show a list of topics associated with symbolic names
Keyword	Show a list of topics associated with index keywords

When you create a new character style, the default link type is None (index only). This default setting is the only one that does not mark a character style as active, which means that you cannot use it to implement hot spots. However, you can use it to automatically generate index entries. Like the Body text setting of paragraph styles, the default link type is provided so that you can easily import styles from a Word document or template.

For more information on using styles, see *Using Styles in Doc-to-Help* (page 143) and *Working With Styles in Word* (page 102).

Creating a Jump to a Topic in Another Document

In online Help targets, a jump is a hyperlink that displays another topic in the current Help window. In printed manual targets, a jump can refer to another topic by its physical page number.

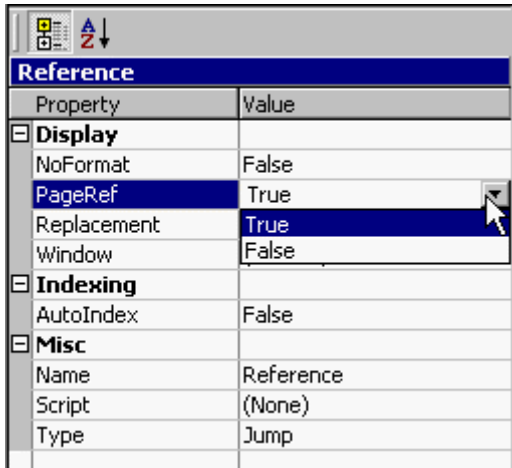
By using the following method you can link to a topic in any document in your project. By applying an active character style to selected text, you can automate linkage from that text to a topic using the identical text for its heading. You can create your own styles or use the built-in character style (C1HJump) for jumps or popups.

- Apply the C1HJump character style to the hot spot text in your Microsoft Word source document. For specific instructions, see *Applying Character Styles in Microsoft Word* (page 104).
- If you don't want to use one of the built-in styles, you can create one of your own. For specific instructions, see *Creating Styles in Word* (page 102).
- If you are using a character style you created, add a like-named character style to your Doc-To-Help project and set its Type property to Jump in the property pane.
- For specific instructions, see *Creating New Character Styles* (page 150) and *Making a Character Style Active* (page 197.)
- Ensure that a link tag of the destination topic exactly matches the text of the hot spot.

For more information, see *Adding a Link Tag to a Topic* (page 178).

To enable automatic page number references for jumps in printed manual targets:

- On the Icon bar, click **Project**.
- Select the **Character Styles** item from the left pane and select a character style whose Type property is set to Jump.
- In the property pane, set the PageRef property value to **True**.



Reference	
Property	Value
Display	
NoFormat	False
PageRef	True
Replacement	True
Window	False
Indexing	
AutoIndex	False
Misc	
Name	Reference
Script	(None)
Type	Jump

Creating Popup Topics

In online Help targets, a popup is a hyperlink that displays another topic in a separate popup window. In printed manual targets, a popup can refer to another topic by its physical page number.

- Identify the character style you want to use to mark the hot spot text in Microsoft Word.
- If you don't want to use one of the built-in styles, you can create one of your own. For specific instructions, see *Creating Styles in Word* (page 102).
- Apply the character style to the hot spot text.

For specific instructions, see *Applying Character Styles in Microsoft Word* (page 104).

- If you haven't already, add a like-named character style to your Doc-To-Help project and set its Type property to Popup in the property pane.

For specific instructions, see *Creating New Character Styles* (page 150) and *Making a Character Style Active* (page 197.)

- Ensure that the link tag of the destination topic exactly matches the text of the hot spot.

For more information, see [Adding a Link Tag to a Topic](#) (page 178).

Note: When adding a popup in WinHelp, set the NonScrolling property of the “target” paragraph style to **False**.

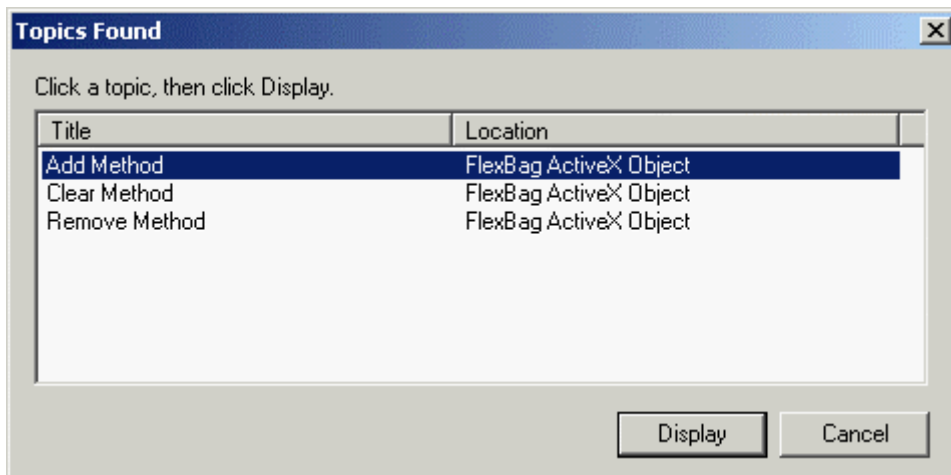
To enable automatic page number references for popups in printed manual targets:

- On the Icon bar, click **Project**.
- Select the **Character Styles** item from the left pane and select a character style whose Type property is set to Popup.
- In the property pane, set the PageRef property value to **True**.

Creating Keyword Links (KLinks)

In online Help targets, a keyword link displays a list of available topics that have been associated with an index keyword.

For example, if you were describing the FlexBag ActiveX Object and created the keyword link *FlexBag* in your document, a user would be able to click *FlexBag* in the Help file, and a dialog box would appear, allowing the user to choose any of the topics that have been associated with *FlexBag*, as shown below.



To create a keyword link:

- Identify the character style you want to use to mark the hot spot text in Microsoft Word.
- If you don't want to use one of the built-in styles, you can create one of your own. For specific instructions, see [Creating Styles in Word](#) (page 102).

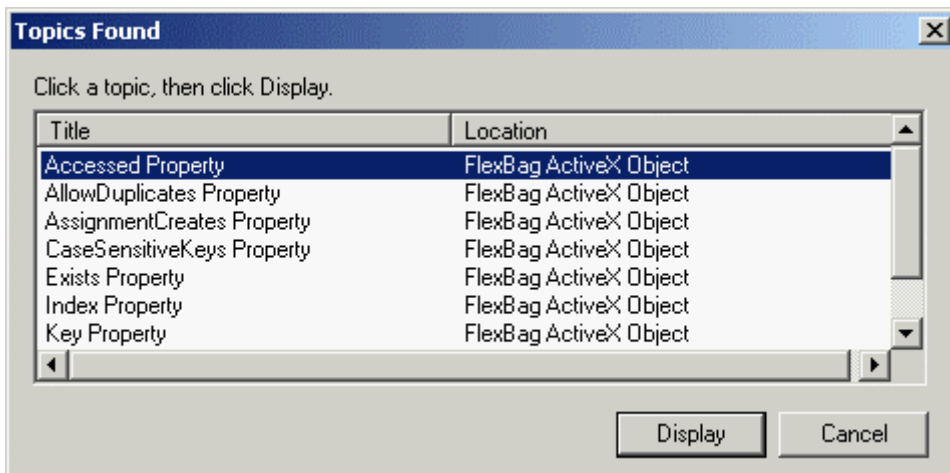
- Apply the character style to the hot spot text.
For specific instructions, see *Applying Character Styles in Microsoft Word* (page 104).
- If you haven't already, add a like-named character style to your Doc-To-Help project and set its Type property to Keyword in the property pane.
For specific instructions, see *Creating New Character Styles* (page 150) and *Making a Character Style Active* (page 197.)
- In the **Index** icon view, set the **Index Collections** button to **Keywords** and ensure that the name of an index keyword exactly matches the text of the hot spot.
For more information, see *Building an Index* (page 219).

Note: Keyword links have no effect in printed manual targets.

Creating Group Links (ALinks)

In online Help targets, a group link (ALink) displays a list of available topics associated with a named group, which is private to the Help project and not visible to the reader.

For example, if you were describing the benefits of the FlexBag ActiveX Object in your document, you might write, “The FlexBag ActiveX Object has *additional properties* which make it possible to manipulate keys more easily.” If you created the group link *additional properties*, a user would be able to click *additional properties* in the Help file, and a dialog box would appear, allowing the user to choose any of the topics that have been associated with *additional properties*, as shown below.



Note: Since additional properties is a group, there is NOT a visible section in your document or Help file called additional properties.

To create a group link:

- Identify the character style you want to use to mark the hot spot text in Microsoft Word.
- If you don't want to use one of the built-in styles, you can create one of your own. For specific instructions, see *Creating Styles in Word* (page 102).
- Apply the character style to the hot spot text.

For specific instructions, see *Applying Character Styles in Microsoft Word* (page 104).

- If you haven't already, add a like-named character style to your Doc-To-Help project and set its Type property to Group.

For specific instructions, see *Creating New Character Styles* (page 150) and *Making a Character Style Active* (page 197.)

- From the **Index** icon view, select **Groups** from the Index dropdown.
- ensure that the name of a topic group exactly matches the text of the hot spot.

For more information, see *Creating Topic Groups* (page 228).

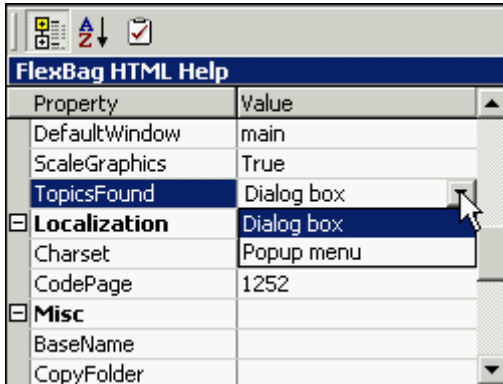
Note: Group links have no effect in printed manual targets.

Modifying the Topics Found Display

When creating group links for HTML Help, Doc-To-Help provides you with the option of modifying how your “Topics Found” message is displayed. By setting the TopicsFound property, your topics found message can display in the standard dialog or in a popup.


To set the TopicsFound property:

- Click the **Project** icon.
- Select **Help Targets** from the left pane.
- From the right pane, select the HTML Help target you want to modify.
- From the property pane, select the TopicsFound property and choose the display option from the drop-down menu.

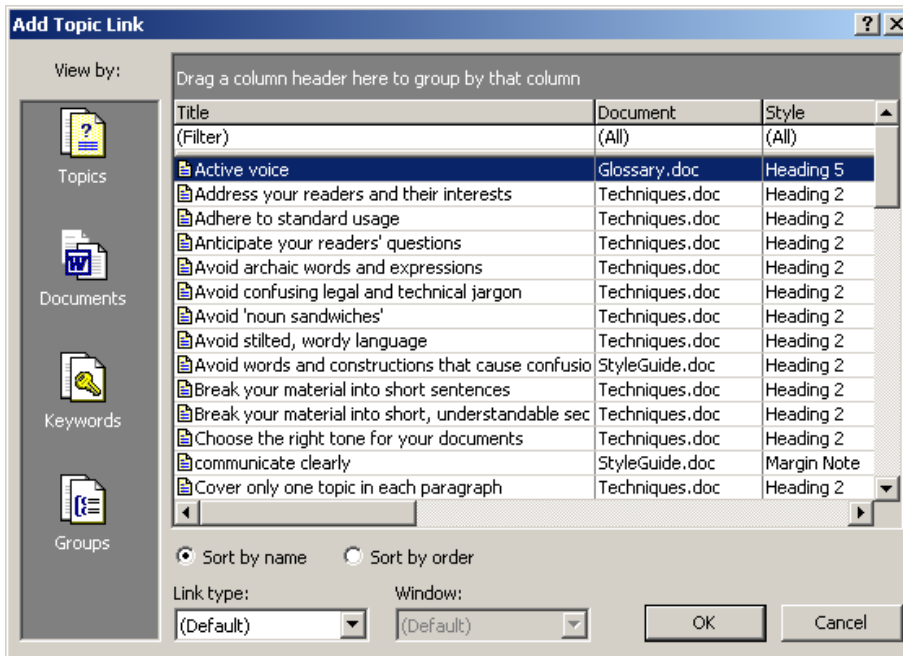


Using the Add Topic Link Dialog Box

By using the Doc-To-Help **Add Topic Link** dialog box, creating topic links and jumps becomes a simple, intuitive procedure. To add a link to a topic using the **Add Topic Link** dialog box:

- In Microsoft Word, highlight the word or words you wish to use as your topic hot spot.
- Click the **Add Topic Link** button .
- From the **Add Topic Link** dialog box, choose the topic you wish to associate with your hot spot.

Note: To facilitate selecting topics, you can use the radio buttons to sort alphabetically or by order (as they appear in each document).



- From the **Link Type:** drop-down, select the **Jump** or **Popup** option.
- From the **Window:** drop-down, select the window type where you wish to display your topic and click **OK**.

For more information on customizing windows, see *Customizing Help Windows* (page 233).

- Save and close the Word document.
- Compile and view your Help project.

Note that the words you highlighted are now defined as a jump to the associated topic.


Alternatively, you can perform the same steps with the **Insert | ComponentOne Doc-To-Help | Topic Link** menu command.

Creating Cross References

By utilizing Doc-To-Help's **Insert Cross Reference** and **Complete Cross Reference** buttons, you can automatically add updateable page references in printed manuals and hyperlink jumps in online Help.

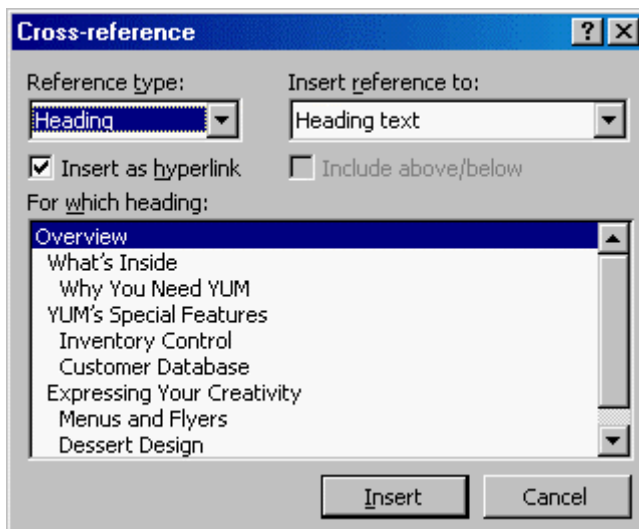
Inserting and Completing a Cross Reference


In Word, position your cursor where you want the reference to appear.

- Type the referring text, such as "For more information, see" or "See also" followed by a space.
- Click the **Insert Cross Reference** button  on the Doc-To-Help toolbar.

The **Cross Reference** dialog box opens.

- Set the **Reference Type** set to *Heading*.
- Set **Insert Reference To** to *Heading Text*.
- Select the heading you want to refer the reader to.



- Click the **Insert** button. The heading text is inserted.
- Click the **Close** button.
- With the insertion point immediately following the cross-reference (reference field), click the **Complete Cross Reference** button  on the Doc-To-Help toolbar.

The page reference is inserted and the heading text is enclosed in quotes.

- Type a period after the page number, if necessary.

Alternatively, you can perform the same steps with the **Insert | ComponentOne Doc-To-Help | Cross Reference** and the **Edit | Complete Cross-reference** menu commands.

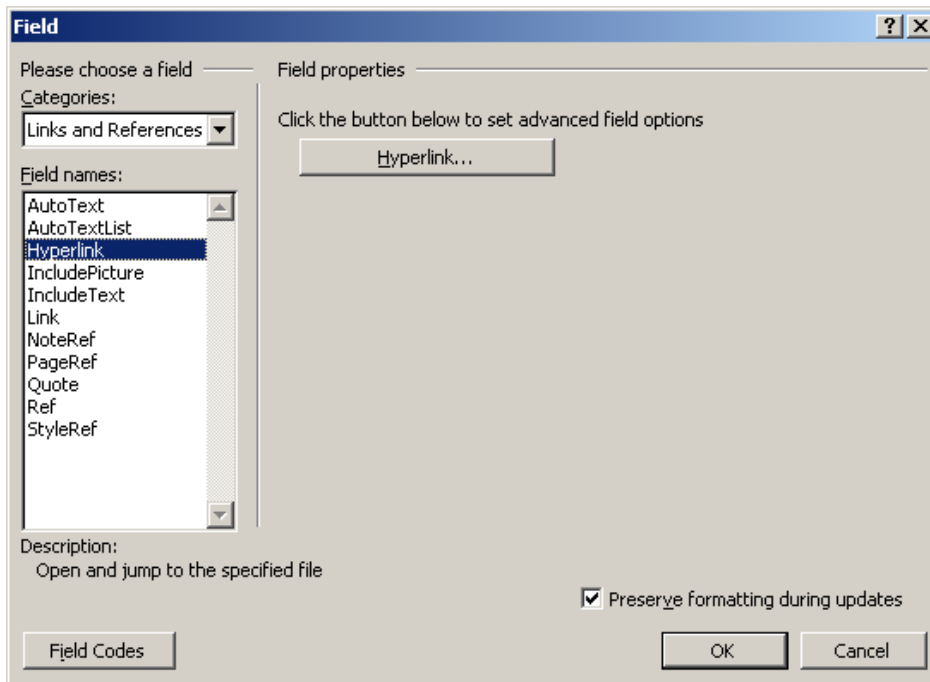
Note: The “on page” text, added automatically by Doc-To-Help, can be modified by changing the OnPage property setting.

To locate the OnPage property, click the Project icon, then select Project Settings from the left pane.

Using the Word Hyperlink Field to Create Cross Document Links

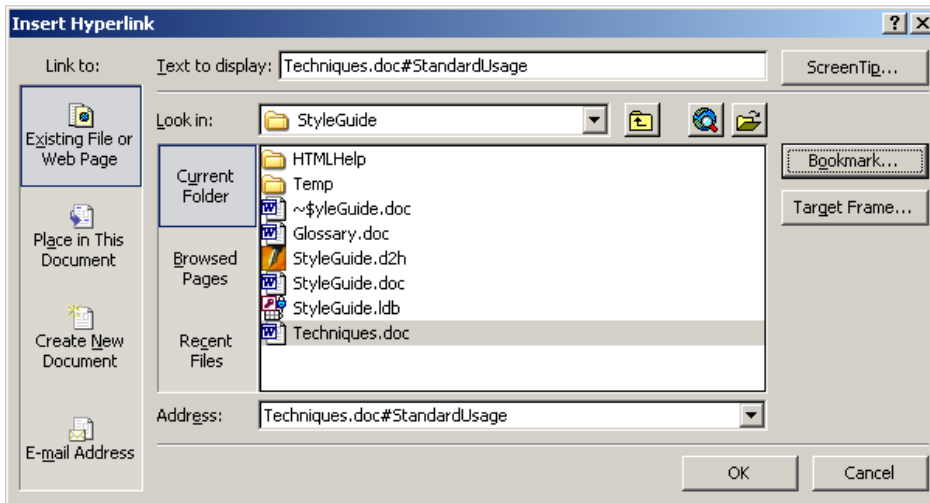
By using the Microsoft Word Hyperlink field you can create cross-references to other documents in your project or to bookmarks within other documents in your project.

- Select **Insert | Field** from the Word menus.
- The **Field** dialog box opens.



- Select **Links and References** from the categories dropdown.
- Select **Hyperlink** from the Field names list.
- Click the **Hyperlink** button.

The **Insert Hyperlink** dialog box opens.



- Select the document for cross-reference.
- If you are referencing a bookmark, click the **Bookmark** button.
- Click **OK**.

Organize to meet your readers' needs

People read documents to get answers. They want to know how to do something or what happens if they don't do something. Organize your document to respond to these concerns. Frequently this means describing events as they occur--you fill out an application to get a benefit, you submit the application, the agency reviews the application, the agency makes a decision on the application.

Think through the questions your readers are likely to ask and then organize them in that order. For regulations, you can organize them into a comprehensive table of contents that will be an outline of the document.

For more information on organization and standardization, see { HYPERLINK "Techniques.doc" \ "StandardUsage" }

Using Margin Notes


The **Margin Note** drop down makes it easy to place text or graphics in the left margin, next to the main body of the text. Margin notes do not appear in the Help unless you explicitly link them to the text, in which case they become pop-ups. Text is automatically formatted to appear in Margin Note style, smaller than Body Text and italicized.

Adding Margin Notes

The **Create Margin Note** command allows you to place text or graphics in the left margin, next to the main body of the text. Margin notes do not appear in the Help unless you explicitly link them to the text, in which case they become pop-ups. Text is

automatically formatted to appear in Margin Note style, smaller than Body Text and italicized.

Place your cursor anywhere in the paragraph you want to annotate.

- Click the **Margin Note** button  on the Doc-To-Help toolbar.
- Select **Create** from the drop-down menu.




A two-column table is placed in the document with the body text in the right hand table cell. If Gridlines is selected in the Table menu, you'll see a dotted outline of the table. Note that the gridlines do not print.

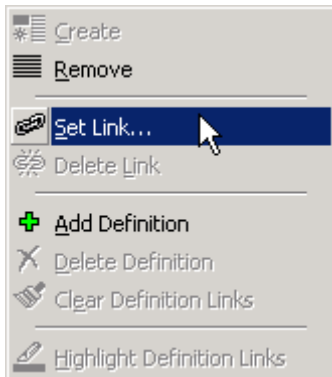
- Type the margin note in the left column of the table.

Alternatively, you can perform the same steps with the **Edit | Margin Note** menu command.

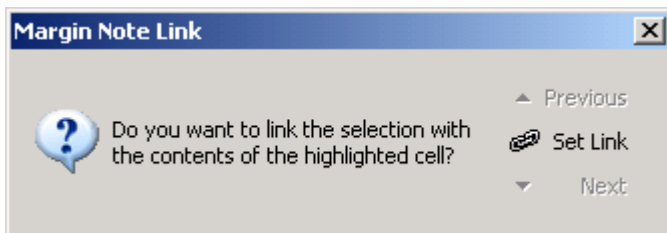
Linking to Margin Notes

The **Set Link** command creates a hypertext link between text or graphic and the margin note. If you're linking the margin note to text, that text appears underlined and blue in Help. A graphic will look no different as hypertext, so you may wish to add such help-only advice as "Click this picture for more information." When the blue, underlined text or the selected graphic is clicked, the margin note displays in a popup window.

- Click in the right column of the table, that is, in the cell inhabited by the body text.
- Highlight the text that you want to link to the margin note.
- Click the **Margin Note** button  on the Doc-To-Help toolbar.
- Select **Set Link** from the drop-down menu.



- The Margin Note Link box opens. If you have more than one margin note, you can select the note you want by using the **Previous** and **Next** icons.




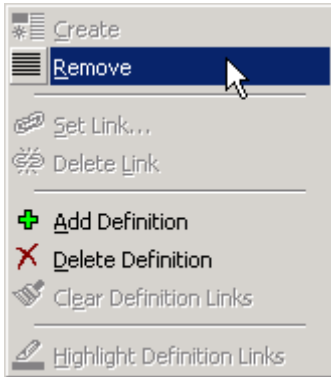
Doc-To-Help places a Word comment next to the text that you have chosen. This comment signals the Doc-To-Help compiler to create a popup window for the margin note.

Alternatively, you can perform the same steps with the **Edit | Margin Note** menu command.

Removing Margin Notes

Place your cursor anywhere in the paragraph adjacent to the margin note.


- Click the **Margin Note** button  on the Doc-To-Help toolbar.
- Select **Remove** from the drop-down menu.



Doc-To-Help removes the table and margin note.

Alternatively, you can perform the same steps with the **Edit | Margin Note** menu command.

Deleting Margin Note Links


- Place your cursor anywhere inside the margin note link you wish to delete.
- Click the **Margin Note** button  on the Doc-To-Help toolbar.
- Select **Delete Link** from the drop-down menu.

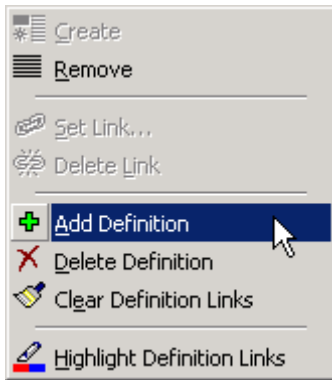


Alternatively, you can perform the same steps with the **Edit | Margin Note** menu command.

Adding Additional Margin Note Definitions

The **Add Definition** command allows you to add multiple margin notes within one paragraph.

- Place your cursor anywhere in the paragraph you want to annotate.
- Click the **Margin Note** button  on the Doc-To-Help toolbar.
- Select **Add Definition** from the drop-down menu.




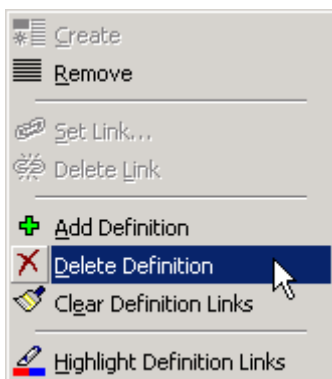
Doc-To-Help splits the margin note column adding an additional cell to the left column of the margin note table.

- Type the margin note in the new cell.

Alternatively, you can perform the same steps with the **Edit | Margin Note** menu command.

Deleting Margin Note Definitions

- Place your cursor anywhere in the margin note cell you wish to delete.
- Click the **Margin Note** button  on the Doc-To-Help toolbar.
- Select **Delete Definition** from the drop-down menu.




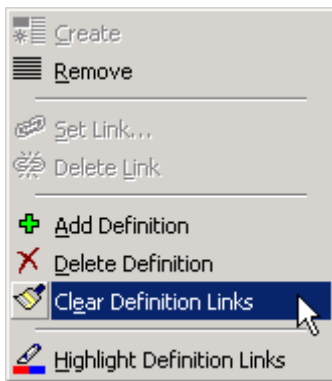
Doc-To-Help removes the margin note cell.

Alternatively, you can perform the same steps with the **Edit | Margin Note** menu command.

Clearing Margin Note Links

The **Create Margin Note** command allows you to remove all links associated with a specific margin note.

- Place your cursor anywhere in the margin note cell you wish to modify.
- Click the **Margin Note** button  on the Doc-To-Help toolbar.
- Select **Clear Definition Links** from the drop-down menu.




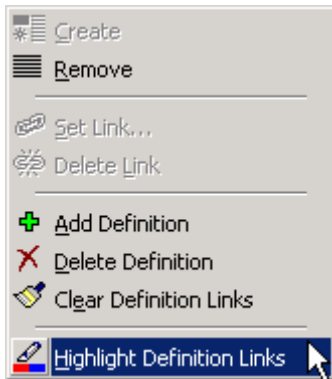
Doc-To-Help removes all links that are associated with that margin note.

Alternatively, you can perform the same steps with the **Edit | Margin Note** menu command.

Viewing Margin Note Links

In some instances you may have multiple margin notes with multiple links for each note. It can become difficult to distinguish which links are associate with which margin notes. Doc-To-Help allows you to easily view the links for an individual margin note.

- Place your cursor anywhere in the margin note you wish to view.
- Click the **Margin Note** button  on the Doc-To-Help toolbar.
- Select **Highlight Definition Links** from the drop-down menu.



Doc-to-Help color codes the margin note cell and the associated links for that margin note.

Alternatively, you can perform the same steps with the **Edit | Margin Note** menu command.

Multiple Paragraphs in Margin Notes

A problem arises with margin notes when you want to have a margin note that contains line breaks. Separating the lines into paragraphs won't work because Doc-To-Help's **Link to Margin Note** command sees just the text up to the first paragraph mark. The easy way to solve this problem is to use line breaks (SHIFT+ENTER) instead of hard returns at the end of each paragraph.

Mid Topic Jumps

The MidTopic property defines a paragraph style or a topic type as an “in topic” jump location similar to an HTML anchor. By defining a paragraph style or topic type as MidTopic and defining a jump method, end users have jump functionality within a lengthy topic.

Inserting a Mid Topic Jump

- Open your source document, select or insert your mid topic heading and define it with the MidTopic paragraph. The mid topic heading must be formatted with a “heading” paragraph on a separate line (not part of the body text).
- Save the document and open or return to the project editor.
- In the left pane, select the **Help Targets** item.
- Choose a Help target from the right pane, and build the Help target.

At this point you have created your mid topic jump location. You can create a hotspot to this location in several ways.

To create a hotspot using the Add Topic Link button:

- Open the source document where you defined the mid topic location heading.
- Select your hotspot text.
- Click the **Add Topic Link** button.

The **Add Topic Link** dialog box opens.

- Click the **Topics** icon in the **View by:** pane.
- Select the topic you defined as your mid topic jump location.
- Click **OK**.
- Close the source document.
- In the left pane of the project editor, select the **Help Targets** item.
- Choose a Help target from the right pane, and build the Help target.

For detailed information on using the Add Topic Links command, see Using the Add Topic Link Dialog Box (page 203.)

To create a hotspot using a character style:

- In your source document and Help target templates, define a “jump” character style. For more information, see Working With Styles in Word (page 102).
- Open the source document where you defined the mid topic location heading.
- Select your hotspot text and define it with your “jump” character style.
- Close the source document.
- Create a matching “jump” character style in the Doc-To-Help project editor. For more information, see Creating New Character Styles (page 150).
- In the right pane of the project editor, select the new character style.
- In the character style property pane, set the Type property to **Jump**.
- In the left pane, select the **Help Targets** item.
- Choose a Help target from the right pane, and build the Help target.

For more information on using the character styles as hotspots, see [Creating a Jump to a Topic in Another Document](#) (page 198.)


Using Dynamic Links

By using Doc-To-Help's Add Dynamic Link feature, you can turn any text into a hyperlink that is associated with one or more index keywords or index groups. Simply highlight the desired text, click the **Add Dynamic Link** button on the Doc-To-Help tool bar and select the desired keywords (KLinks) or keyword groups (ALinks) to create your hyperlink association.

For more information on indexing, see [Building an Index](#) (page 219).

Adding a Dynamic Link

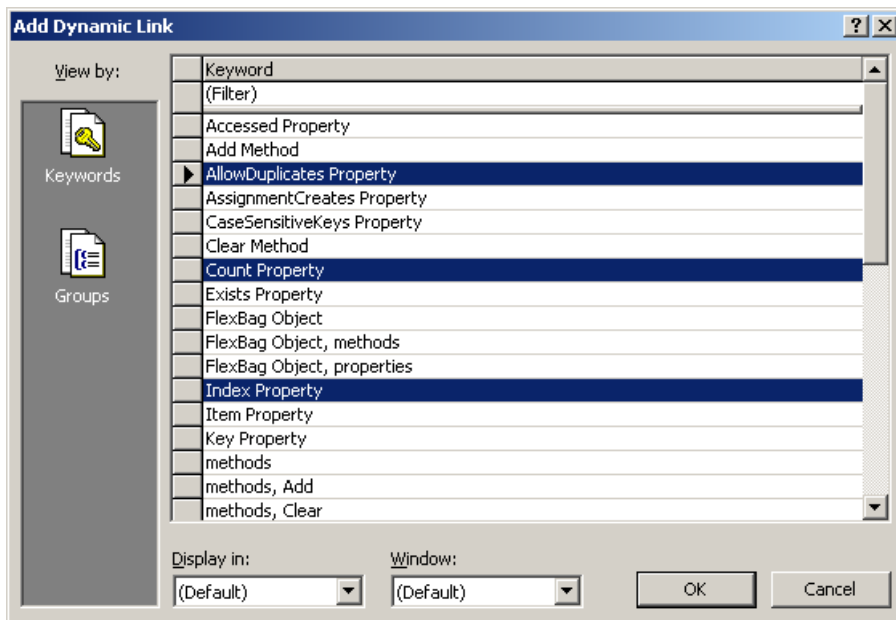
To add a link to an index keyword(s) or to an index group using the **Add Dynamic Link** dialog box:

- In Microsoft Word, highlight the word or words you wish to use as your dynamic link hot spot.
- Click the **Add Dynamic Link** button .
- From the **Add Dynamic Link** dialog box, choose the index keyword or index keyword group you wish to associate with your hot spot and click **OK**.

Doc-To-Help supports multiple index keyword selection by using the *Shift* and *Ctrl* keys. Simply press and hold the **Shift** key, for consecutive selections, or **Ctrl** key, for nonconsecutive selections, and choose the record selector buttons to the left of the index keywords as shown below.

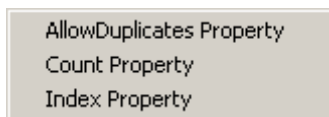
- The **Display in:** combo box allows you to override the TopicsFound property in HTML Help targets on a case-by-case basis. **Note:** The default for this option is a pop-up menu.
- If desired, select which window you want your topics to open in from the **Window:** combo box.

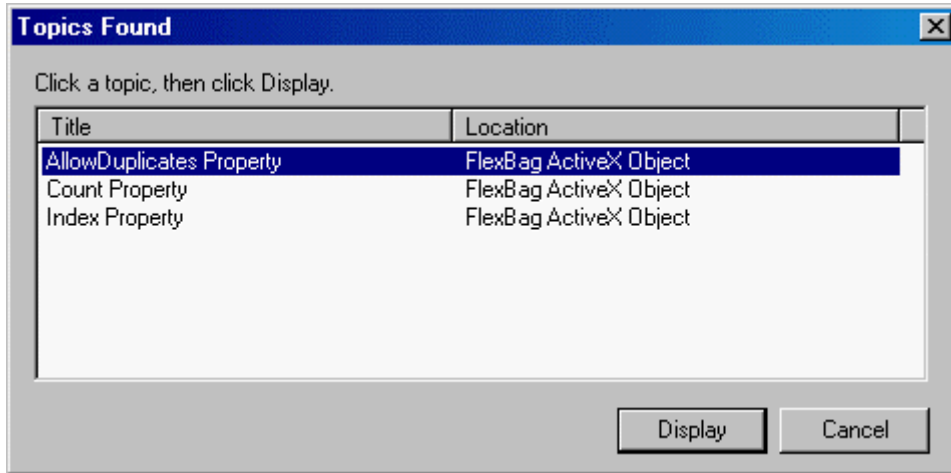
The window dropdown contains the currently defined help windows. You can modify the contents of this dropdown by creating custom help windows. For more information, see [Customizing Help Windows](#) (page 233).



- Save and close the Word document.
- Compile and view your Help project.

Depending on how you set the **Display in:** combo box, clicking the hyperlink presents you with either a pop-up or topics found dialog box from which to choose a topic.





Alternatively, you can perform the same steps with the **Insert | ComponentOne Doc-To-Help | Dynamic Link** menu command.

Building an Index


To construct an index for the reader, such as the Index tab in HTML Help, you define a set of keywords and associate each with one or more Help topics. There are four ways to create index keywords:

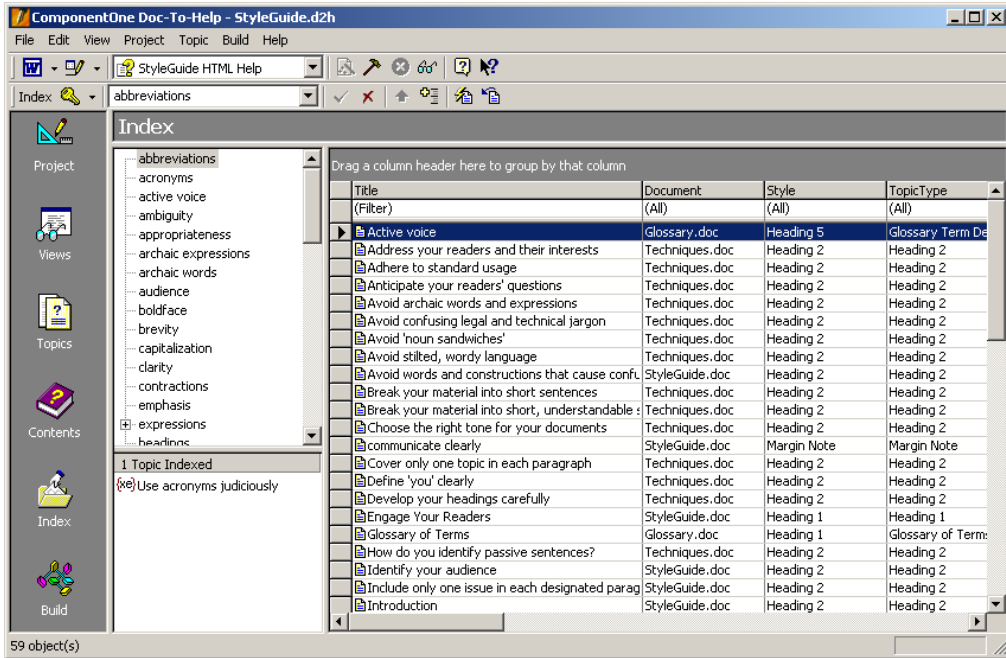
- Natively, by using the inherent capabilities of Microsoft Word (XE fields).
- Manually, by typing them into ComponentOne Doc-To-Help 7.5.
- Automatically, by setting the appropriate property in a paragraph or character style.
- Programmatically, by writing script code.

Likewise, you can use any of these four methods to associate a keyword with a Help topic.

You can also organize Help topics according to a private index that the reader never sees. For example, you may want to mark certain topics as being for advanced users, then provide a hot spot that displays them in a list (like an ALink macro in WinHelp). To construct the private index, you use group names instead of index keywords.

Project Editor Index View

The **Index** view provides an interface for associating index keywords or named groups with specific Help topics. Index keywords are visible to the reader; only the author sees named groups. The upper left pane contains the index keywords that the reader can see while the lower left pane shows which topics are indexed to the keyword selected in the upper left pane. Note that the topics that are indexed by virtue of XE fields are identified with the  symbol in this pane. The right pane displays a sorted list of topics that can be indexed. The topics can be sorted and filtered in the same manner as in the Topic window.



Using the Index Toolbar

On the **View** menu, choose **Toolbars**, then click **Index**. A check mark next to the command indicates that the toolbar is displayed.

After a Doc-To-Help project has been compiled, you can use the index toolbar to view index collections and modify the Help project index.



From left to right, the Index toolbar contains the following commands:

- Index Collections** Allows you to view the groups or keywords collections.
- Index Elements** Indicates/selects which index element is currently being viewed
- Add** Adds an index element
- Delete** Deletes an index element
- Up One Level** Moves an index element up one level
- Show Secondary** Shows secondary or child index elements
- Index Topics** Adds the currently highlighted topic to the current index keyword

Unindex Topics

Removes the currently highlighted topic from the current index keyword

Adding Index Keywords Manually

On the Icon bar, click **Index**.

1. Verify that the **Index Collections** button on the Index toolbar is set to **Keywords**.
2. Right-click any primary keyword in the upper left pane of the Index window and choose **New** from the shortcut menu.
3. The focus is moved to the **Index Elements** textbox. Replace <new> with the keyword you wish to add to the index and press **Enter** or click the adjacent check mark button.

Note that after pressing **Enter**, the keyword is added to the index list. The focus remains in the Index Elements textbox, allowing you to add multiple keywords.

Alternatively, you can add a new keyword by clicking the **Project** menu, choosing **New**, then clicking **Keyword**.

Creating Index Keywords Automatically Using Styles

For any active paragraph style, Doc-To-Help can create index keywords from topic titles and automatically associate them with the appropriate topic. Similarly, for character styles, Doc-To-Help can create index keywords from hot spot text and automatically associate them with the topic that contains the hot spot.

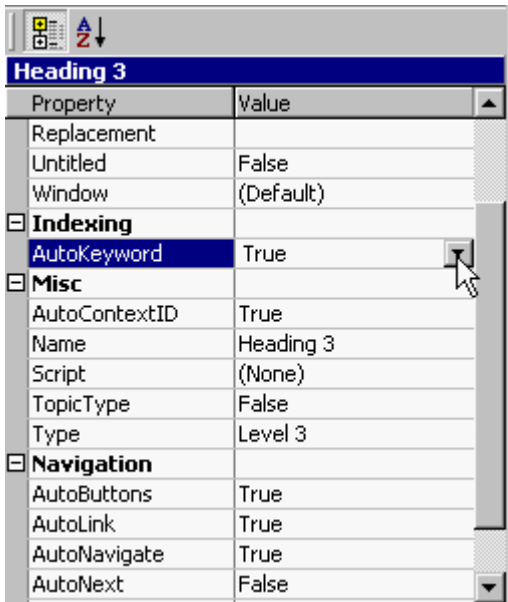
Note: Automatically created keywords will not actually be added to your project until you build the project.

For more information on using styles, see Using Styles in Doc-to-Help (page 143) and Working With Styles in Word (page 102).


To enable automatic indexing for a paragraph style:

- On the Icon bar, click **Project**.
- Select the Paragraph Styles item from the left pane.
- Select the paragraph style to be edited.

- In the paragraph style property pane, toggle the AutoKeyword property to *True*. Index keywords are then created from topic titles formatted with this paragraph style and are associated with the appropriate topic.



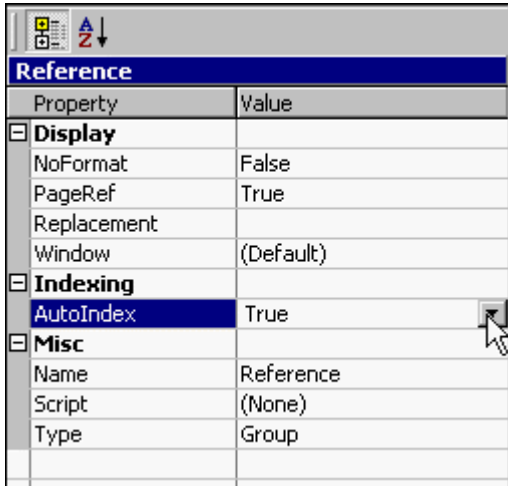
Note: This setting is not available for styles whose **Outline level** is set to *Body text*.

Note that the topics that are indexed in this manner are identified with the  symbol in the lower left pane of the **Topics** icon view.

Once keywords have been automatically added to a project by setting AutoKeyword to True and building the project, subsequently setting AutoKeyword to False will not automatically remove them; you must remove them manually.


To enable automatic indexing for a character style:

- On the Icon bar, click **Project**.
- Select the **Character Styles** item from the left pane.
- Select the character style to be edited.
- In the character style property pane, toggle the AutoIndex property to *True*. Index keywords are then created from text formatted with this character style and are associated with the topic that contains the text.



Reference	
Property	Value
Display	
NoFormat	False
PageRef	True
Replacement	
Window	(Default)
Indexing	
AutoIndex	True
Misc	
Name	Reference
Script	(None)
Type	Group

Note: This setting is not available for styles whose link Type is set to *Group*.

Note that the topics that are indexed in this manner are identified with the  symbol in the lower left pane of the **Topics** icon view.

Adding Secondary Index Keywords Manually

- On the Icon bar, click **Index**.
- Verify that the **Index Collections** button on the Index toolbar is set to **Keywords**.
- Select the primary keyword in the upper left pane or in the **Index Elements** list to which you wish to add a secondary keyword.
- Click the **Show Secondary** button on the index toolbar.

The **Index Elements** textbox is cleared.

- Enter a new secondary keyword, then press **Enter** or click the adjacent check mark button. The new keyword is added to the upper left pane of the Index window.

Alternatively, you can add a new secondary keyword by highlighting the primary keyword, clicking the **Project** menu, choosing **New**, then clicking **Keyword**. Enter the new keyword in the **Index Elements** textbox, and press **Enter**.

Note that after pressing **Enter**, the secondary keyword is added to the index list below the parent primary keyword. The focus remains in the Index Elements textbox, allowing you to add multiple secondary keywords to this primary keyword.

Renaming an Index Keyword

- On the Icon bar, click **Index**.
- Verify that the **Index Collections** button on the Index toolbar is set to **Keywords**, and ensure that keywords (not groups) are displayed in the upper left pane.
- In the upper left pane, right-click the desired keyword, then click **Rename**.
- Type the new keyword name, then press **Enter**. Doc-To-Help moves the keyword to preserve alphabetical order, if necessary.

Alternatively you can select the keyword and then click it a second time so that the cursor blinks at the end of it, type the new keyword name, and press **Enter**.

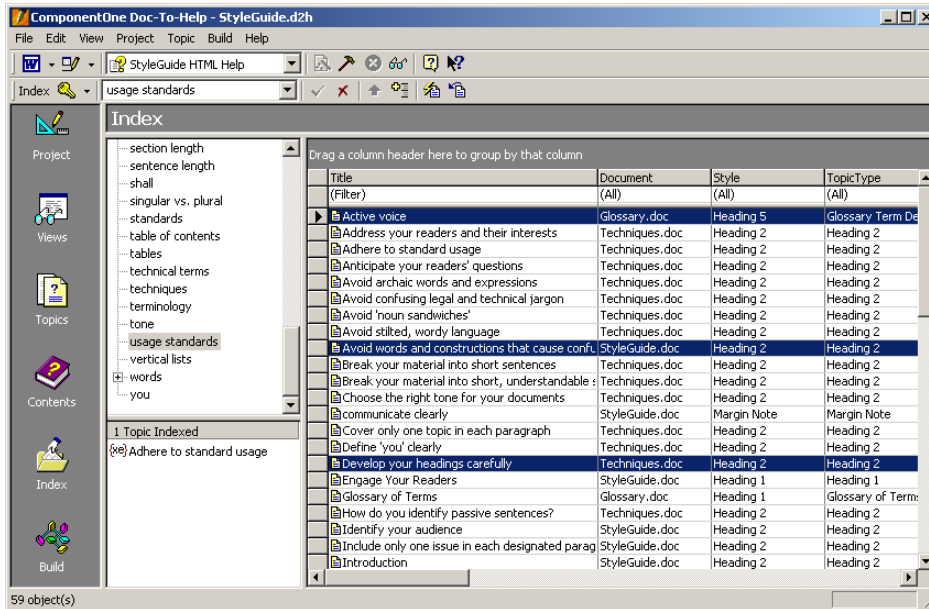
Assigning Topics to an Index Keyword

- On the Icon bar, click **Index**.
- Verify that the **Index Collections** button on the Index toolbar is set to **Keywords**.
- Select the desired keyword in the upper left pane of the Index window.


For large indexes, you can save time by using the auto-completion feature of the **Index Elements** list on the Index toolbar. For more information, see [Using the Index Elements Auto-Completion Feature](#) (page 230).

- Select one or more topics from the right pane that you wish to attach to the index keyword.

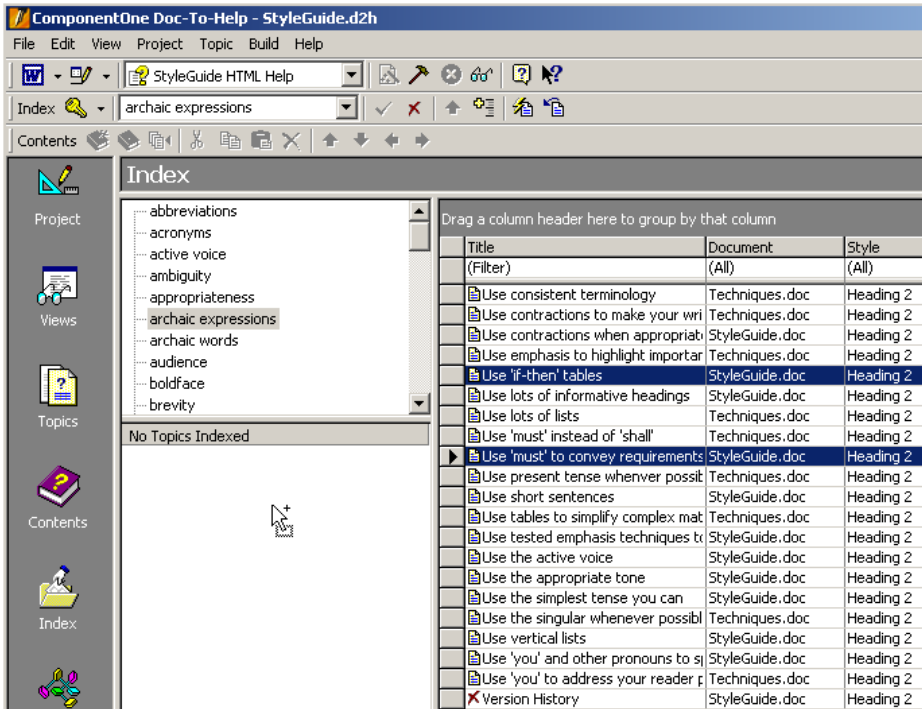
Doc-To-Help supports multiple topic selection by using the *Shift* and *Ctrl* keys. Simply press and hold the **Shift** key, for consecutive selections, or the **Ctrl** key, for nonconsecutive selections, and choose the record selector buttons to the left of the topic titles as shown below.



- Select and right-click any of the highlighted topics and choose **Index** from the shortcut menu or click the **Index Topics** button on the index toolbar.

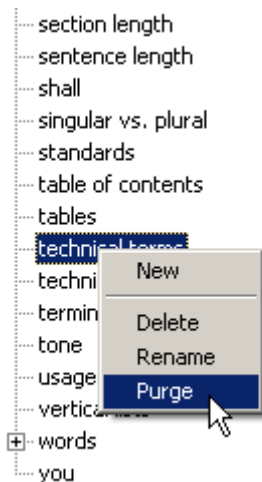
Note that the selected topics are added to the list of indexed topic for those keywords in the bottom left pane. Topics that are indexed in this manner are identified with the  symbol.

Alternatively, you can select the keyword and the topics and “drag and drop” them into the Topics Indexed pane.



Removing Unused Index Keywords

Keeping your project index free of unused index keywords can be time consuming. Even though Doc-to-Help does not use these keywords in your Help targets, it is a good practice to remove unused and out of date keywords. To purge your index of unused keywords, right click anywhere within the keyword pane and select **Purge**.



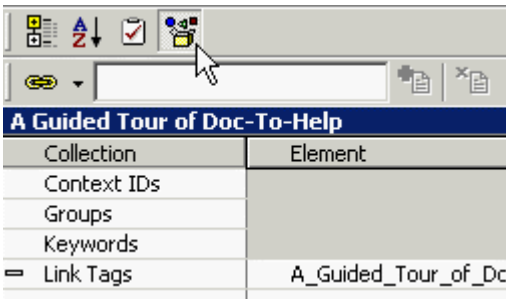
Warning: Index keywords cannot be recovered once they are purged.

Adding an Index Keyword to a Topic

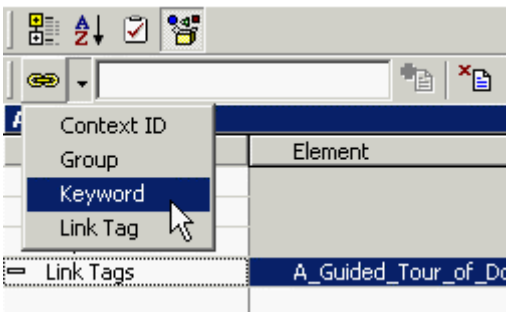
In some cases, choosing a topic and associating an index keyword with it is more efficient than creating index keywords and associating topics to each. This “topic to keyword” workflow makes it easier to create index keywords based on the existing topic titles. After a Doc-To-Help project has been compiled, you can use the Topics Property Toolbar to insert index keywords.

To associate a topic keyword with a specific topic:

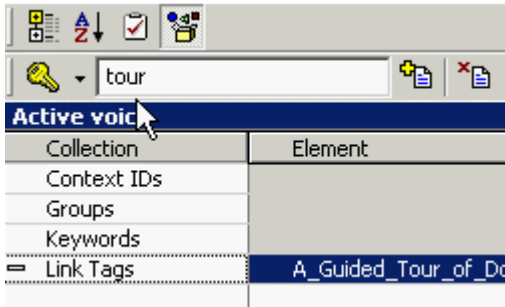
- On the Icon bar, click **Topics**.
- Select the **All Topics** item from the left pane.
- Select the desired topic for editing from the right pane.
- Click the **Collections** button on the Topics Property Toolbar.



- Choose **Keyword** from the **Collection Type** drop-down.



- Type the keyword you want to associate into the **Element Textbox** and press **Enter**.



Note that the keyword has been associated with the highlighted topic and is displayed to the right of the **Keywords** collection.

Keywords created in this manner are automatically added to the main **Index** tab of Doc-To-Help and associated with the current topic. If you enter a keyword that already exists, Doc-To-Help associates the current topic with the existing keyword.

Note: You cannot specify secondary index keywords in the topic property pane.

Creating Topic Groups

To create a named topic group manually:

- On the Icon bar, click **Index**.
- Verify that the **Index Collections** button on the Index toolbar is set to **Groups**.
- Click the **Index Elements** textbox and enter a group name to be added to the index.
- Press **Enter** or click the adjacent check mark button.

Note that after pressing **Enter** the topic group is added to the topic group list. The focus remains in the Index Elements textbox, allowing you to add multiple topic groups.

Alternatively, you can add a new topic group by clicking the **Project** menu, choosing **New**, then clicking **Group**.

To create a named topic group within a script:

See *Creating Topic Groups in a Script* (page 338).

Renaming a Group

- On the Icon bar, click **Index**.

- Verify that the **Index Collections** button on the Index toolbar is set to **Groups**.
- In the upper left pane, right-click the desired group, then choose **Rename** from the shortcut menu.
- Type the new group name, then press **Enter**.

Note that Doc-To-Help moves the group to preserve alphabetical order, if necessary.

Assigning Topics to a Group

- On the Icon bar, click **Index**.
- Verify that the **Index Collections** button on the Icon toolbar is set to **Groups**.
- Select the desired topic group in the upper left pane of the Index window.

For large group lists, you can save time by using the auto-completion feature of the **Index Elements** list on the Index toolbar. For more information, see Using the Index Elements Auto-Completion Feature (page 230).

- Select one or more topics from the right pane that you wish to add to the topic group.

Doc-To-Help supports multiple topic selection by using the **Shift** and **Ctrl** keys. Simply press and hold the **Shift** key, for consecutive selections, or the **Ctrl** key, for nonconsecutive selections, and choose the record selector buttons to the left of the topic titles.

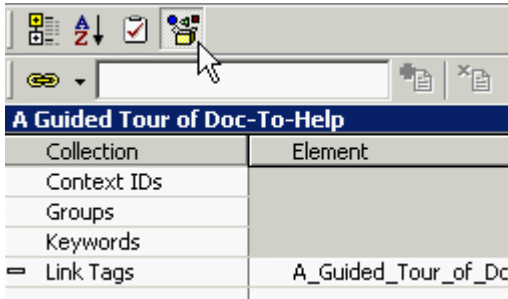
- Select and right-click any of the highlighted topics and choose **Index** from the shortcut menu or click the **Index Topics** button on the index toolbar.

Note that the selected topics are added to the list of indexed topics for that topic group in the lower left pane.

Adding a Group to a Topic

To associate an index group with a specific topic:

- On the Icon bar, click **Topics**.
- Select the **All Topics** item from the left pane.
- Select the desired topic for editing from the right pane.
- Click the **Collections** button on the Topics Property Toolbar.



- Choose **Group** from the **Collection Type** drop-down.
- Type the group name to be associated with the topic into the **Element Textbox** and press **Enter**.

Note that the group has been associated with the highlighted topic and is displayed to the right of the **Groups** collection.

Groups created in this manner are automatically added to the main **Index** tab of Doc-To-Help and associated with the current topic. If you enter a group that already exists, Doc-To-Help associates the current topic with the existing group.

Removing Topics From Index Keywords or Topic Groups

To remove a topic from an index keyword or group:

- On the Icon bar, click **Index**.
- Verify that the **Index Collections** button on the Icon toolbar is set to **Groups** or **Keywords** as required.
- From the upper left pane, choose the group or keyword to be removed.
- In the lower left pane, right-click the topic to be removed and choose **Unindex**.

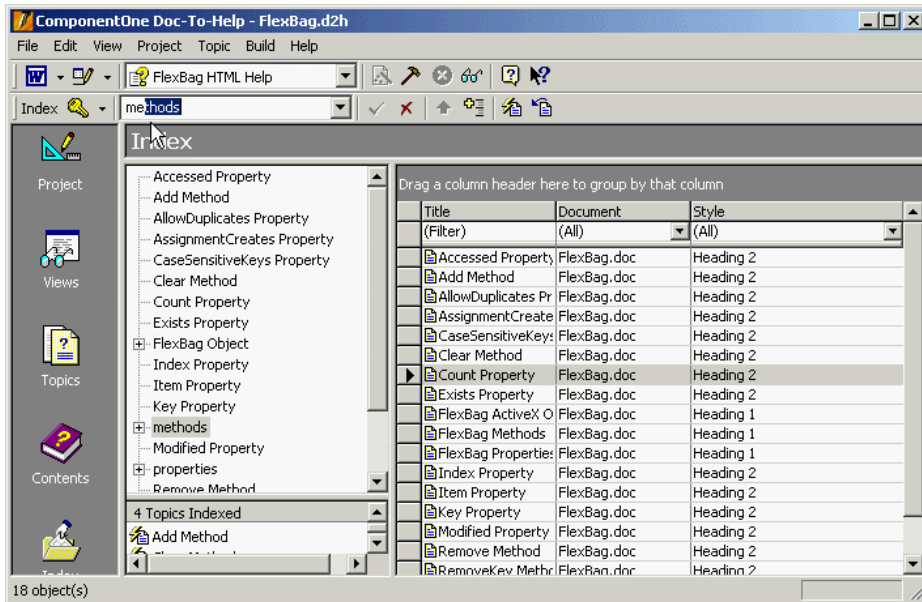
Using the Index Elements Auto-Completion Feature

When you are trying to find a specific keyword or group in a large index, you can save time by using the auto-completion feature of the **Index Elements** list on the Index toolbar.

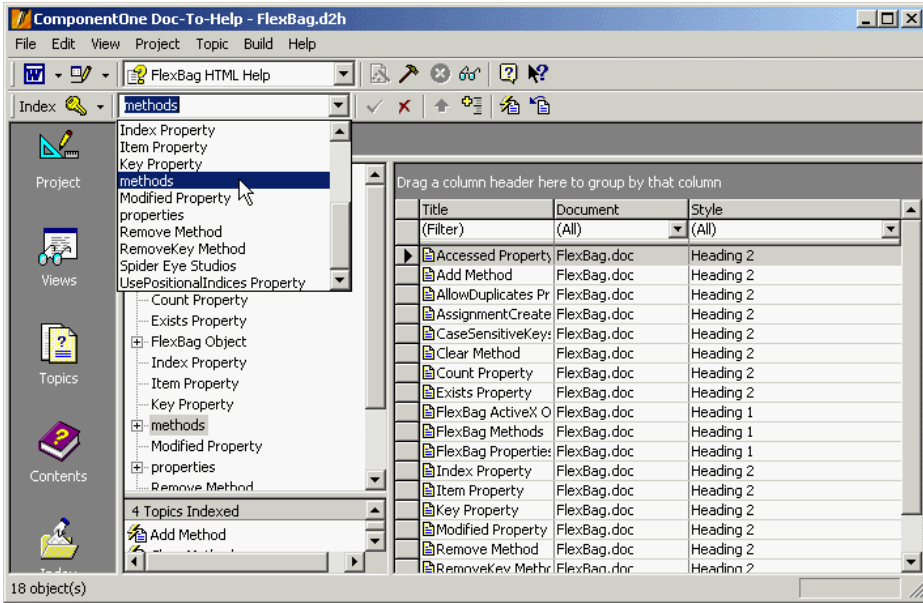
- On the Icon bar, click **Index**.
- Set the **Index Collections** button on the Icon toolbar to either **Keywords** or **Groups**.

- Enter a letter in the **Index Elements** list box, and Doc-to-Help fills in the remaining letters of a word in the list of index keywords or groups.

For example, if you are looking for the keyword **methods**, and you enter the letter **m** in the **Index Elements** list box, Doc-to-Help fills in the remaining letters of a word in the list of index keywords. If you enter another letter after the **m**, such as an **e**, Doc-to-Help finds the first word in the list beginning with **me**.



- Once the remaining letters are filled in, click the arrow and select the topic from the drop-down list of keywords. If you press **Enter** instead of selecting the topic from the list, you will get an error telling you that the keyword already exists.



Customizing Help Windows

The Help targets use the default window, named `main`, to specify the size, location, appearance, and the user interface presented to the reader. You can change the default characteristics by editing the properties of this window in ComponentOne Doc-To-Help 7.5 Pro.

You can also define secondary windows for displaying certain kinds of information. For example, if a topic lists a series of steps for carrying out a procedure, you may want to provide links to other topics with illustrations or additional instructions. By associating a secondary window with the character style that implements the link, you can display multiple topics simultaneously while giving the reader control over the level of detail and the layout of the Help windows.

By selecting the **Windows** item on the project editor tree list, you are presented with a list of the currently defined windows in the right pane. By choosing any one of the windows, the lower left pane displays the property settings for that window.

HTML Help Window Properties

Appearance Properties

<code>TriPaneWindow</code>	When true, enables the standard tripane format for HTML Help.
----------------------------	---

Buttons Properties

<code>Jump1Caption</code>	Contains the caption for the Jump1 button.
<code>Jump1URL</code>	Contains the URL of the HTML file for the Jump1 button.
<code>Jump2Caption</code>	Contains the caption for the Jump2 button.
<code>Jump2URL</code>	Contains the URL of the HTML file for the Jump2 button.
<code>ShowBackButton</code>	When selected, the Back button is included in the Help window.
<code>ShowForwardButton</code>	When selected, the Forward button is included in the Help window.
<code>ShowHideShowButton</code>	When selected, the Hide and Show buttons are included in the Help window.

ShowHomeButton	When selected, the Home button is included in the Help window.
ShowJump1Button	When selected, the Jump1 button is included in the Help window. Chose the caption and URL for this button with the Jump1Caption and Jump1URL properties.
ShowJump2Button	When selected, the Jump2 button is included in the Help window. Chose the caption and URL for this button with the Jump2Caption and Jump2URL properties.
ShowLocateButton	When selected, the Locate button is included in the Help window.
ShowNextButton	When selected, the Next button is included in the Help window. The BinaryTOC property must be set to True for the navigation buttons to function.
ShowOptionsButton	When selected, the Options button is included in the Help window.
ShowPreviousButton	When selected, the Previous button is included in the Help window. The BinaryTOC property must be set to True for the navigation buttons to function.
ShowPrintButton	When selected, the Print button is included in the Help window.
ShowRefreshButton	When selected, the Refresh button is included in the Help window.
ShowStopButton	When selected, the Stop button is included in the Help window.

Miscellaneous Properties

Caption	Represents the string to be displayed in the caption bar of the Help window.
Global	When True, \$global_ is prepended to the window name in generated files and <OBJECT> tags.
Name	Returns a string containing the name of a specified object (read-only).

Navigation Properties

AutoHideNavigationPane	Minimizes the HTML Help navigation pane when HTML Help is not the active window.
AutoSyncNavigationPane	Allows the heading or topic in the Navigation pane to automatically synchronize with whatever topic appears in the Topic pane.
HideNavigationPane	When true, the navigation pane for the Help file is hidden by default.
NavigationPaneWidth	Defines the width of the navigation pane of the Help file in pixels.

Position Properties

AlwaysOnTop	When true, the Help window remains on top of all other windows in the desktop.
Height	Determines the height of the Help window.
Left	Determines the location of the left edge of the Help window. In conjunction with the Top property, determines the position of the Help window on the screen.
Maximized	If true, the Help window is automatically maximized when displayed.
SaveUserPosition	When true, stores the size and position of the Help window in the registry when the user modifies it.
Top	Determines the location of the top edge of the Help window. In conjunction with the Left property, determines the position of the Help window on the screen.
Width	Determines the width of the Help window.

Tab Properties

FavoritesTab	When enabled, the favorites tab is added to the Help window.
SearchTab	Adds a search tab to the Help window.
SearchTabAdvanced	Adds additional functionality to the search tab. Note this property is only available when the SearchTab property is true.

WinHelp 4.0 Window Properties

Appearance Properties

NonScrolling	When the NonScrolling property of a paragraph style or topic type is set to true, any topic headings formatted with that paragraph style or topic type will appear in the non-scrolling region of the Help window.
TopicColor	Displays the current color setting for the topic Help window. To change the color, click the adjacent change button to open a standard color selection dialog box.

Buttons Properties

ShowBackButton	When selected, the Back button is included in the Help window.
ShowBrowseButtons	When selected, the Browse buttons are included in the Help window.
ShowContentsButton	When selected, the Contents button is included in the Help window.
ShowFindButton	When selected, the Find button is included in the Help window.
ShowHelpTopicsButton	When selected, the Help Topics button is included in the Help window.
ShowIndexButton	When selected, the Index button is included in the Help window.
ShowOptionsButton	When selected, the Options button is included in the Help window.
ShowPrintButton	When selected, the Print button is included in the Help window.

Miscellaneous Properties

Caption	Represents the string to be displayed in the caption bar of the Help window.
Name	Returns a string containing the name of a specified object (read-only).

Position Properties

AdjustForScreenSize	Determines whether the Help window conforms to the resolution of the reader's display.
AlwaysOnTop	When true, the Help window remains on top of all other windows in the desktop.
AutoSizeHeight	When true, the height of secondary Help windows are automatically resized to fit the length of the current topic.
Height	Determines the height of the Help window.
Left	Determines the location of the left edge of the Help window. In conjunction with the Top property, determines the position of the Help window on the screen.
Maximized	If true, the Help window is automatically maximized when displayed.
Top	Determines the location of the top edge of the Help window. In conjunction with the Left property, determines the position of the Help window on the screen.
Width	Determines the width of the Help window.

Creating a New Help Window

To create a new window:

- On the Icon bar, click **Project**.
- Select the **Windows** item from the left pane.
- Right-click anywhere in the right pane and choose **New** from the shortcut menu.
Note that the focus of your cursor has moved to the top of the window list pane.
- Type the name of the new window and press **Enter**.
Note that the new window has been added to the list.

Alternatively, you can add a new window by clicking the **Project** menu, choosing **New**, then clicking **Window**.

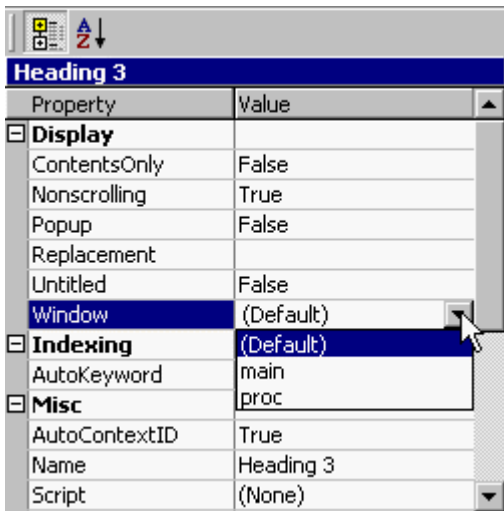
Assigning a Window to a Style

In order for a secondary window to be used, you must assign it to a style as follows:

- On the Icon bar, click **Project**.
- Select the **Paragraph Styles** or **Character Styles** item from the left pane.
- In the right pane, select the name of the style to be edited.

Note that the property pane changes to show the properties of the selected style.

- In the property pane, set the Window property to the window type you would like to be assigned to this style.



You can assign one window to multiple styles. You can also assign the *main* window to one or more styles to prevent the associated topics from being displayed in a secondary window.

Editing Help Window Properties

- On the Icon bar, click **Project**.
- Select the **Windows** item from the left pane.
- Select the name of the window to be edited.

Note that the lower left pane changes to show the properties of the selected window.

- Change the window properties as desired.

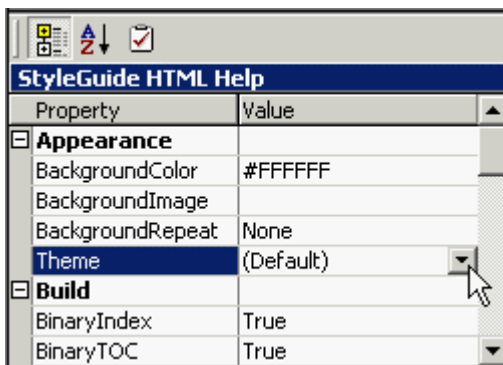
Note that your changes affect the selected window instantly and need not be saved explicitly. However, the full effect of your changes is not realized until you build the Help target.

Using Themes to Modify the Help Window

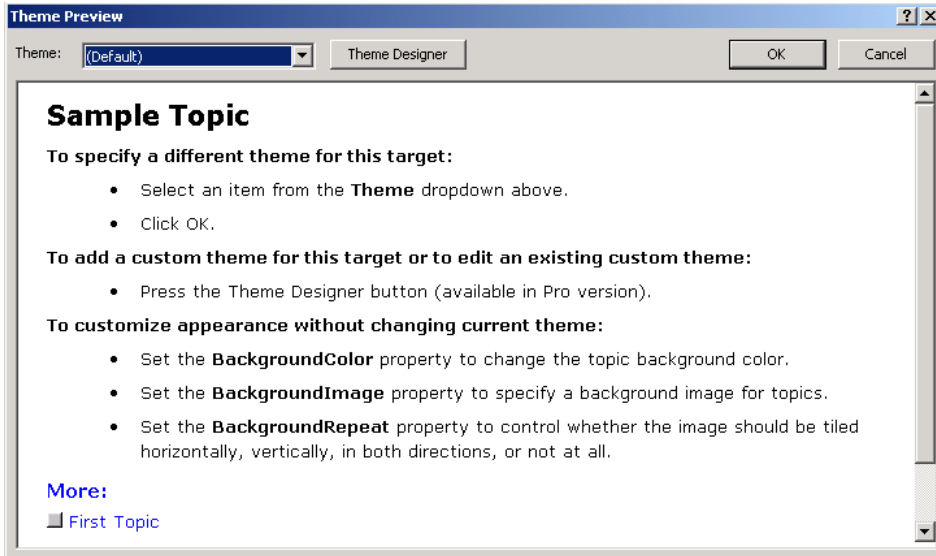
Doc-To-Help allows you to modify the style of your HTML and HTML Help target windows by using the Theme property. From the **Theme Preview** dialog, you can view each of the themes provided with Doc-To-Help and select the one that best fits your purposes. To customize your themes, see Using the Theme Designer (page 417).

To use the Theme property:

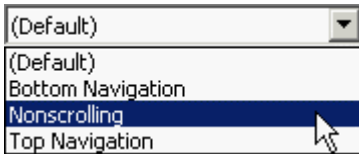
- Select the **Projects** icon.
- Select **Help Targets** from the left pane.
- Select the Help target you wish to modify from the right pane (HTML or HTML Help only).
- From the property pane, select the Theme property and click the button in the box to the right of the property name.



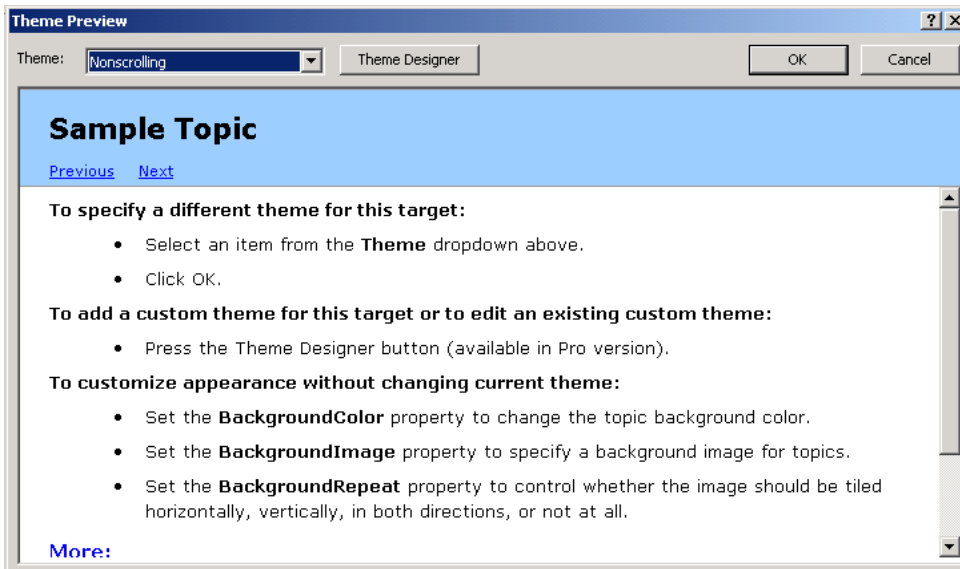
The **Theme Preview** dialog opens.



- From the Theme dropdown, select the theme style you wish to preview.



- After you have viewed the themes and selected the one that suits your needs, click **OK** to accept the theme.



- Build the Help target to reformat your Help window.

Note: To modify your themes, you can access the Theme Designer by clicking the Theme Designer button to the right of the theme dropdown. For more information, see *Using the Theme Designer* (page 417).

HTML Theme Descriptions

Default	The default HTML window includes a contents pane along the left side of the window and navigational hyperlinks above the Help text.
Nonscrolling	The Nonscrolling HTML window includes a contents pane along the left side of the window and a colored non-scrolling area at the top of the Help pane. The non-scrolling area contains the topic heading and navigational hyperlinks.
Tripane	The Tripane HTML window includes a contents pane along the left side of the window and a navigation pane along the top of the window.

HTML Help Theme Descriptions

Default	The default HTML Help window includes a table of contents pane along the left side of the window and the Help content in the right pane.
Bottom Navigation	The bottom navigation HTML Help window includes a table of contents pane along the left side of the window. The right hand Help content pane includes previous/next icons along the bottom of the pane.
Nonscrolling	The Nonscrolling HTML Help window includes a contents pane along the left side of the window and a colored non-scrolling area at the top of the Help pane. The non-scrolling area contains the topic heading and navigational hyperlinks.
Top Navigation	The top navigation HTML Help window includes a table of contents pane along the left side of the window. The right hand Help content pane includes previous/next icons along the top of the pane.

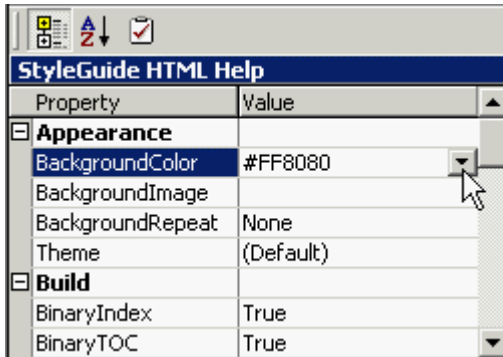
Modifying the Background of the Window Pane

By modifying the `BackgroundColor`, `BackgroundImage` and `BackgroundRepeat` properties, you can customize the background of your Help contents pane for HTML and HTML Help. The `BackgroundColor` property allows you to define the background color of the pane, while using a combination of the `BackgroundImage` and `BackgroundRepeat` properties allows you to select a graphic and position it as part of your background.

Changing the Background Color

To change the background color of your Help contents:

- Select the **Projects** icon.
- Select **Help Targets** from the left pane.
- Select the Help target you wish to modify from the right pane (HTML or HTML Help only).
- From the property pane, select the `BackgroundColor` property and click the button in the box to the right of the property name.



The standard Windows **Color** dialog opens.

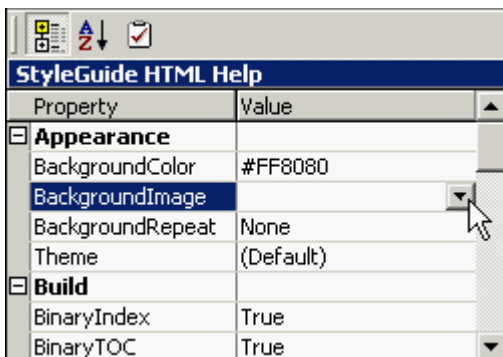
- Select a color from the dialog and click **OK**.
- Build the Help target to reformat your Help contents pane with the new background color.

Note: To preview your color selection before building your Help target, you can open the Theme Preview dialog. For more information, see [Using Themes to Modify the Help Window](#) (page 239).

Inserting and Positioning a Background Image

To insert and position a background image in your Help contents pane:

- Select the **Projects** icon.
- Select **Help Targets** from the left pane.
- Select the Help target you wish to modify from the right pane (HTML or HTML Help only).
- From the property pane, select the BackgroundImage property and click the button in the box to the right of the property name.

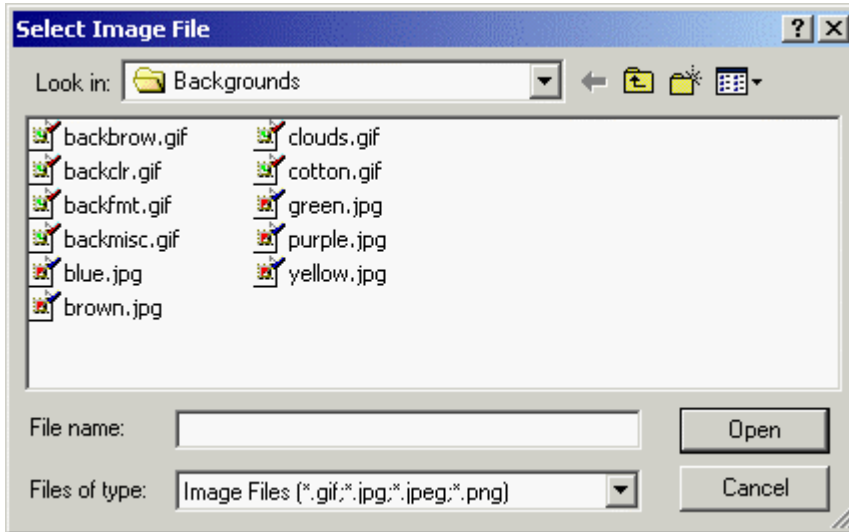


The **Select Image File** dialog opens.

Note, that the Select Image File dialog opens by default to:

C:\Program
Files\ComponentOne\DocToHelp6\Themes\Images\Backgrounds.

This directory contains the default background images included with Doc-To-Help. Any customized background images you plan to use should also be placed in this directory prior to inserting them into your Help project. **Doc-To-Help retrieves background images from this directory, images contained in alternate directories will not be used by Doc-To-Help during the build process.**



- Select a background image from the dialog and click **OK**.

By default, the image will be placed in the upper left corner of the Help contents pane. Your options for image positioning are:

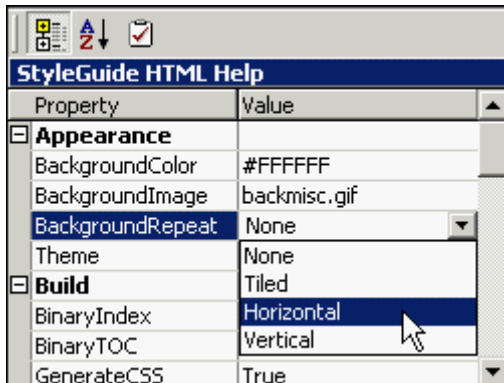
None The background image is placed in the upper left corner of the Help contents pane.

Tiled The background image is tiled, covering the entire area of the Help contents pane.

Horizontal The background image is repeated horizontally along the top edge of the Help contents pane.

Vertical The background image is repeated vertically along the left edge of the Help contents pane.

- Select your background image preference from the dropdown to the right of the BackgroundRepeat property in the Help target property pane.



- Build the Help target to reformat your Help contents pane with the new background image.

Note: To preview your background image selection before building your Help target, you can open the Theme Preview dialog. For more information, see *Using Themes to Modify the Help Window* (page 239.)

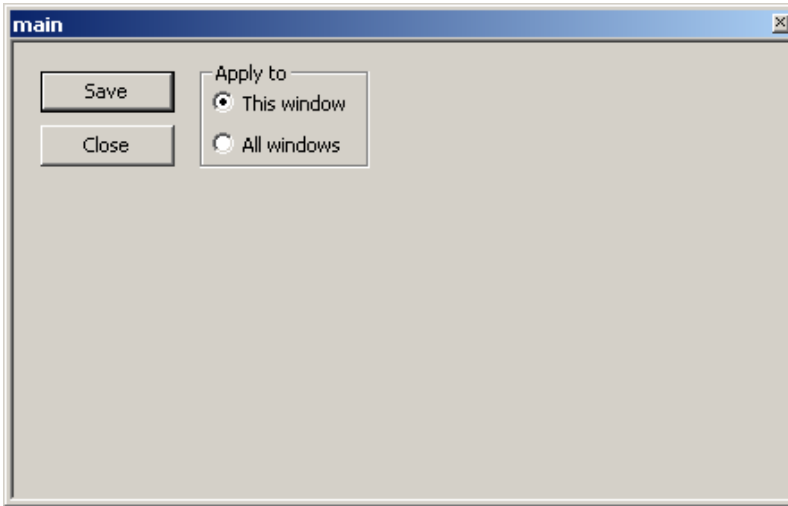
Modifying the Window Size Using the Size Tool

Doc-To-Help provides you with an easy-to-use utility that allows you to modify the size and position of your Help windows.

- On the Icon bar, click **Project**.
- Select the **Windows** item from the left pane.
- In the right pane, select and then right-click the window name to be edited and choose **Size Tool** from the shortcut menu.

The size tool dialog box opens.

- Drag any side or corner of the window to adjust it to the size you want.



- Drag the window to the desired position on the screen and click **OK**.

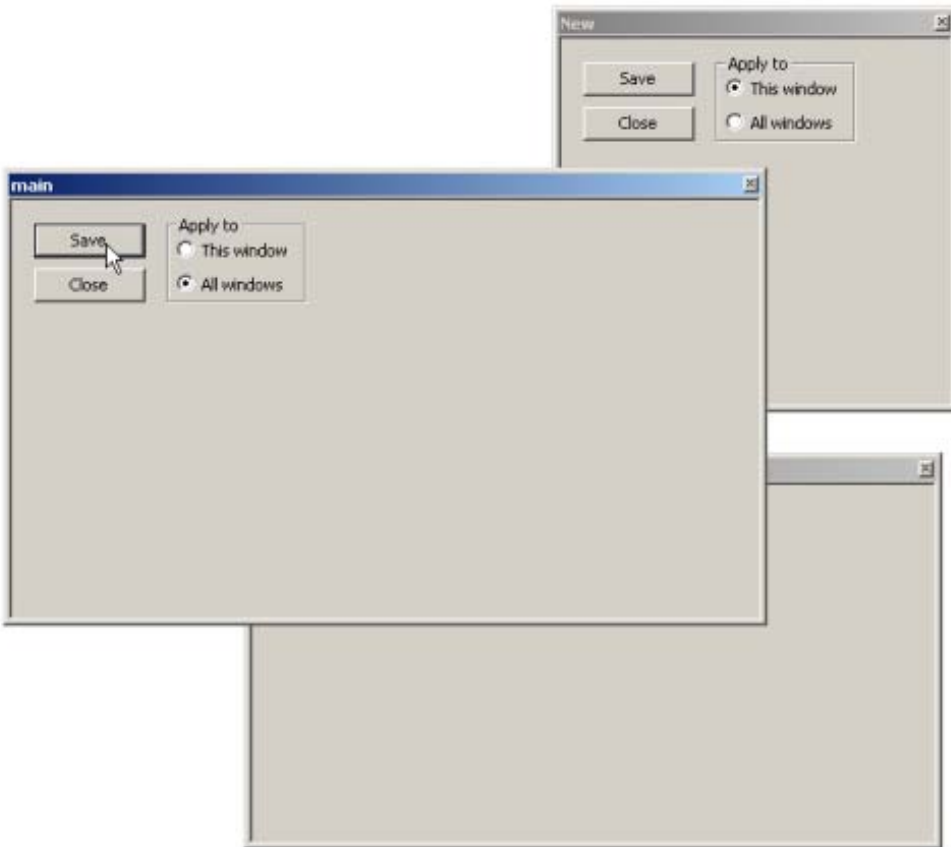
Note: A practical method of using the tool is to open your compiled Help file window and reposition and adjust the size of that window as needed. Then open the **Size Tool** and stretch the size tool dialog box to cover your Help file window and click **OK**. In this way, you have set the new window properties as the default for that window.

Modifying Multiple Windows

Doc-To-Help allows you to modify all windows in your Help system at one time. By opening one Window Size Tool for each defined window you wish to modify, you can reposition them until you are satisfied with the relative locations, then save them all simultaneously.

- On the Icon bar, click **Project**.
- Select the **Windows** item from the left pane.
- In the right pane, select and then right-click the window name to be edited and choose **Size Tool** from the shortcut menu.
- Drag any side or corner of the window to adjust it to the size you want.
- Repeat this process for each window you wish to resize leaving all windows open.
- Click **All Windows** in any of the size tool windows and click **Save**.

All open size tools close and the size and location settings for all windows is saved.



Tips and Techniques

This chapter shows how you can enhance your Help projects with glossary entries, graphics, macros and Internet links. It also describes the printing, reporting, and project organizer features available through ComponentOne Doc-To-Help 7.5.

Working with Graphics

When preparing your source documents for Doc-To-Help, you can include static graphics just as you would in a "normal" document. You do not need to use any nonstandard commands or specify target-specific escape sequences.

However, some of the graphics features introduced in Word 97 are not suitable for Help authoring, and you should avoid using them. Here are some guidelines to follow when inserting pictures into source documents:

- Don't use floating pictures. Since online Help systems don't have the concept of Z-order, the placement of the picture may not be what you expect.
- Instead of pasting a picture into a document directly from the Clipboard, save the picture to a file and insert it as a linked object. That is, use the **Link to file** option in the **Insert Picture** dialog box. This will save you time if the pictures are likely to change.
- If a picture is too large, you can crop it or scale it in Word, and Doc-To-Help applies the appropriate transformations to the image file it generates during compilation. However, if you crop or scale a linked picture, use the **Insert and Link** option in the **Insert Picture** dialog box. Otherwise, Doc-To-Help reads the picture file from disk as is.

Inserting Graphics Files

- On the **Insert** menu in Microsoft Word, point to **Picture**, then click **From File**. The **Insert Picture** dialog box opens.
- Specify a folder and file name, then choose **Insert**, **Link to file** or **Insert and Link**.

Scaling Inserted Graphics

- Select the graphic you inserted, right-click, and select **Format Picture** from the shortcut menu.

Note: To scale the graphic using the features contained within Microsoft Word, you must have “inserted”, not “linked” the graphic.

- Select the **Size** tab.
- In the Scale area, change **Height** and **Width** as desired.
- Click **OK**.
- Save and close the Word document.
- In the Project Editor, click the **Project** icon.
- Select **Help Targets** from the left pane.
- Select the correct Help Target from the right pane. In the property pane, verify that the ScaleGraphics property is set to True.

Note: The ScaleGraphics property allows you to determine on a target-by-target basis if the Word scaling will be applied.

- Click the **Make Target** button to compile the Help target.

Inserting Objects to Create Special Effects

Some of Microsoft Word's drawing and text layout features are ideal for producing hard copy output, but ill-suited for creating online Help. AutoShapes, text boxes, and WordArt effects can only reside in the drawing layer; you cannot insert them as inline elements. Since you have no control over the placement of floating objects in the RTF files passed to the WinHelp compiler, you really can't take advantage of these features.

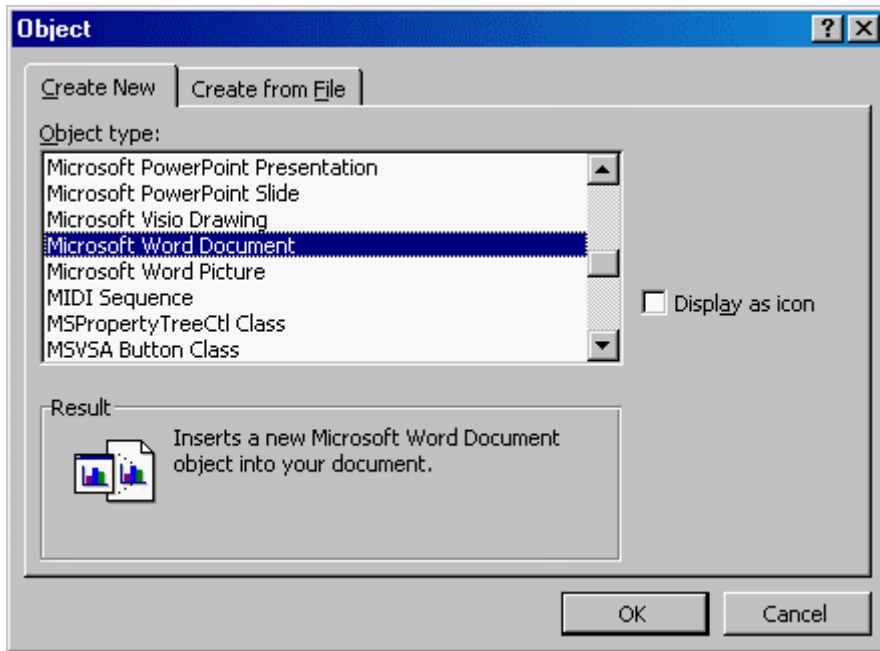
Although the WinHelp compiler supports Word tables, it does not support table borders or shaded cell backgrounds. If you want to see these table features in a Help file, you have to create a bitmap that looks like a table or take a screen shot of a table in Word.

Fortunately, you can circumvent these restrictions on tables by embedding another Word document within your source document as follows:

In your source document, click where you want to insert a table or drawing object.

- On the **Insert** menu, click **Object**.

The following dialog box opens.



- In the **Object type** list, select *Microsoft Word Document*.
- If checked, clear the **Display as icon** check box, then click **OK** to create a new document.
- Insert the table or drawing object as you would in a "normal" Word document.
- When you are finished editing, click **Close and Return** on the **File** menu to return to the original document.

When Doc-To-Help compiles the source document, it treats the embedded Word document as a graphic and creates an appropriate image file for the current Help target.

Example

The following table was inserted into the source document as an embedded Word document.

Wine Tasting Chart	
intensity	delicate powerful
dry or sweet	bone drydessert
body	light body very full body
acidity	soft, gentle very crisp
tannin	none heavy tannins
oak	none heavy oak
complexity	direct very complex

Like tables, Doc-To-Help cannot translate floating pictures into the final Help system. Converting a floating picture to an Inline shape will ensure that this image is included in the final document.

Converting Floating Pictures to Inline Shapes

- Select the picture to be converted.
 Note that floating pictures are only visible in online layout or page layout view.
- Right-click the picture, then choose **Format Picture** from the shortcut menu.
- On the **Layout** tab, select the **In line with text** wrapping style, then click **OK**.

Creating Hot Spots With Graphics

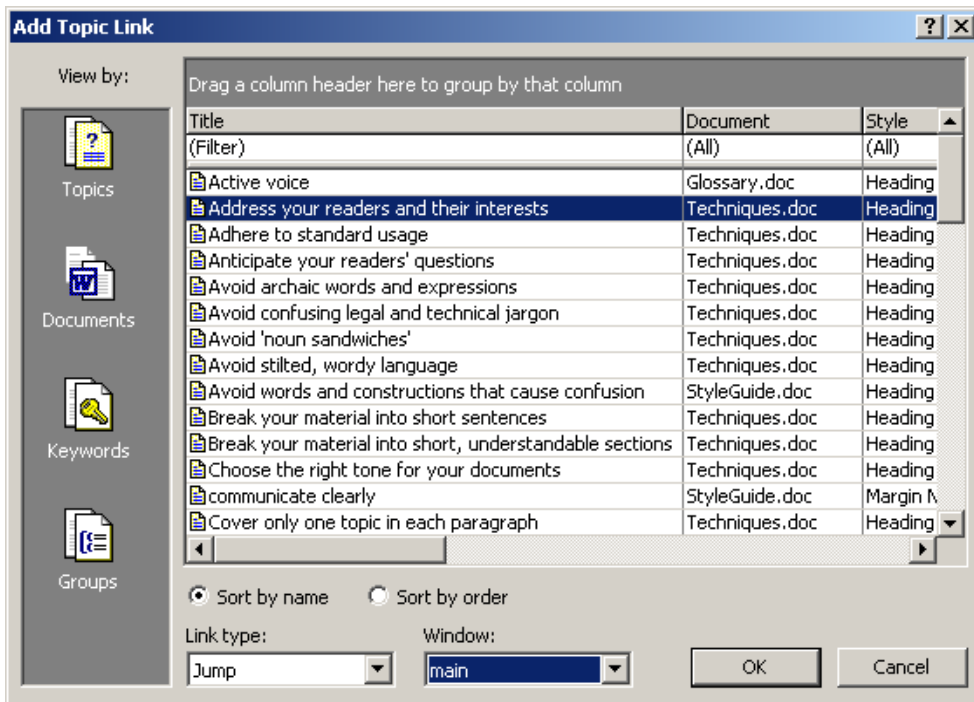
Typically, you implement links and hot spots as text formatted with an active **character style**. Doc-To-Help derives a link tag from the text during compilation, this link tag links this text to the destination hot spot. Depending upon the style's link type, Doc-To-Help looks for a matching topic, index keyword, or group name. You can also override or cancel the default link tag by writing a script and associating it with the character style. Please review the section Links and Hot Spots (page 197) if you are not already familiar with the text to link tag process.

You can also implement links and hot spots using graphics instead of text. As with textual links, Doc-To-Help uses active character styles to determine both the link type and the destination, and you can override or cancel the default behavior with scripts. Doc-To-Help still needs some text from which it can determine the destination and create the link tag, however. Since Microsoft Word does not provide an intrinsic way to attach an arbitrary string to a picture, Doc-To-Help provides a command that does just that.

To create a graphical hot spot:


- Insert the picture you want to use as the hot spot.
 For specific instructions, see Inserting Graphics Files (page 249.)

- Select the hot spot graphic then click the **Add Topic Link** dialog box button.
- In the **Add Topic Link** dialog box, choose the topic you wish to associate with your hot spot.
- From the **Link Type:** drop-down, select the **Jump** or **Popup** option.
- From the **Window:** drop-down, select the window type where you wish to display your topic and click **OK**.



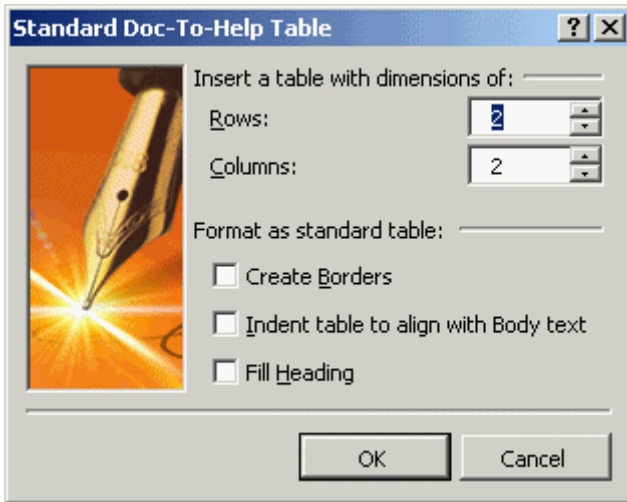
For more information on customizing windows, see *Customizing Help Windows* (page 233).

Inserting Standard Tables with Doc-To-Help

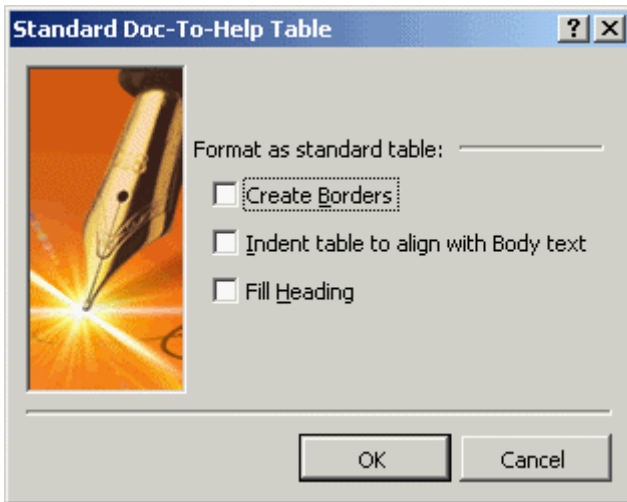
You can use the **Standard Table** command to insert basic tables quickly and easily into your source documents. These tables can be aligned with your body text and formatted by selecting options contained in the Standard Doc-To-Help Table dialog. To access the Standard Table dialog, click the Standard Table button . The command behaves differently depending on whether you have selected text.

If no text is selected, you are presented with a dialog that allows you to specify the number of rows and columns in your table. The table also allows you to determine if

your table has borders, whether it is aligned with your body text and whether the heading has a fill.



If text is selected, Doc-To-Help uses the convert text to table features contained within Word to format the table. If the text contains tabs, each tabbed section is placed into an individual table cell. If the selected text consists of more than one paragraph, each paragraph is placed into a new row in the table. When you click the **Standard Table** button with text selected you are presented with the following dialog.



As you can see, the number of rows and columns is automatically selected for you and you are left with the standard “formatting” options.

By selecting the **Create Borders** checkbox, your table is formatted with a double line border as show below.

Table with **Create Borders** selected.

By selecting the **Indent table to align with Body text** checkbox, your table is aligned with the body text in your document as shown below.

Table with Indent table to align with Body text selected.

By selecting the **Fill Heading** checkbox, your table is formatted with the uppermost set of table cells containing a gray fill as show below.

Table with Fill Heading selected.

Alternatively, you can perform the same steps with the **Table | Standard Doc-To-Help Table** menu command.

Linking to Internet Sites

You can use Microsoft Word's hyperlink feature to enhance your online Help targets with links to Web pages and downloadable files. Word supports Internet links via HYPERLINK field codes, which Doc-To-Help recognizes during source document compilation.

When Doc-To-Help builds an online Help system, it automatically converts HYPERLINK field codes into instructions that are appropriate for the active Help target. For WinHelp, it generates `ExecFile` macros; for HTML and HTML Help, it generates `<A HREF>` tags.

When compiling WinHelp, Doc-To-Help renders hyperlinks as ordinary topic jumps. On most systems, WinHelp topic jumps appear as single-underlined green text. Clicking a hyperlink always opens the default Internet browser in a separate application window; you cannot display a Web page within the confines of the main WinHelp window or a secondary window.

In HTML and HTML Help, hyperlinks are rendered according to your reader's browser settings. Unlike WinHelp, you can control whether hyperlinked URL's display within the same window or a separate application window. By default, they are displayed within the same window (or frame).

Entering Hyperlink Text

The simplest way to insert a hyperlink is to type a URL and let Word convert it to a HYPERLINK field for you. The following examples are all valid hyperlinks:

<http://www.componentone.com/>

www.componentone.com

<ftp://ftp.componentone.com/>

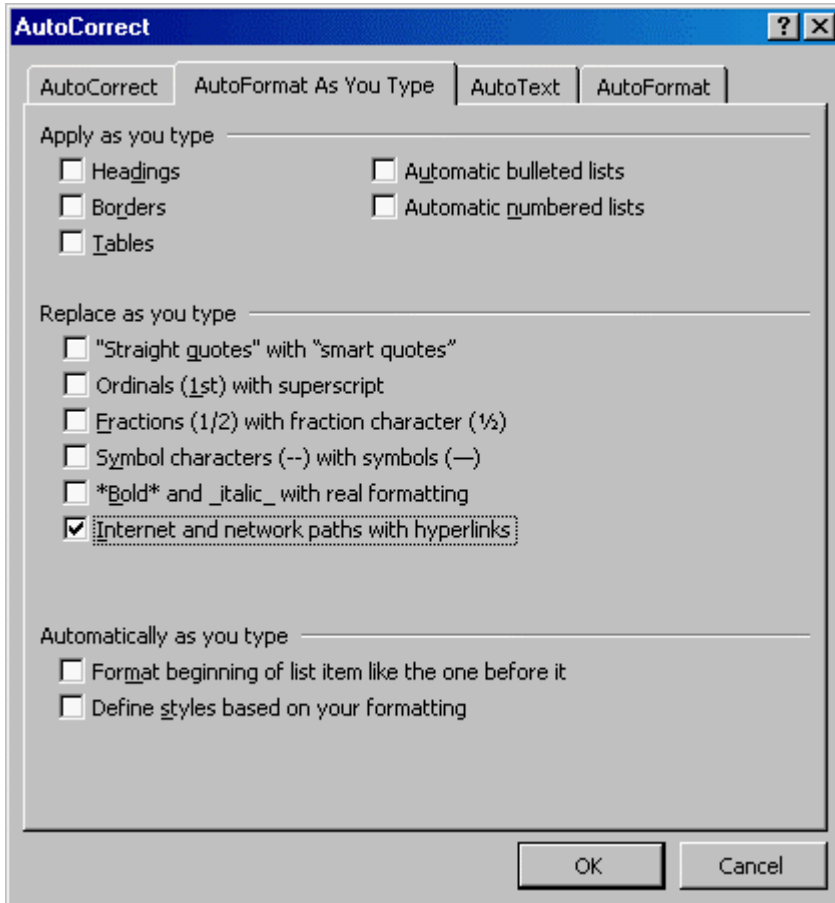
To enter hyperlink text:

- Type the URL as you would normal text. You can generally omit the `http://` and `ftp://` protocol prefixes.
- Terminate the URL with a space, tab, punctuation mark, or the ENTER key. Word automatically applies the *Hyperlink* style to the text you typed and converts it into a field code having the following format:

```
{ HYPERLINK your-URL }
```

If Word does not convert the text as described here, the option has been turned off. You can turn it back on as follows:

- On the **Tools** menu in Microsoft Word, click **AutoCorrect**.
The **AutoCorrect** dialog box opens.
- On the **AutoFormat As You Type** tab, check the box labeled **Internet and network paths with hyperlinks**, then click **OK**.



Inserting Hyperlink Fields

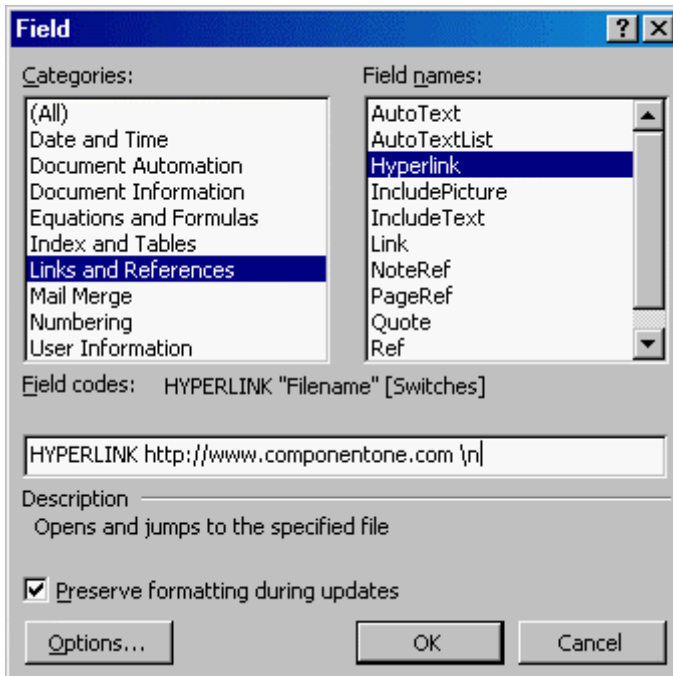
To create a hyperlink that displays in the current window:

- On the **Insert** menu in Microsoft Word, click **Hyperlink**.
- The **Insert Hyperlink** dialog box opens.
- Enter the text to display for your hyperlink in the **Text to display** box.
- Enter or locate the URL in the **Type the file or Web page name** box.
- You can enter the address of a Web page (HTTP protocol) or a downloadable file (FTP protocol). You can locate a specific file or website by clicking the File or Web Page buttons on the right. In addition you can add any valid protocol that Word recognizes (mailto, nntp, help, etc).
- When you are finished, click **OK**.

To create a hyperlink that opens a new application window:

- On the **Insert** menu in Microsoft Word, click **Field**.
The **Field** dialog box opens.
- In the **Categories** list, select **Links and References**.
- In the **Field names** list, select **Hyperlink**.
- In the **Field codes** box, enter the URL and the \n switch as shown in the following figure, then click **OK**.

Make sure that the URL is separated from the field code and the switch with a space and that you include the HTTP protocol identifier at the beginning of the link.



Note: The \n switch has no effect in WinHelp builds.

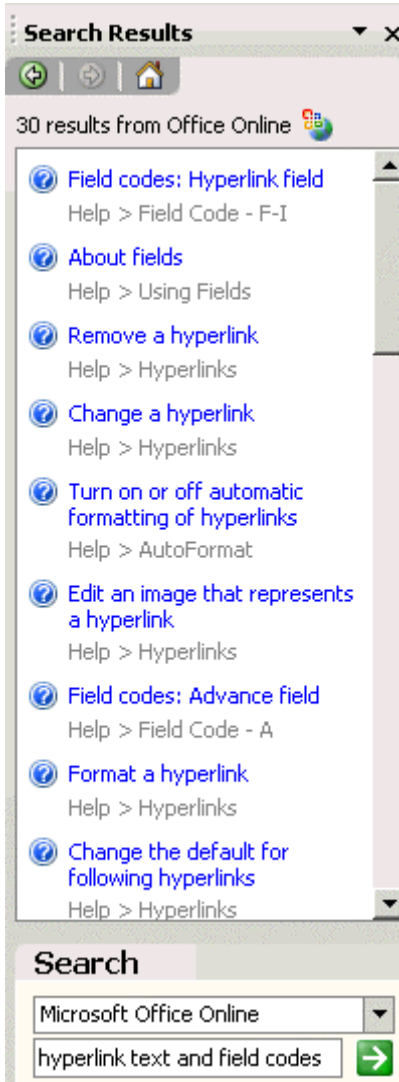
Editing Hyperlink Fields

To change the URL or optional argument for a hyperlink field:

- Right-click the hyperlink text, then point to **Hyperlink**, then click **Edit Hyperlink on the submenu**.
- The **Edit Hyperlink** dialog box opens.
- Specify new display text in the **Text to display** box, if desired.
- Specify a new URL in the **Type the file or Web page name** box, if desired.
- When you are finished, click **OK**.

Alternatively, you can edit the field code directly within the document:

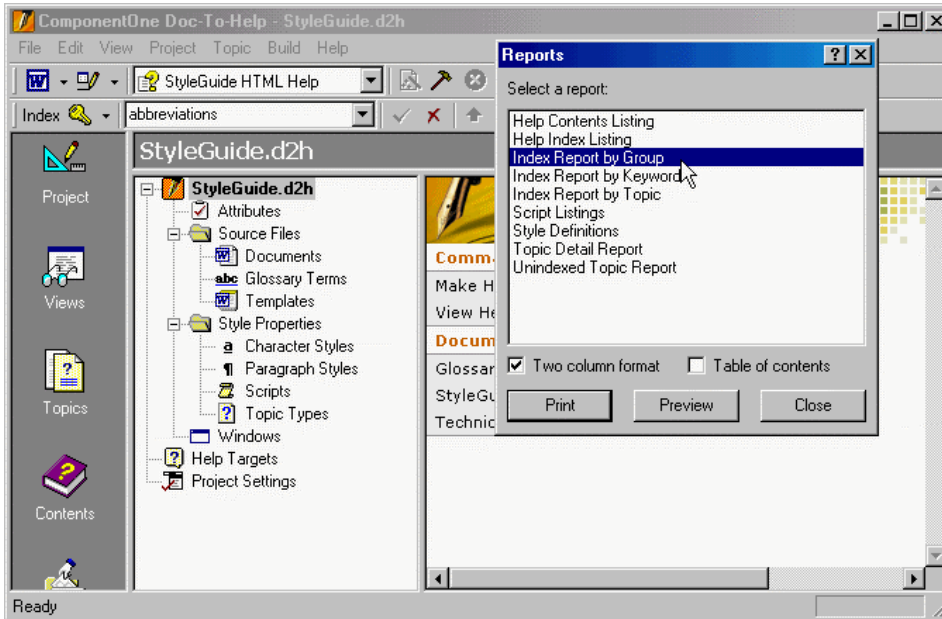
- Right-click the hyperlink text, then click **Toggle Field Codes**.
Word replaces the hyperlink text with a field code string enclosed in curly braces.
- Edit the field code as desired. Include the \n switch to indicate that the hyperlink should open in a separate window. See Word's Help for more information on hyperlink text and field codes.



- When you are finished editing, right-click the hyperlink text, then click **Toggle Field Codes** to restore the original hyperlink text.

Generating Project Reports

Doc-To-Help offers a variety of project reports and printable views to help you organize your work and keep track of topics and index elements. You can either produce hard copy or preview the output on screen.



One way to get information on your current project is to print out the current view in the Doc-To-Help visual environment. This type of report prints the items in the panes of the current view.

Printing the Current View

- On the **File** menu, click **Print**. Doc-To-Help opens a standard print dialog box.
- Change the print job settings, if desired:

The output of the **Print** command reflects the either the right pane or the properties pane, depending on which has focus.

To change the paper size, source, or orientation, click **Print Setup** on the **File** menu.

Previewing Pages Before Printing

- On the **File** menu, click **Print Preview**. Doc-To-Help replaces its main window with a preview window that displays individual pages as they will be printed.
- Use the row of named buttons at the top of the preview window to navigate, zoom, or print:

Print	Open the print dialog box to start a print job.
Next Page	Display the next printed page.
Prev Page	Display the previous printed page.

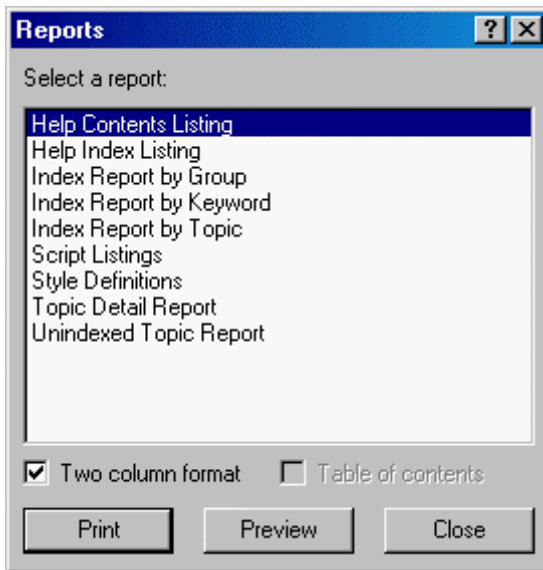
One Page/Two Page	Preview one or two printed pages at a time.
Zoom In	Enlarge the preview image.
Zoom Out	Shrink the preview image.
Close	Return the main window to its normal state.

Whenever you open the preview window, Doc-To-Help rescales the preview image so that an entire page fits within it. If the resulting text is too small, you can resize or maximize the main window or click the **Zoom In** button.

Selecting a Report for Printing

- On the **View** menu, click Reports.
- Select the report you want to see.

For a description of each, see Available Reports (page 263).



Depending upon which report is selected, it may be available in **Two column format** with an optional **Table of contents** at the end of the print job. If available, select or clear the corresponding check boxes as desired.

- Click **Print** to send the report to the default printer, or click **Preview** to view the report within the Doc-To-Help application window. Click **Close** to dismiss the dialog box without printing or previewing.

Available Reports

This version of Doc-To-Help includes the following reports:

Help Contents Listing

This report depicts an online Help contents with all books expanded and all topics visible. It supports the two-column option.

Help Index Listing

This report depicts an online Help index with all secondary keywords visible. It supports the two-column option.

Index Report by Group

For each group name in the project, this report shows a list of associated topics, if any. Groups are listed in alphabetical order. This report supports both the two columns and table of contents options.

Index Report by Keyword

For each index keyword in the project, this report shows a list of associated topics, if any. Keywords are listed by full pathname in alphabetical order. This report supports both the two columns and table of contents options.

Index Report by Topic

For each topic in the project, this report shows a list of associated keywords and groups, if any. Topics are listed in alphabetical order by title. This report supports both the two columns and table of contents options.

Script Listings

This report lists the code for each script defined in the project. Scripts are listed in alphabetical order. This report supports the table of contents option.

Style Definitions

This report lists the properties of each style defined in the project. Styles are listed in alphabetical order. This report supports the table of contents option.

Topic Detail Report

For each topic in the project, this report lists its properties. Topics are listed in alphabetical order by title, and each topic starts on a new page. This report supports the table of contents option.

Unindexed Topic Report

This report lists each topic that is not associated with an index element (keyword or group). Topics are listed alphabetically by title, along with the document and style from which the topic was derived.









Managing Project Files

Doc-To-Help provides an Organizer dialog box similar to the one in Microsoft Word, which you can use to transfer styles, scripts, window definitions, and index elements from one project to another. You can also use the Organizer dialog box to delete these items from a project.

Copying Data Between Projects

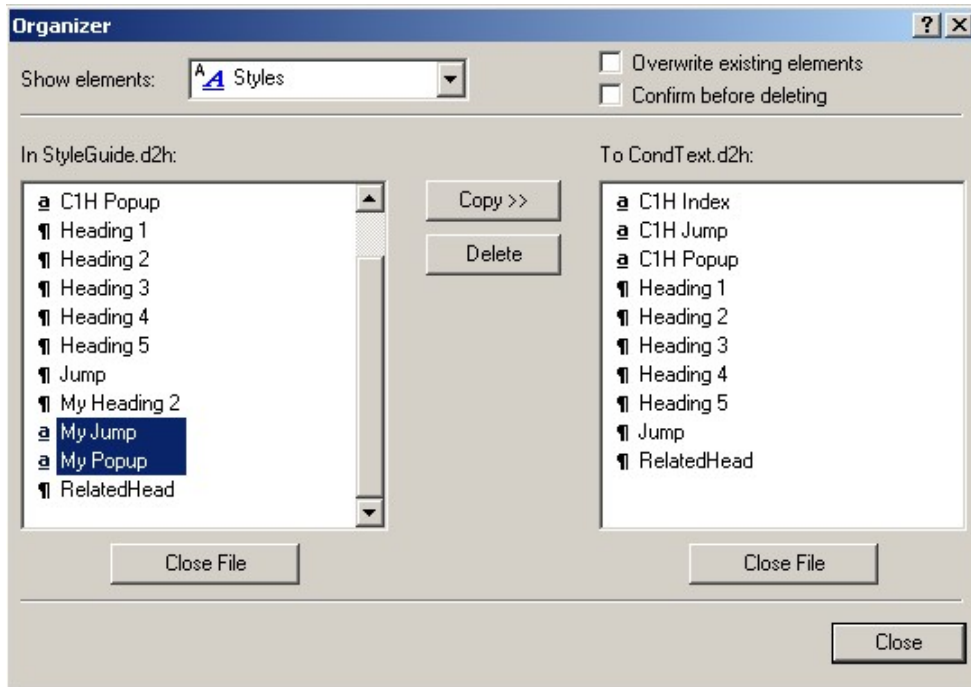
On the **Project** menu, click **Organizer**. The **Organizer** dialog box opens.

Initially, all elements in the active project are displayed in the list on the left side of the dialog box. Use the **Show elements** list to view a specific type:

	Paragraph and character style behaviors
	Group names
	Index keywords
	Script code modules
	HTML Help window definitions
	WinHelp window definitions
	Help target properties
	Topic type definitions

- Click **Open File** to display a file selection dialog box. Specify a folder and file name for an existing Doc-To-Help project, then click **Open**. The list on the right side of the Organizer dialog box now contains elements from the specified project. Initially, none of the elements are selected, and the **Copy** button is unavailable.
- By default, like-named elements are not copied from one project to another. If desired, check the **Overwrite existing elements** box to force the Organizer dialog box to redefine existing elements in the destination project.
- Select one or more elements in the list on the right, then click **Copy**. When you are finished, click the **Close** button at the lower right corner of the dialog box.

The following figure shows the Organizer dialog box ready to copy two styles from one project (StyleGuide.d2h) to a new project file (CondText.d2h).



Deleting Individual Project Elements

To delete an individual style, script, topic type or window outside of the Organizer:

- Click the **Projects** icon.
- In the left pane of the project editor, choose the appropriate item.
- Right-click the element to be deleted, then click **Delete**.

To delete an individual index keyword or group:

- Click the **Index** icon.
- Ensure that the appropriate category is displayed by choosing **Groups** or **Keywords** from the index toolbar.
- Right-click the element to be deleted, then click **Delete**.

Doc-To-Help prompts before deleting a script, but deletes all other element types without confirmation. If you delete a script or window that is assigned to one or more styles, Doc-To-Help also detaches the deleted element from those styles.

You can also use the Organizer dialog box to delete individual project elements:

- On the **Project** menu, click **Organizer**.
The **Organizer** dialog box opens.
- Initially, all elements in the active project are displayed in the list on the left side of the dialog box. Use the **Show elements** list to view a specific type (groups, keywords, scripts, styles, or windows).
- Select the **Confirm before deleting** check box if you want the Organizer dialog box to prompt you before deleting individual elements.
- Select one or more elements in the list on the left, then click **Delete**. When you are finished, click the **Close** button in the lower right corner of the dialog box.

Modifying the Glossary

To assist in developing an online and printed glossary, Doc-To-Help automatically creates an empty glossary document whenever a new project is created. During compilation, Doc-To-Help examines the project documents for text that matches the glossary entries, converting these matches into hyperlinks. Whenever the user clicks on one of the hyperlinks, a DHTML popup opens containing the corresponding glossary entry description.

Glossary entries can be viewed by clicking the **Glossary** item in the project editor tree.

By opening and adding entries to the glossary document within Doc-To-Help, you can quickly create and update your glossary as needed. The glossary document remains as your final document in the Help project by default.

Opening the Glossary in Microsoft Word

- Highlight the Document item on the project editor tree list.
- Select then right-click the glossary document and choose Open from the shortcut menu.

Note that the document has been opened within Microsoft Word.

Editing the Glossary in Doc-To-Help

- Highlight the Document item on the project editor tree list.
- Select and right-click the glossary document and choose **Edit** from the shortcut menu.

Note that the document has been opened within the right pane of the project editor.

Adding and Compiling Glossary Entries

- Open the glossary document in Word or within the Doc-To-Help project editor.
- Type the entry name on the first line following the Glossary heading and the entry description on the second line.
- Format the entry name with the Heading 5 paragraph style and the description text with normal text style so that they look like the entries in the illustration below.

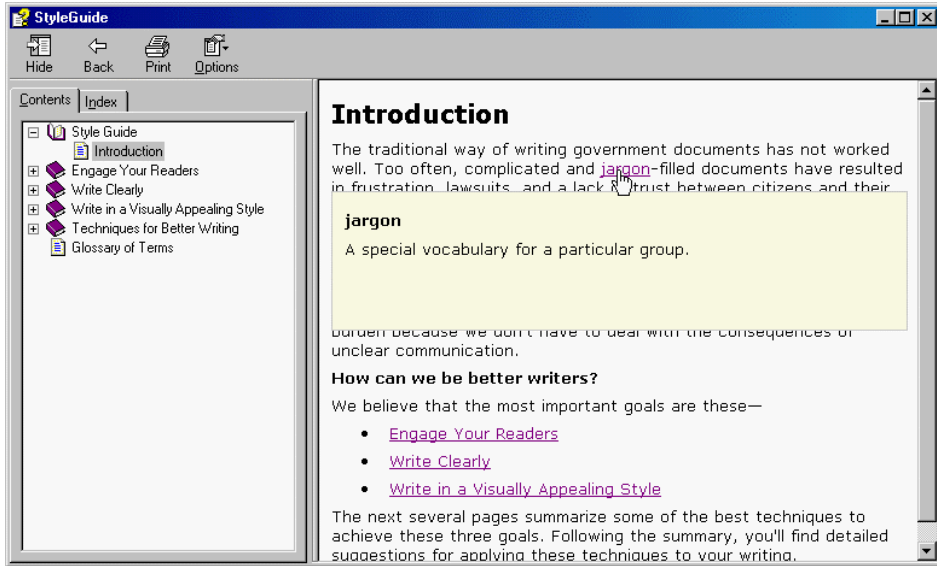
Active voice¶

A style of writing in which the subject of a sentence performs or causes the action expressed by the verb.¶

Jargon¶

A special vocabulary for a particular group.¶


- After you have finished entering and formatting your glossary entries, close save and close the Glossary.doc.
- Make sure that the value of the Glossary property, under Project Settings, is set to Glossary.doc
- Select the **Build** menu and click **Rebuild Target**. It is important to rebuild the entire target to ensure that recent changes will be included. After the target is compiled, click the **View Target** button. Notice that wherever the Help text matches your glossary entries (Heading 5), Doc-To-Help has formatted the text as a Dynamic HTML popup.



Using the Add Glossary Terms Tool

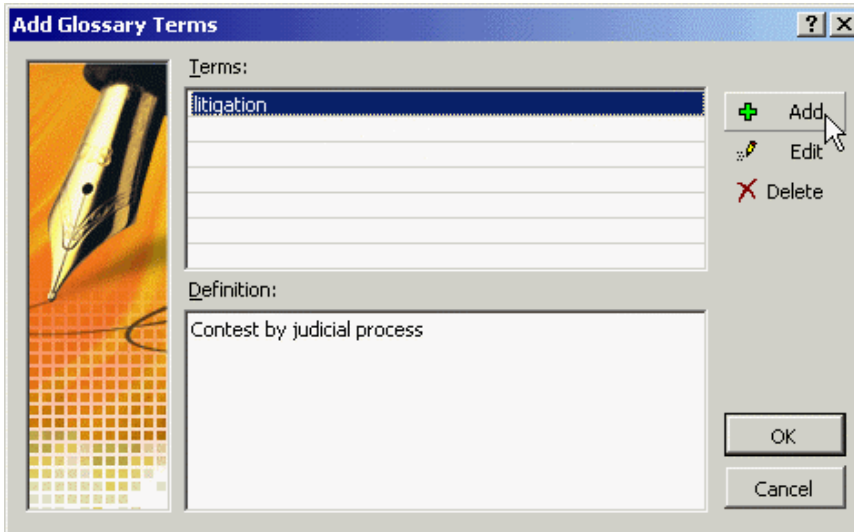
The **Add Glossary Terms** tool provides you with an easy way to add additional entries to your glossary without opening the glossary document itself. In this way, you can add new entries as you see them from any source document.

To add glossary entries:

- Open any of your source documents.
- Place your cursor in front of the word that you want to add to the glossary and click the Add Glossary Terms button .

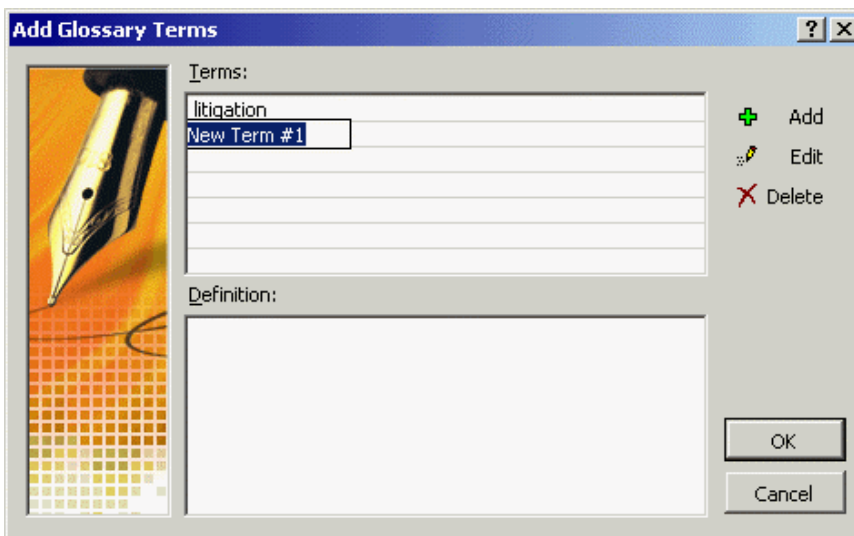
The **Add Glossary Terms** dialog opens with the word you selected in the terms box.

- Click inside the definition box and type the definition of the term.

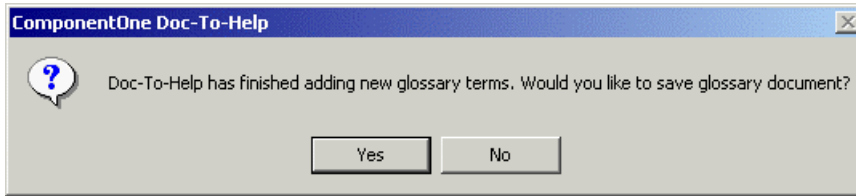


At this point, you can close the **Add Glossary Terms** dialog by clicking **OK**, or you can add additional terms by clicking the **Add** button.

- If you click the **Add** button, you are prompted to add new terms and definitions.




When you click the **OK** button you are prompted with the following message.



Alternatively, you can perform the same steps with the **Insert | ComponentOne Doc-To-Help | Glossary Terms** menu command.

Sorting Glossary Entries

To make alphabetizing your glossary terms simple, Doc-To-Help automatically sorts your glossary entries when you click the **Sort Glossary** button .

To sort your glossary entries:

- Open the Glossary source document.
- Click the **Sort Glossary** button.
- The glossary entries are automatically sorted alphabetically.
- **Save** and **Close** the glossary document.

Alternatively, you can perform the same steps with the **Tools | Sort Glossary** menu command.

Note: If you add additional entries after performing the sort, it will be necessary to resort the glossary.

Filtering Topics from the Glossary

In some cases, your topics may contain multiple instances of a specific glossary term. By default, Doc-To-Help inserts the glossary pop-up at the first instance of the term in each topic. If you want to modify your Doc-To-Help project so that it uses an instance of the glossary term other than the first, you can use the **D2HNoGloss** character style.

Simply apply the **D2HNoGloss** character style to each instance of a glossary term in a topic until you come to the term you want to designate as the pop-up. Do not apply **D2HNoGloss** to this term. You do not have to apply the **D2HNoGloss** style to subsequent terms because Doc-To-Help creates the pop-up at the first instance of the term that is not formatted with the **D2HNoGloss** style.

Note: If all instances of the glossary term are formatted with the **D2HNoGloss** style, Doc-To-Help does not create a glossary pop-up in that topic.

Deleting Help Topics

Doc-To-Help derives topics from text formatted with heading styles in your source documents. After removing any of the headings from your source documents, Doc-To-Help deletes the existing topic when the file is compiled. Additionally, if you modify a heading in your source documents, Doc-To-Help deletes the old topic from the database and adds the modified heading. In both cases, after compiling the Help project, the modified topic is marked as deleted in the topics pane of the project editor as show below.

Title	Document	Style
(Filter)	(All)	(All)
Break your material into short sections	Techniques.doc	Heading 2
Break your material into short sentences	Techniques.doc	Heading 2
Break your material into short, understandable sections	Techniques.doc	Heading 2
Carefully Develop your headings	Techniques.doc	Heading 2
Choose the right tone for your documents	Techniques.doc	Heading 2
citizens	StyleGuide.doc	D2HMarginNote
communicate clearly	StyleGuide.doc	D2HMarginNote
Cover only one topic in each paragraph	Techniques.doc	Heading 2
Define 'you' clearly	Techniques.doc	Heading 2
Develop your headings carefully	Techniques.doc	Heading 2
Engage Your Readers	StyleGuide.doc	Heading 1
Glossary of Terms	Glossary.doc	Heading 1


Removing Deleted Topics

To remove a topic that has been marked as “deleted” from the project editor:

In the right pane of the **Topics** view, select and right-click the deleted topic (marked with an X) to be removed and select **Delete** from the shortcut menu.

Alternatively, if you want to remove all deleted topics, you can right-click any deleted topic and select **Purge** from the shortcut menu.

Using HTML Help Object Tags

By clicking the **Insert HTML Help ActiveX Control** button , Doc-To-Help gives you access to the HTML Help ActiveX control contained within the HTML Help Workshop. Key features of the HTML Help ActiveX control include: an expanding table of contents, keyword search, shortcuts, and pop-up help topics.

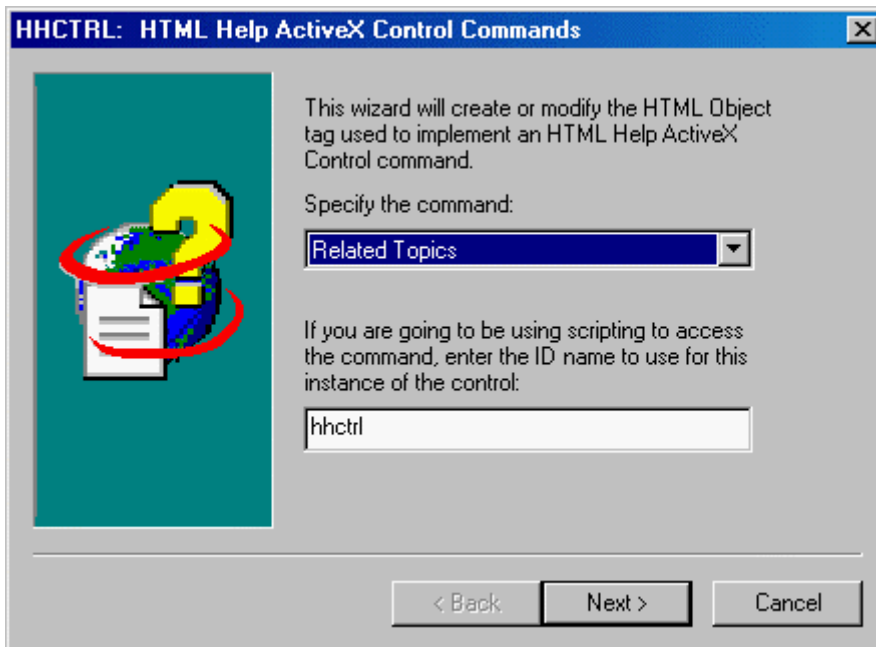
For HTML Help ActiveX Control documentation, visit the Microsoft HTML Help Workshop website.

Inserting an HTML Help Object Tag

Inserting the HTML Help Object Tag into a document involves adding a tag at the location you want the functionality to appear. The HTML Help ActiveX Control Wizard enables you to insert the object tag for the control.

- On the icon bar, select **Project**.
- In the left pane, click **Documents**.
- In the right pane, select then right-click the document you wish to edit and choose **Open** from the shortcut menu.
- Position the cursor at the location where you want the ActiveX control functionality to appear.
- Click **Insert HTML Help Object Tag** button.

The HTML Help ActiveX Control Wizard opens.



- Select the ActiveX object from the drop-down menu and follow the instructions provided by the wizard.

Once you are finished, Doc-To-Help marks the tag code as HTML PassThrough.

Alternatively, you can perform the same steps with the **Insert | ComponentOne Doc-To-Help | HTML Help ActiveX Control** menu command.

Using WinHelp Macros

By using the WinHelpMacro property, Doc-To-Help allows you easy access to any of the more than fifty macros which you can use to enhance the capabilities of your Help system. Macro uses include:

- Modifying the WinHelp button bar and menus.
- Adding keyboard shortcuts to WinHelp commands.
- Jumping to other Help files.
- Launching Windows applications.

Note: Help macros are only supported by WinHelp; not by HTML-based platforms.

Running WinHelp Macros

There are four ways you can run a Help macro.

Startup Macros

A Startup macro is run whenever the user loads the help file. Examples of Startup macros include adding new buttons and menus, registering a function or functions in a Dynamic Link Library (DLL) and playing sound. The settings for Startup macros are stored in the [Config] section of the Help Project File (HPJ File).

Help Topic Macros

You can create Windows Help macros that start whenever the user enters a certain topic. This is useful for customizing topics with buttons, menu items, and markers. For example, you may want certain buttons and menu items to be available only when a certain topic is displayed. Or perhaps you want to play a particular sound when displaying a topic.

Hotspot Macros

Hotspot macros look like jump text, but when you click on them they execute Help macros. Examples of Hotspot macros are text that when clicked runs another Windows application or prints the help topic.

Window Macros


Window macros are associated with the windows defined in the Project Editor. They execute when a window is activated. You might, for example, want to play a warning sound when an error message window is displayed. Window macros are added in the Macros tab of the Window Settings dialog box.

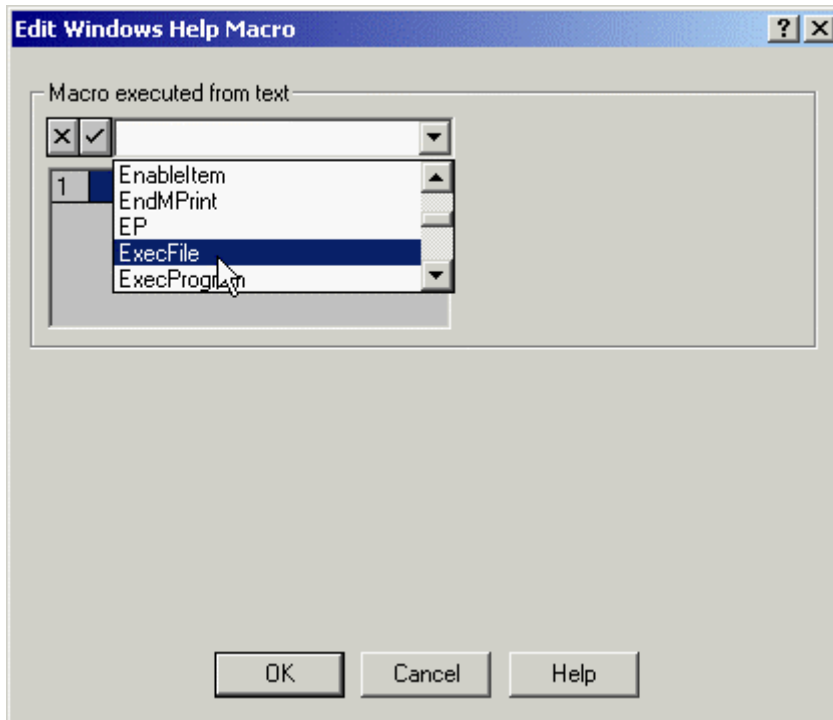
Sample WinHelp Macro Uses

The following sections give a brief overview of some of the enhancements that can be incorporated into your WinHelp files. In many cases the same macros can be run in startup, topic, hotspot or windows modes.

Launching an Application with Hotspot Macros

This procedure turns any text you choose into a hotspot that will open any application located in your Windows registry. In the following example, when the hotspot is clicked, Windows Notepad opens.

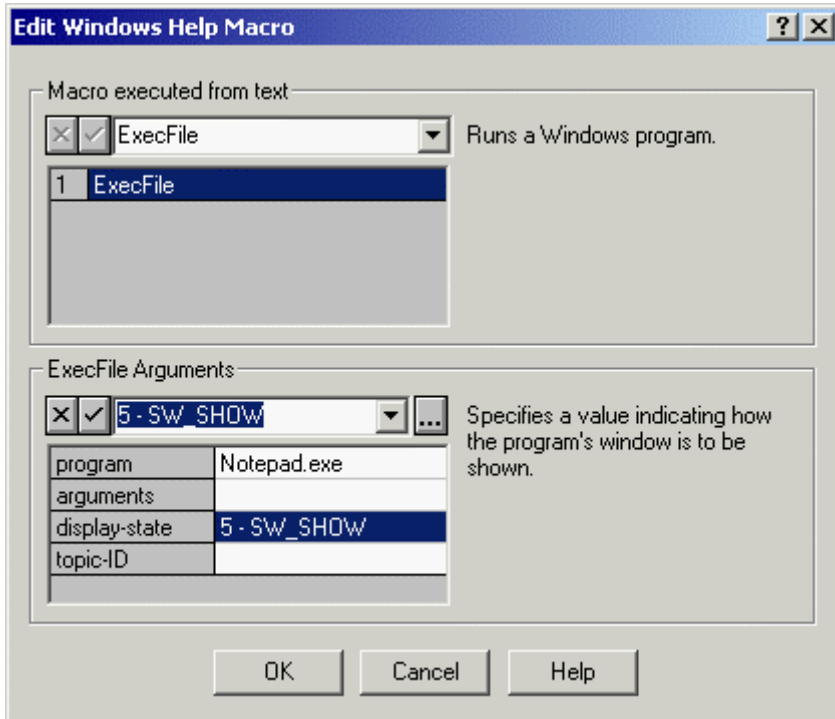
- Open the source document that will contain the hotspot.
- Select the text for the hotspot and click the **Insert Help Macro** button .
- The Edit Windows Help Macro dialog opens.
- From the drop-down list box in the “Macro executed from text” area, select **ExecFile**.



- Type **Notepad.exe** in the textbox at the top of the “ExecFile Arguments” area and click the checkmark button.

- Select the display-state box, and from the drop-down at the top of the ExecFile Arguments area, select the value, **5 - SW_SHOW**. Display-state determines how the application appears when it is started. In this case you want the Notepad.exe to open as it normally would when the hotspot is clicked.

Your dialog should look like the example below.



- Click **OK** to insert the Macro Comment into your source document.
- **Close** the source document and build your WinHelp file.
- Open your WinHelp file and click the hotspot to test the macro.

Alternatively, you can perform the same steps with the **Insert | ComponentOne Doc-To-Help | Help Macro** menu command.

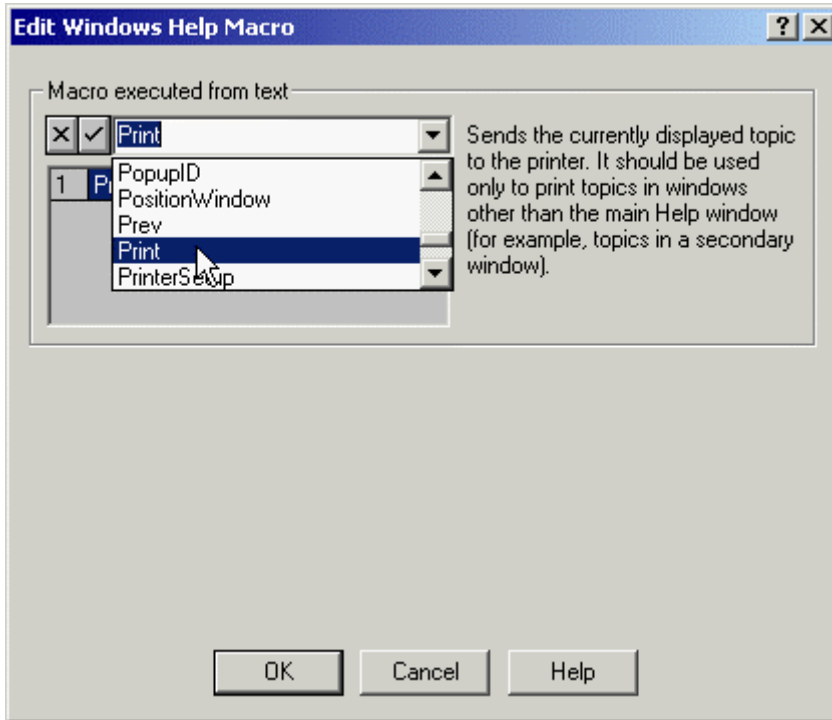
Printing with Hotspot Macros

This procedure turns any text you choose into a hotspot that will print your current Help topic.

- Open the source document that will contain the hotspot.
- Select the text for the hotspot and click the Insert Help Macro button.

The Edit Windows Help Macro dialog opens.

- From the drop-down list box in the “Macro executed from text” area, select **Print**.



- Click **OK** to insert the Macro Comment into your source document.
- **Close** the source document and build your WinHelp file.
- Open your WinHelp file and click the hotspot to test the macro.

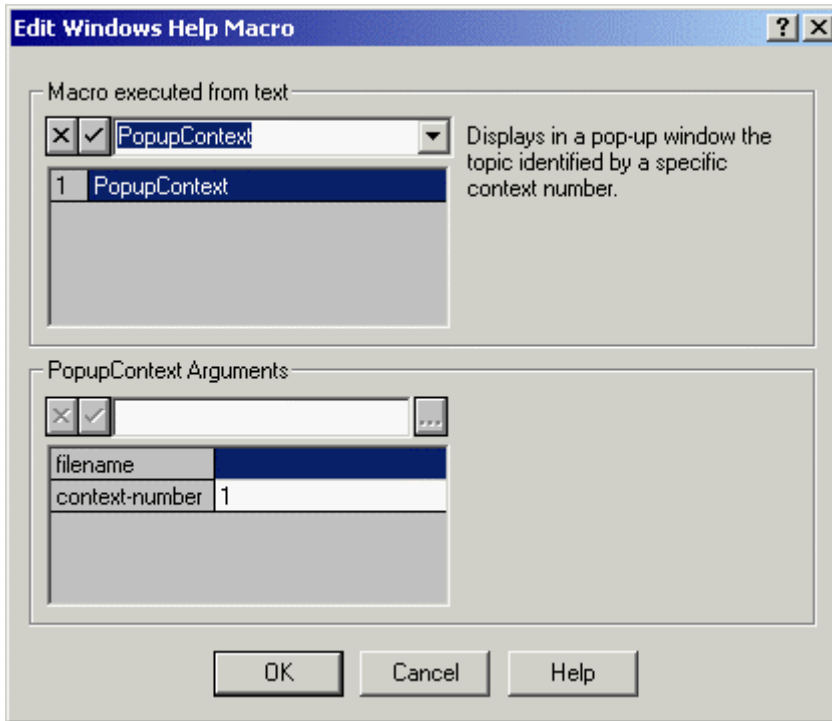
Alternatively, you can perform the same steps with the **Insert | ComponentOne Doc-To-Help | Help Macro** menu command.

Displaying Help File Topic with Hotspot Macros

You can also use hotspot macros to perform certain Help-specific commands. This procedure allows you to display a Help topic which is located in a separate Help file.

- Open the source document that will contain the hotspot.
- Select the text for the hotspot and click the **Insert Help Macro** button.
- The Edit Windows Help Macro dialog opens.

- From the drop-down list box in the “Macro executed from text” area, select **PopupContext** (or its short form, **PC**).

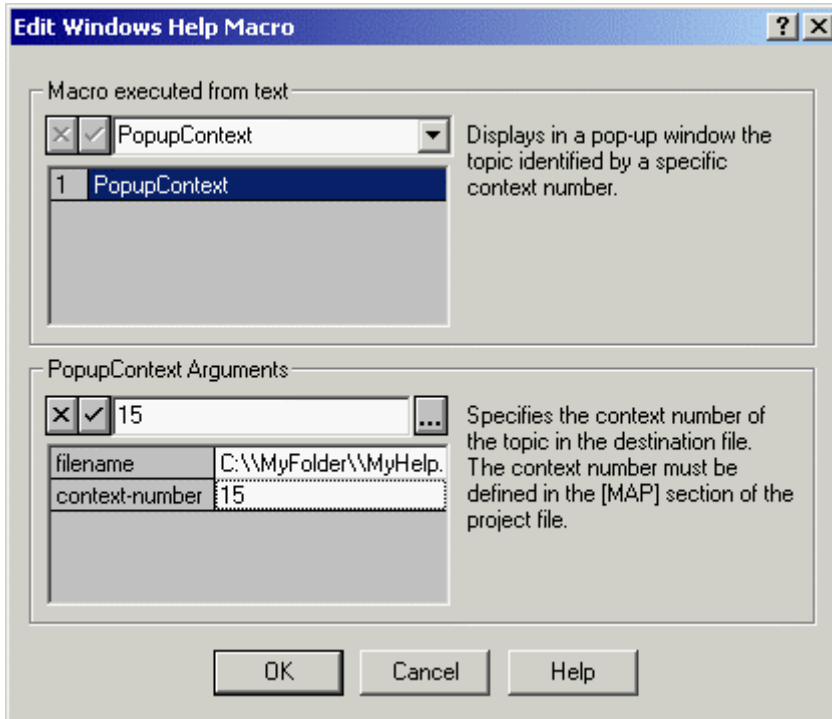


- Type the name and path of your second Help file in the text box at the top of the “PopupContexts Arguments” area and click the checkmark button.

Note: When defining the path to the Help file, use double back-slashes (\\) in place of the typical single back-slash. Example – C:\\MyFolder\\MyFile.hlp.

- Select the context-number box, and type the Context ID number of the topic you wish to display. For more information on Context Ids, see Context Sensitive Help (page 186).

Your dialog should look similar to the example below.



- Click **OK** to insert the Macro Comment into your source document.
- **Close** the source document and build your WinHelp file.
- Open your WinHelp file and click the hotspot to test the macro.

Alternatively, you can perform the same steps with the **Insert | ComponentOne Doc-To-Help | Help Macro** menu command.

Adding New Menus and Menu Items with Startup Macros

In the procedure below we will create several macros that add a new menu and new menu item when a Help file is loaded. In particular, we will add a new menu called Programs that has a menu item called Excel. Selecting Excel will load Microsoft Excel.

- Open your Help project.
- Select the **Project** icon, then **Help Targets** from the left pane.
- Select your WinHelp target from the right pane.
- From the property pane, click the WinHelpMacro property.

Property	Value
MapFileForC	False
MapFileForVB	False
MapNumbers	Automatic+Custom
Name	MacroTest WinHelp
SkipGlossary	False
Template	C1H_HELP.DOT
Title	
WinHelpMacro	

The Edit Windows Help Macro dialog opens.

- From the drop-down list box in the “Macro executed when help file is loaded” area, select InsertMenu.

Edit Windows Help Macro

Macro executed when help file is loaded

InsertMenu Inserts a new menu in the Windows Help menu bar.

1	InsertMenu
---	------------

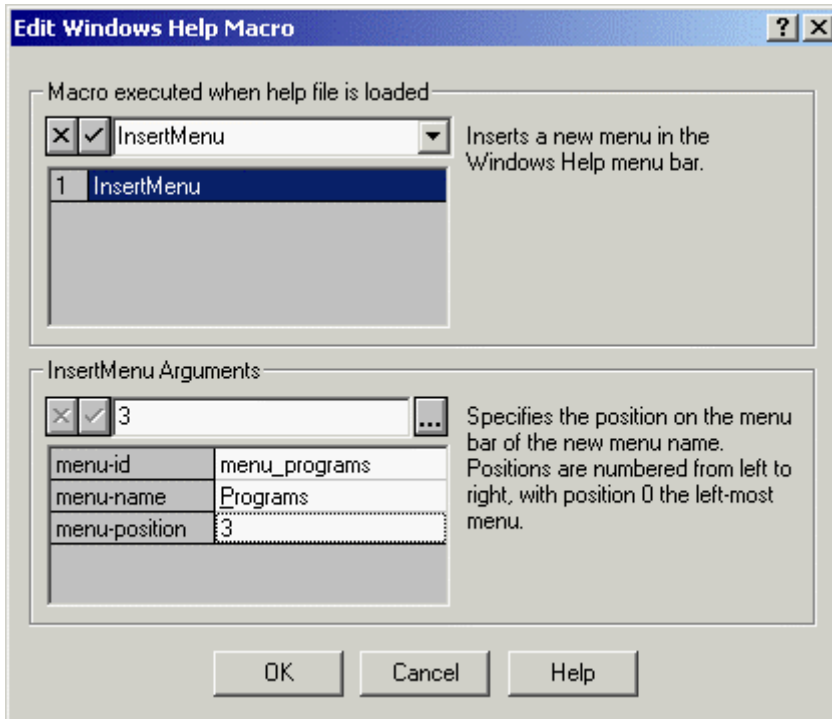
InsertMenu Arguments

menu-id	
menu-name	
menu-position	0

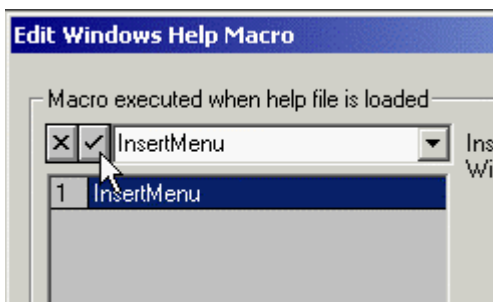
OK Cancel Help

- Select the menu-id box and type **menu_programs** in the textbox at the top of the InsertMenu Arguments area. This is the name you use to identify the menu. It can be any unique identifier. You can also add items to Help's Standard menus.
- Select the menu-name box and type **&Programs** in the textbox at the top of the InsertMenu Arguments area. This is the text that you want to appear on the menu. The character following the ampersand is the hotkey.

- Select the menu-position box and type **3** in the textbox at the top of the InsertMenu Arguments area. This specifies the position on the menu bar of the new menu name. Positions are numbered from left to right, with the leftmost being position 0.
- Click the InsertMenu Arguments checkmark button.

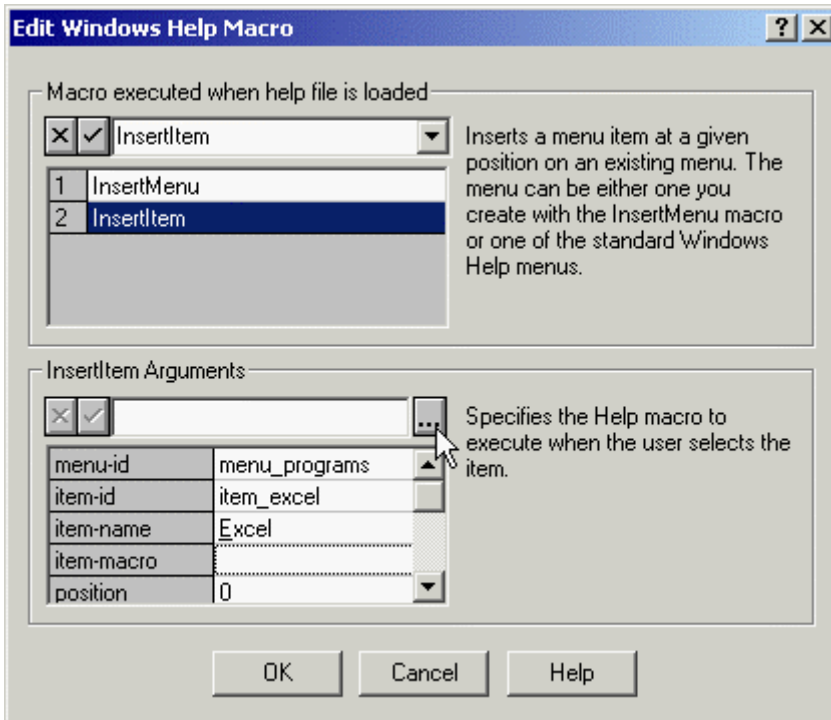


- In the “Macro executed when help file is loaded” click the checkmark button. A new blank cell opens beneath the InsertMenu macro.



At this point we have added a new menu called Programs which will appear to the right of the Bookmark menu. In the next series of steps we will add the menu item "Excel" that when clicked will run Microsoft Excel.

1. From the drop-down list box, select **InsertItem** for the second cell.
2. Select the menu-id box and type **menu_programs** in the textbox at the top of the InsertItems Arguments area. This must be a unique identifier to identify the menu or one of the names Help uses to identify its standard menus (if you're adding to a standard menu).
3. Select the item-id box and type **item_excel** in the textbox at the top of the InsertItems Arguments area. This is the label Help uses internally to identify the menu item. Use any label, but it must be different from any other item-id you have used.
4. Select the item-name box and type **&Excel** in the textbox at the top of the InsertItems Arguments area. This will place the word Excel on the Programs menu and make the letter "E" its keyboard shortcut.
5. Select item-macro, then click the ellipsis button.

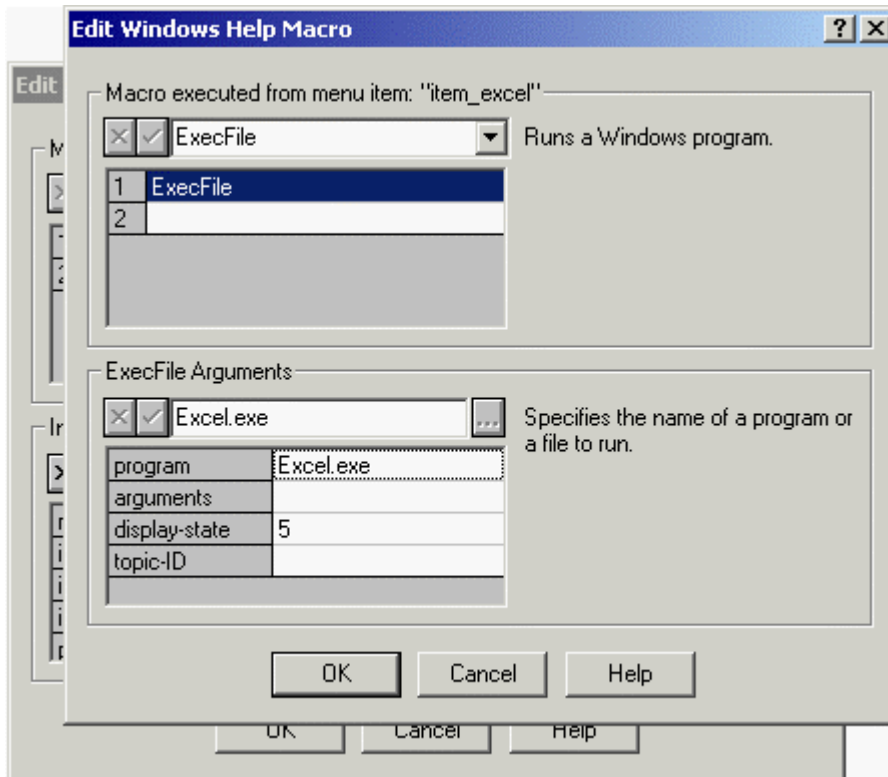


A second Help Macro dialog box appears. You have to specify which macro you want executed when the user clicks on the menu item Excel. In this example,

when the menu item is clicked, Excel loads. This second dialog box allows you to specify the macro and arguments you want executed when the user clicks the Excel menu item.

- From the drop-down list box in the “Macro executed from menu item. ‘item_excel’” area, select **ExecFile**.
- Type **Excel.exe** in the textbox at the top of the “ExecFile Arguments” area and click the checkmark button.
- Select the display-state box, and from the drop-down at the top of the ExecFile Arguments area, select the value, **5 - SW_SHOW**.

Your dialog should look like the example below.



- Click **OK**. The first Help Macro dialog box reappears and the ExecProgram macro appears in the item-macro box in the InsertItem Arguments section.
- In the position box, type **0**. This is the number specifying where the new menu item will appear. The number must be an integer. The first item on a menu is represented by 0, the second item is 1, and so forth. So in this example the Exec will be the first item in the Programs menu.

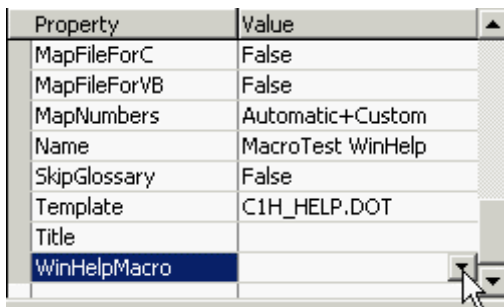
- Click the checkmark button in the "InsertItem Arguments" area. This completes the arguments.
- Click the checkmark button in the "Macro executed when help file is loaded" area. This completes the macro definition.
- Click **OK**. Doc-To-Help adds the appropriate macro to your Help Project File (HPJ File).
- Build your WinHelp file.
- Open your WinHelp file and test the menu.

Adding a Button with Startup Macros

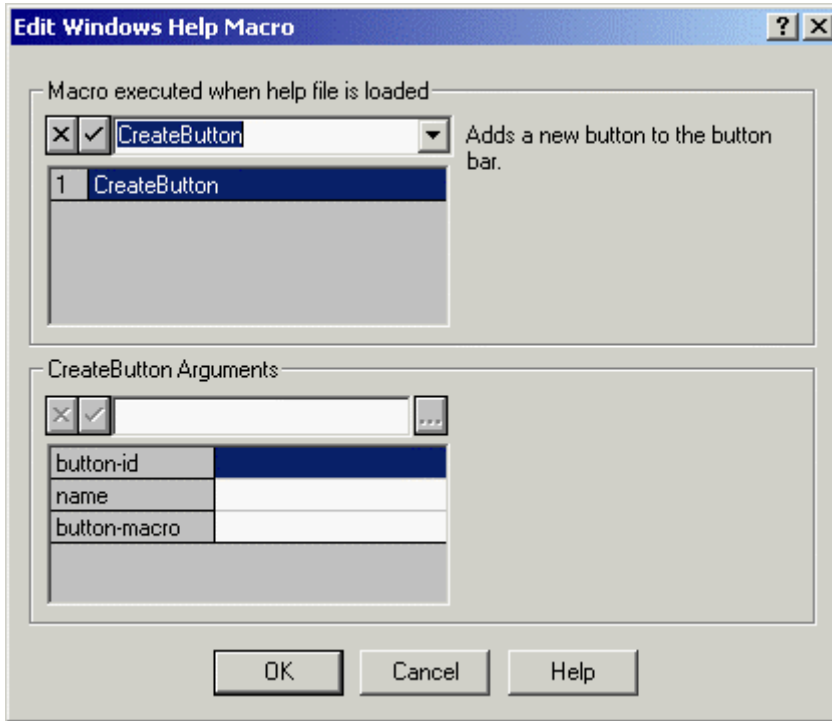
You can use startup macros to create a button that appears every time a user opens the Help file. The following example creates a button labeled WinHelp Help that appears on the button bar every time the Help file is open. When the button is clicked, it takes the user to the WinHelp help file.

In this example we will use "nested" macros. You can have a single macro use up to two more additional macros. This nesting capability lets you combine the features of macros. In this example, two macros are used: CreateButton and ExecFile. ExecFile is the nested macro.

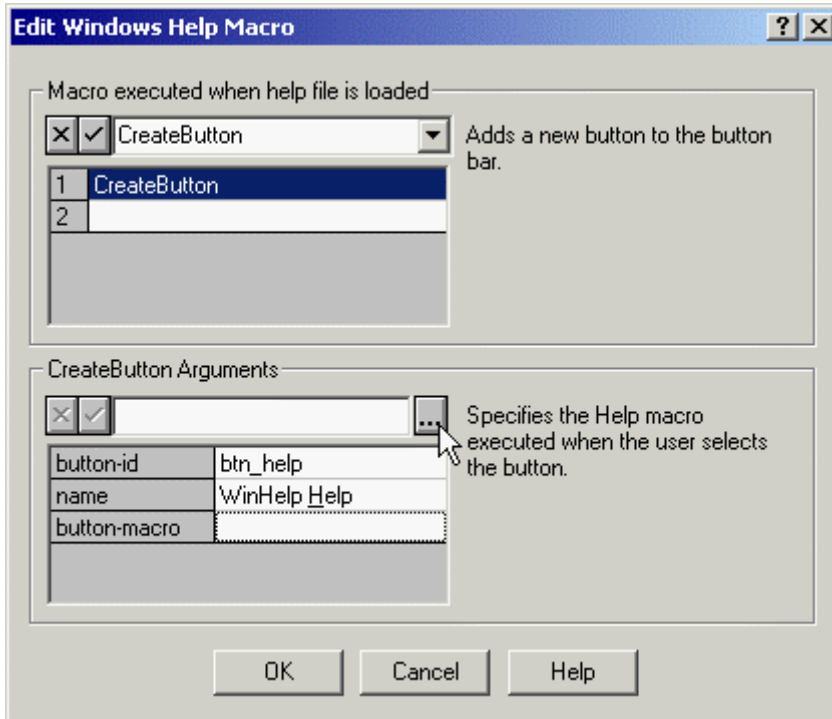
- Open your Help project.
- Select the **Project** icon, then **Help Targets** from the left pane.
- Select your WinHelp target from the right pane.
- From the property pane, click the WinHelpMacro property.



- The Edit Windows Help Macro dialog opens.
- From the drop-down list box in the "Macro executed when help file is loaded" area, select **CreateButton**.



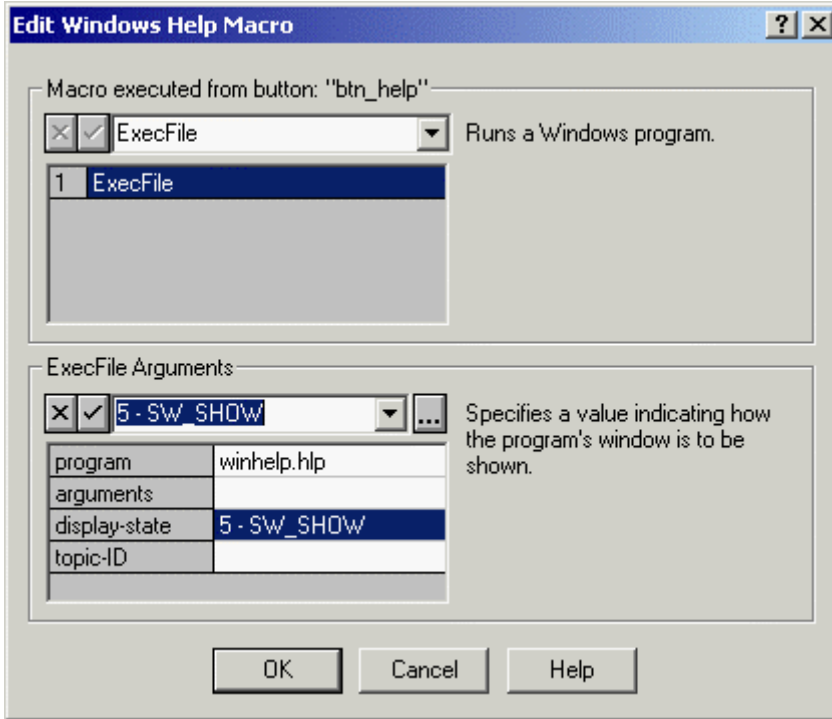
- Select the button-id box and type **btn_help** in the textbox at the top of the CreateButton Arguments area. This is the name you use to identify the button. It can be any unique identifier.
- Select the name box and type **WinHelp &Help** in the textbox at the top of the CreateButton Arguments area. This is the text that you want to appear on the button. The character following the ampersand is the hotkey.
- Select the button-macro cell and click the ellipsis button as shown below.



A second Help Macro dialog box appears. You specify here which macro you want executed when the user clicks on the WinHelp Help button. In this example, when the button is clicked, the WinHelp Help file opens.

1. From the drop-down list box, select **ExecFile**. The ExecFile Arguments section appears in the dialog box.
2. Select the program box and type **winhelp.hlp** in the textbox at the top of the ExecFile Arguments area. This is the file that will appear when the user clicks on the button.
3. Select the display-state box, and from the drop-down at the top of the ExecFile Arguments area, select the value, **5 - SW_SHOW**.

Your dialog should look like the example below.



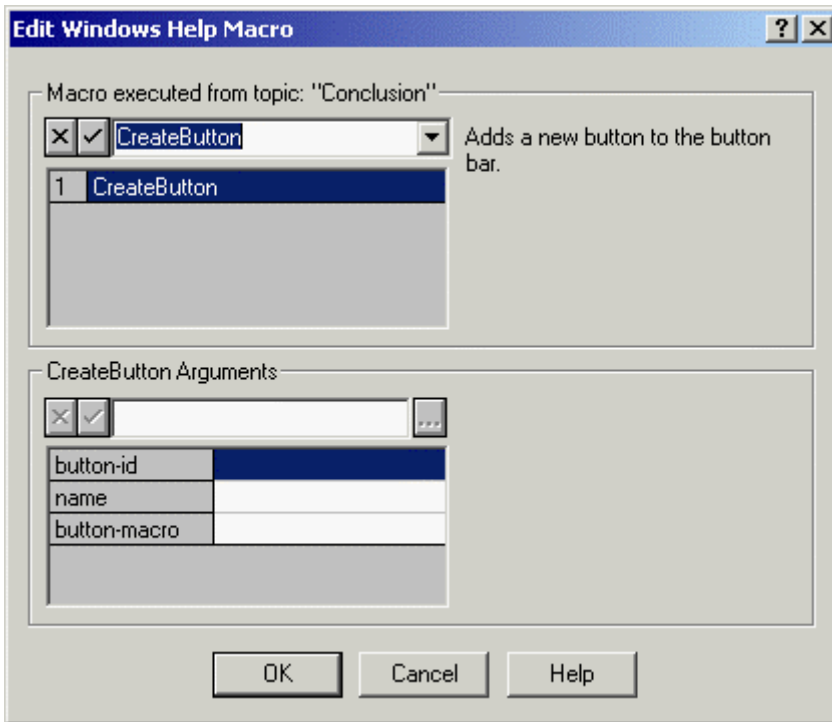
4. Click **OK** to return to the first dialog box.
5. Click **OK**.
6. Build your WinHelp file.
7. Open your WinHelp file and test the button.

Adding a Button with Topic Macros

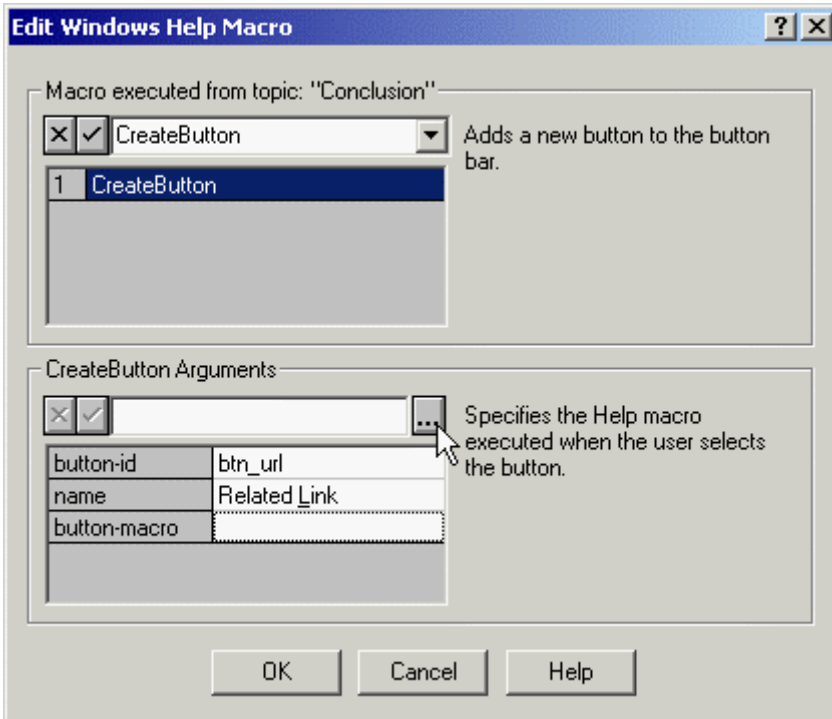
You can create Windows Help macros that start whenever the user enters a certain topic. This is useful for customizing topics with buttons, menu items, and markers. The procedure below uses "nested" macros. In this example, you will provide a button that takes the user to a URL you have defined.

- Open your Help project.
- Select the **Topics** icon.
- Select the topic you want to use from the right pane.
- From the property pane, click the WinHelpMacro property.
- The Edit Windows Help Macro dialog opens.

- From the drop-down list box in the “Macro executed from topic:” area, select **CreateButton**.



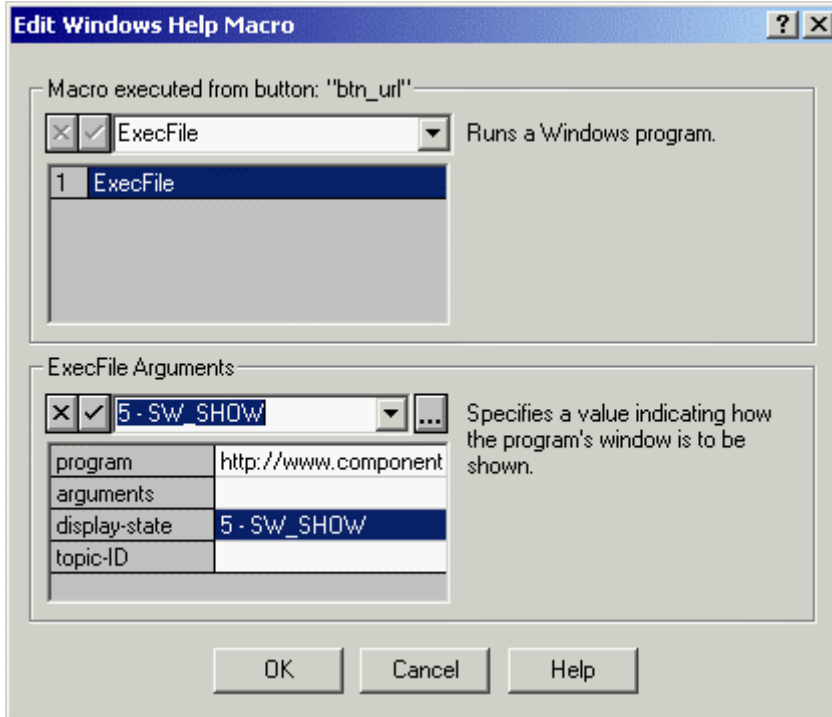
- Select the button-id box and type **btn_url** in the textbox at the top of the CreateButton Arguments area. This is the name you use to identify the button. It can be any unique identifier.
- Select the name box and type **Related &Link** in the textbox at the top of the CreateButton Arguments area. This is the text that you want to appear on the button. The character following the ampersand is the hotkey.
- Select the button-macro cell and click the ellipsis button as shown below.



A second Help Macro dialog box appears. You specify here which macro you want executed when the user clicks on the Related Link button. In this example, when the button is clicked, a browser opens with the ComponentOne website.

- From the drop-down list box, select **ExecFile**. The ExecFile Arguments section appears in the dialog box.
- Select the program box and type **http://www.componentone.com** in the textbox at the top of the ExecFile Arguments area.
- Select the display-state box, and from the drop-down at the top, select the value, **5 - SW_SHOW**.

Your dialog should look like the example below.



- Click **OK** to return to the first dialog box.
- Click **OK**.
- Build your WinHelp file.
- Open your WinHelp file, go to the topic you used for your button, and test the button.

Editing and Deleting Macros

Startup, Topic and Window Macros

Startup, topic and window macros can be edited or removed by selecting and modifying the related WinHelpMacro property as needed.

Hotspot Macros

Hotspot macros can be deleted by removing the corresponding comment from the source document. Modify hotspot macros by deleting the existing comment and reinserting the new macro.

Macro Definitions

Following is an alphabetical listing of all the Windows Help Macros. When you use the Doc-To-Help Macro Editor, you'll see an explanation for each macro when you select that macro from a drop-down list. More details on Help Macros can be found in the Help Author's Guide.

Macro Name	Description
About	Displays the About dialog box.
AddAccelerator (AA)	Assigns a Help macro to an accelerator key (or key combination) so that the macro is run when the user presses the accelerator key(s).
ALink	Searches for keywords specified by A footnotes.
Annotate	Displays the Annotation dialog box.
AppendItem	Appends a menu item to the end of a menu you create with the InsertMenu macro.
Back	Displays the previous topic in the Back list. The Back list includes the last 40 topics the user has displayed since starting WinHelp.
BackFlush	Removes the back history list from the current window. This macro does not affect the history list displayed in the History window.
BookmarkDefine	Displays the Define dialog from the Bookmark menu.
BookmarkMore	Displays the More dialog from the Bookmark menu. The More command appears on the Bookmark menu if the menu lists more than nine bookmarks.
BrowseButtons	Adds browse buttons to the button bar.
ChangeButtonBinding	Assigns a Help macro to a Help button.
ChangeEnable (CE)	Assigns a macro to a button bar button and enables that button.
ChangeItemBinding (CIB)	Assigns a Help macro to an item previously added to a Windows Help menu using the AppendItem macro.
CheckItem (CI)	Places a checkmark beside a menu item.
CloseSecondaries (CS)	Closes all but the current secondary window.
CloseWindow	Closes either a secondary window or the main Help window.

Compare	Displays a Help file in a second instance of WinHelp. The current Help file and the second Help file are displayed side-by-side. Most actions performed in one Help file will be automatically reflected in the other file.
Contents	Displays the Contents topic in the current Help file.
ControlPanel	Opens a control panel applet with a specific tab on top.
CopyDialog	Displays the Copy dialog from the Edit menu.
CopyTopic	Copies all the text in the currently displayed topic to the Clipboard.
CreateButton (CB)	Adds a new button to the button bar.
DeleteItem	Removes a menu item that was added by using the AppendItem macro.
DeleteMark	Removes a text marker added with the SaveMark macro.
DestroyButton (DB)	Removes a button added with the CreateButton macro.
DisableButton	Grays out a button added with the CreateButton macro. This button cannot be used in the topic until an EnableButton macro is executed.
DisableItem (DI)	Grays out a menu item added with the AppendItem macro. The menu item cannot be used in the topic until an EnableItem macro is executed.
EnableButton (EB)	Re-enables a button disabled with the DisableButton macro.
EnableItem (EI)	Re-enables a menu item disabled with the DisableItem macro.
EndMPrint	Dismisses the printing message box and terminates the printing of multiple topics.
ExecFile (EF)	Runs a program or runs a program and opens a specified file.
ExecProgram (EP)	Executes a Windows application.
Exit	Exits the Windows Help application: the same effect as selecting Exit from the File menu.
ExtAbleItem	Enables or disables a menu item.
ExtInsertItem	Inserts a menu item at a given position on an existing menu.

ExtInsertMenu	Inserts a submenu in a previously defined menu.
FileExist	Checks to see whether the specified file or program exists.
FileOpen	Displays the Open dialog box from the File menu.
Find	Displays the Find tab in the Help Topics dialog box.
Finder	Displays the Help Topics dialog box.
FloatingMenu	Displays the context (floating) menu at the current mouse cursor position. This menu also appears when users click a topic using their right mouse button.
Flush (FH)	Causes WinHelp to process any pending messages, including previously called macros.
FocusWindow	Changes the focus to the specified window, either the main Help window or a secondary window.
Generate	Posts a message to the currently active Help window.
GotoMark	Jumps to a marker set with the SaveMark macro.
HelpOn	Displays the Help file for the Windows Help application. The macro carries out the same action as choosing the How to Use Help command on the Help menu.
HelpOnTop	Toggles the authored on-top state of a Help window.
History	Displays the history list, which shows the last 40 topics the user has viewed since opening a Help file in Windows Help. It has the same effect as choosing the History button.
IfThen	Executes a Help macro if a given marker exists, using the IsMark macro to make the test. The result of the test can be reversed by enclosing the IsMark macro within the Not macro.
IfThenElse	Executes one of two Help macros depending on whether or not a marker exists, as tested by the IsMark macro. The result of the test can be reversed by enclosing the IsMark Macro within the Not macro.
InitMPrint	Initializes WinHelp in preparation for printing multiple topics.
InsertItem	Inserts a menu item at a given position on an existing menu. The menu can be either one you create with the InsertMenu macro or one of the standard Windows Help menus.

InsertMenu	Inserts a new menu in the Windows Help menu bar.
IsBook	Determines whether WinHelp is running as a standalone system (a double-clicked book icon), or if it is being run from a program. This macro can be used as the first parameter of an IfThen or IfThenElse macro to take specific action depending on whether the current Help file is being run as a double-clicked book icon.
IsMark	Used with IfThen and IfThenElse to determine if a text marker previously created with the SaveMark macro exists.
IsNotMark	Tests whether or not a marker that was set by the SaveMark macro exists.
JumpContents	Jumps to the Contents topic of a specified Help file.
JumpContext (JC)	Jumps to a topic identified by a context number. The context is identified by an entry in the [MAP] section of the HPJ file.
JumpHash	Jumps to a topic identified by a hash number.
JumpHelpOn	Jumps to the Contents topic of the How to Use Help file. The How To Use Help file is either the default WINHELP.HLP or WINHLP32.HLP.
JumpId (JI)	Jumps to the topic with the specified context string in the Help file.
JumpKeyword (JK)	Loads the indicated Help file, searches through the K keyword table, and displays the first topic containing the index keyword specified in the macro.
KLink	Searches for keywords specified by K-footnotes.
MPrintHash	Prints a topic identified by a hash number. This macro must be used in conjunction with the InitMPrint and EndMPrint macros.
MPrintId	Prints a topic. This macro must be used in conjunction with the InitMPrint and EndMPrint macros.
Next	Displays the next topic in the browse sequence for the Help file.
NoShow	Prevents a Help window from being displayed if it has not already been displayed.

Not	Returns zero (FALSE) if the marker text specified by the SaveMark macro exists or non-zero (TRUE) if the marker text does not exist. When used with a macro, the Not macro reverses the results of the macro. Used with IfThen and IfThenElse to determine if a text marker previously created with the SaveMark macro does not exist.
PopupContext (PC)	Displays in a popup window the topic identified by a specific context number.
PopupHash	Displays in a popup window the topic identified by a hash number.
PopupId (PI)	Displays a topic from a specified file in a popup window.
PositionWindow	Sets the size and position of a window.
Prev	Displays the previous topic in the browse sequence for the Help file. If the currently displayed topic is the first topic of a browse sequence, this macro does nothing.
Print	Sends the currently displayed topic to the printer. It should be used only to print topics in windows other than the main Help window (for example, topics in a secondary window).
PrinterSetup	Displays the Printer Setup dialog box from the File menu.
RegisterRoutine (RR)	Registers a function within a DLL as a Help macro.
RemoveAccelerator	Removes the assignment of a macro to an accelerator key or key combination.
ResetMenu	Deletes all added menus and menu items, restores and enables all standard menu items, and restores the item bindings of all standard menu items to their defaults.
SaveMark	Saves the location of the currently displayed topic and file and associates a text marker with that location. The GotoMark macro can then be used to jump to this location.
Search	Displays the dialog for the Search button, which allows users to search for topics using keywords defined by the K footnote character.
SetContents	Designates a specific topic as the Contents topic in the specified Help file.

SetHelpOnFile	Designates the specific Help file that replaces WINHELP.HLP, the default Using Help file in the Windows environment. This macro is obsolete in Windows 95 and will be ignored.
SetPopupColor	Sets the background color for all subsequent popup windows.
ShellExecute	Opens or prints the specified file.
ShortCut	Runs the specified program if it is not already running. If the specified program is running, WinHelp activates it. If the wParam parameter is specified, a WM_COMMAND message with the specified wParam and lParam values are sent to the program.
TCard	Sends a message to the program that is invoking WinHelp as a training card.
Test	Runs an internal WinHelp test.
TestALink	Tests whether an ALink macro has an effective link to at least one topic.
TestKLink	Tests whether a KLink macro has an effective link to at least one topic.
UncheckItem (UI)	Removes a check mark besides a menu item.
UpdateWindow	Jumps to the topic with the specified topic ID in the specified window, and then returns the focus to the window that called the macro.

Doc-To-Help Shortcut Keys

Doc-To-Help provides users with shortcut keys to automate some of the most common commands used while in Microsoft Word.

Key Combination	Result
Ctrl + Numeric Keypad Plus	Adds one point of space after the current paragraph.
Ctrl + Alt + Numeric Keypad Plus	Adds one point of space before the current paragraph.

Ctrl + Numeric Keypad Minus	Subtracts one point of space after the current paragraph.
Ctrl + Alt + Numeric Keypad Minus	Subtracts one point of space before the current paragraph.
Ctrl + Shift + 1	Applies the Heading 1 style.
Ctrl + Shift + 2	Applies the Heading 2 style.
Ctrl + Shift + 3	Applies the Heading 3 style.
Ctrl + Shift + 4	Applies the Heading 4 style.
Ctrl + Shift + B	Applies the Body text style.
Ctrl + Shift + G	Runs the Sort Glossary command.
Ctrl + Shift + H	Runs the Insert HTML Help ActiveX Control command.
Ctrl + Shift + L	Applies the List style.
Ctrl + Shift + M	Runs the Insert Help Macro command.
Ctrl + Shift + S	Applies special bold formatting.
Ctrl + Shift + T	Runs the Standard Doc-To-Help Table command.
Ctrl + Shift + W	Runs the Apply Conditional Text command.
Ctrl + Shift + X	Runs the Add Dynamic Link command.
Ctrl + Shift + Y	Opens the Add Topic Link dialog box.
Ctrl + F1	Help on Doc-To-Help.
Ctrl + F5	Runs the View Target command.
F5	Runs the Make Target command.
Alt + 0	Applies the C1H Continue or C1H Continue 2 style.
Alt + 1	Applies the C1H Number style.
Alt + 2	Applies the C1H Number 2 style.
Alt + 3	Applies the C1H Bullet style.
Alt + 4	Applies the C1H Bullet 2 style.
Alt + 5	Applies the C1H Bullet 2A style.

Alt + Shift + C	Runs the View Conditional Text command.
Alt + Shift + G	Runs the Add Glossary Terms command.
Alt + M	Inserts a margin note.
Alt + Shift + M	Inserts a link to a margin note.
Alt + Shift + R	Completes a cross-reference.
Alt + F12	Switches to the Project Editor.

Conditional Text and Attributes

As you write documentation, you will no doubt write passages that you don't want included in the accompanying online Help file. Likewise, some content may be appropriate for the Help file but not the written documentation. You may even have content that you want targeted at specific audiences. Yet, if you are looking for the ease of single-source Help authoring, making separate files for each platform would defeat this entire concept.

There's a better solution: Doc-To-Help lets you mark text (or graphics) to be included in or excluded from either the manual or any combination of online Help platforms using the conditional text and attribute features.

Specifying Conditional Text

There is more to single-source authoring than merely converting documents to the appropriate format. This is particularly true when designing a Help system for both online and print media. In many cases creating unique content for different platforms would better suit your audience. Here are some examples in which deviating from the single-source philosophy has its advantages:

- You use graphics in the printed manual, but omit them from the online Help to conserve screen real estate and minimize the time needed to load pages.
- You provide links to related topics in the online Help, but omit them from the printed manual to avoid disrupting the narrative flow.
- You need to describe a set of procedures, each of which shares a common set of preliminary steps. In the online Help, you include the preliminary steps at the beginning of each procedural topic to provide adequate context for the reader. In the printed manual, you describe the preliminary steps once, immediately before the procedural topics, to keep them from becoming tedious and repetitive.

Fortunately, Doc-To-Help's conditional text feature can help resolve design issues such as these. By allowing you to designate portions of your source documents for individual platforms, Help targets or attributes (or combinations thereof), Doc-To-Help gives you full control over the content of each one – without sacrificing the benefits of having a single-source project.

Doc-To-Help implements conditional text using the Microsoft Word comments feature. To specify conditional text, you select a range of text in Word, then choose the platform, Help target or attribute in which you want the range to be visible. Since this

implementation uses standard Word constructs, you can subsequently edit a document marked in this manner on other computers.

Conditional Text Target Options

When defining conditional text, Doc-To-Help provides three target options; **Platforms**, **Help Targets** and **Attributes**.

Platforms	This option allows you to incorporate conditional text into any Help file defined with a specific platform.
Help Targets	This option allows you to incorporate conditional text into a specific default or user defined Help target.
Attributes	This option allows you to incorporate conditional text into a specific version of a Help file.

Platforms

HTML (Any)	If chosen, conditional text is viewable in all HTML and HTML Help help targets.
HTML 4.0	If chosen, conditional text is viewable only in the HTML 4.0 targets.
HTML Help 1.x	If chosen, conditional text is viewable only in HTML Help help targets.
JavaHelp 1.1	If chosen, conditional text is viewable only in JavaHelp 1.1 help targets.
Microsoft Help 2.0	If chosen, conditional text is viewable only in Microsoft 2.0 help targets.
Online Help	If chosen, conditional text is viewable only in all online help targets.
Printed Manual	If chosen, conditional text is viewable only in printed manual help targets.
WinHelp 4.0	If chosen, conditional text is viewable only in WinHelp help targets.

Help Targets


Microsoft Help 2.0	This target produces the standard Microsoft Help 2.0. If chosen, conditional text will only be viewable in this target type.
HTML 4.0	This target produces browser neutral HTML in accordance with version 4.0 of the HTML specification published by W3C. If chosen, conditional text will only be viewable in this target type.
HTML Help	This target produces Microsoft HTML Help, which requires Internet Explorer to be installed on the client machine. Internet Explorer version 3.0 is acceptable, but 4.0 or greater is preferred. If chosen, conditional text will only be viewable in this target type.
Printed Manual	This target produces a Microsoft Word document that includes a table of contents, an index and the individual subdocuments generated from your source documents. If chosen, conditional text will only be viewable in this target type.
WinHelp	This target produces the standard 32-bit Windows Help 4.0. If chosen, conditional text will only be viewable in this target type.

Note: The Help Target list contains the default Help targets. Any additional Help targets that the user defines will also appear in the list.

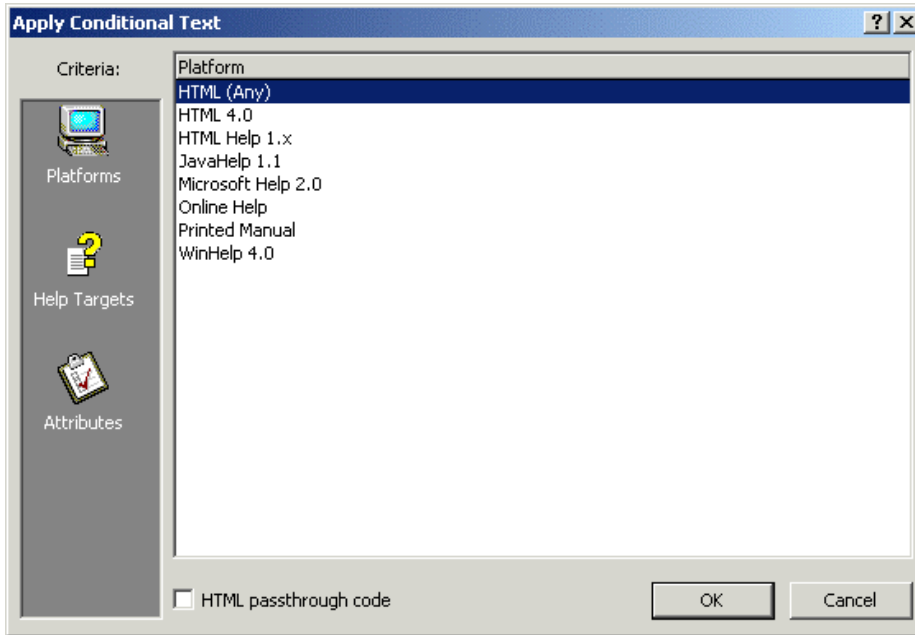
Attributes

The attribute Help target list is populated by attributes that you have created within your project. Attributes provide the Help author with the ability to create conditional text based upon specific Help files. For more information, see Using Attributes (page 305).

Adding Conditional Text Marks

- In Microsoft Word, select the text/image range to be converted into conditional text. Select the range as if you were going to copy it to the Clipboard.
- On the **Doc-To-Help** toolbar, click the **Conditional Text** button .

Doc-To-Help displays the following dialog box.



- In the left hand **Criteria:** toolbar of the **Apply Conditional Text** dialog box, choose the **Platforms**, **Help Targets** or **Attributes** icon as desired.
- From the list in the dialog box, select the item in which you want to include the highlighted text or image.
- Click **OK** to apply conditional text in the source document.
- Note that the selection now includes a comment after the chosen text as in the example below.

This sentence appears in online help targets only. [C:Text1]¶

You can make the text/image appear in multiple Help targets or attributes by simply performing the steps above for each choice you require.

Example

The following sentences are different for each Help target produced for Doc-To-Help.

This text appears in the Printed Manual Target.

In the source document, this was implemented with four separate paragraphs, each of which has a conditional text comment for one target. The following figure shows the appearance of the marked paragraphs in the source document.

This text appears in the WinHelp Target.[C:Text2]

This text appears in the HTMLHelp Target.[C:Text3]

This text appears in the Printed Manual Target.[C:Text4]

Alternatively, you can perform the same steps with the **Edit | Apply Conditional Text** menu command.

Removing Conditional Text Marks

In Microsoft Word, right-click the comment and select Delete Comment from the shortcut menu.

Editing Conditional Text Marks

In this version of Doc-To-Help, you cannot modify the conditional text by simply modifying the comment in Microsoft Word. Generally, it is easier to just remove the conditional text comment and add a new one as described earlier in this section.

Using the HTML Passthrough Code Option

The Doc-To-Help HTML passthrough feature allows you to include HTML code directly in your document without Word treating the code as text. To set HTML code as passthrough:

- In Microsoft Word, select the HTML code to be marked as if you were going to copy it to the Clipboard.
- On the **Doc-To-Help** toolbar, click the **Conditional Text** button.
- From the **Conditional Text** dialog box, select the HTML based platform of your choice and check the **HTML passthrough code** checkbox.

HTML passthrough code

Note: Doc-To-Help does not validate HTML passthrough code, so you should ensure that it works properly before incorporating it into your project.

A note to users of previous versions of Doc-To-Help:

It is no longer necessary to define your code with the HTML character style; any style may be used for your HTML code.

Inserting Files with HTML Passthrough Code

By using the HTML passthrough feature and the CopyFolder property you can insert files (like animated gifs or Excel spreadsheets) directly into your Help target. The CopyFolder property allows you to define a folder to store linked files when using HTML passthrough code.

To define a CopyFolder:

- Open the HTMLHelp folder that Doc-To-Help created when you first built your HTML Help target.
- Create a sub-folder within the HTMLHelp folder.

In this example, we will use **Media** as the name of the folder.

Note: In this example, we placed this CopyFolder within the HTMLHelp folder, but there is no limitation on the location of a CopyFolder. It is a good practice though, to keep the CopyFolder within the Doc-To-Help project folder or Help target folder.

- Save the files you want to insert using HTML passthrough to the media folder.
- From the Doc-To-Help project editor, select **Project** from the Icon Bar.
- Select the **Help Targets** item in the left hand pane.
- Select the **HTML Help** target from the right pane.
- From the Property Pane, select the CopyFolder property and type the following into the right cell:

```
. \HTMLHel p\Medi a
```

To set HTML code as passthrough:

- Select the location for your HTML code and type it directly into your source document.

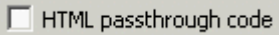
For Example:

```
<i mg src="MyAni matedGi f. gi f" >
<a href="MyExcel Spreadsheet. xl s">
```

Even though you have saved the file you want to insert to a sub-folder under your HTMLHelp folder, you do not need to include a path statement in your tag. This is because you set the CopyFolder property. The file will now be automatically be copied to the HTMLHelp Folder.

Note: Make sure that Word smart quotes are turned off so that straight quotes are included in the tags.

- Select the HTML code and from the Doc-To-Help toolbar, click the **Conditional Text** button.
- From the **Conditional Text** dialog box, select the HTML based platform of your choice and check the **HTML passthrough code** checkbox.



- Rebuild your HTML Help target and check your help file for the inserted file.

For more information, see [Using the HTML Passthrough Code Option](#) (page 303.)

Using Attributes

Attributes provide the Help author with the ability to create conditional text based upon author specified criteria. In some instances, you may wish to include specific text in one group of Help files while having alternate text included in a separate group of Help files. By creating attributes and associating the attribute values with Help targets, you can determine which targets contain which text.

For example, you might create an attribute grouping named Skills that contains the values Novice, Expert and Administrator. In this way, you can modify the text contained in Help files for each audience by simply attaching the value to a given group of Help targets and formatting the conditional text with those attributes values.

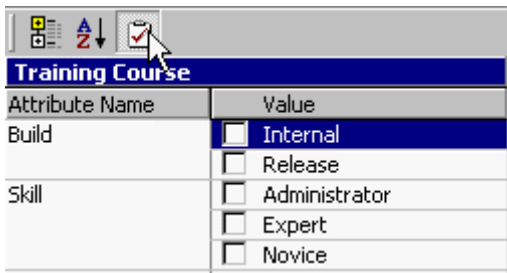
Start by adding a new attribute and values to your project.

Name	Value
(New)	
Build	Internal
Build	Release
Skill	Administrator
Skill	Expert
Skill	Novice

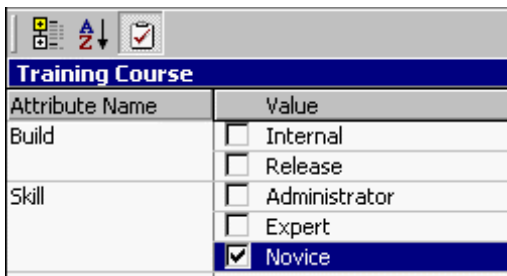
Assign attributes to text in your source document.

This sentence is not assigned an attribute. ¶
 This sentence is assigned the **Novice** attribute. [C.Text1] ¶
 This sentence is assigned the **Expert** attribute. [C.Text2] ¶
 This sentence is assigned the **Administrator** attribute. [C.Text3] ¶

In the project editor, select your Help target from the right pane and click the show attributes button in the property pane.



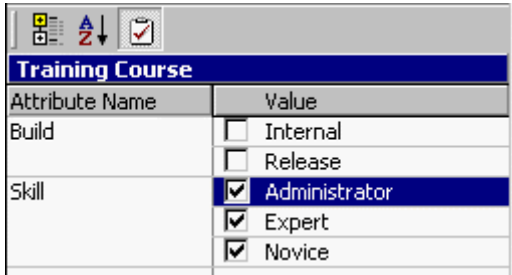
Apply the desired content to the Help target by selecting the attribute values.



Your selection produces the following text in your compiled Help file.

This sentence is not assigned an attribute.
 This sentence is assigned the **Novice** attribute.

By simply changing your choice of attribute values and recompiling, you can modify the contents of your Help file.



You new selection produces the following results in the compiled Help file.

This sentence is not assigned an attribute.

This sentence is assigned the **Novice** attribute.

This sentence is assigned the **Expert** attribute.

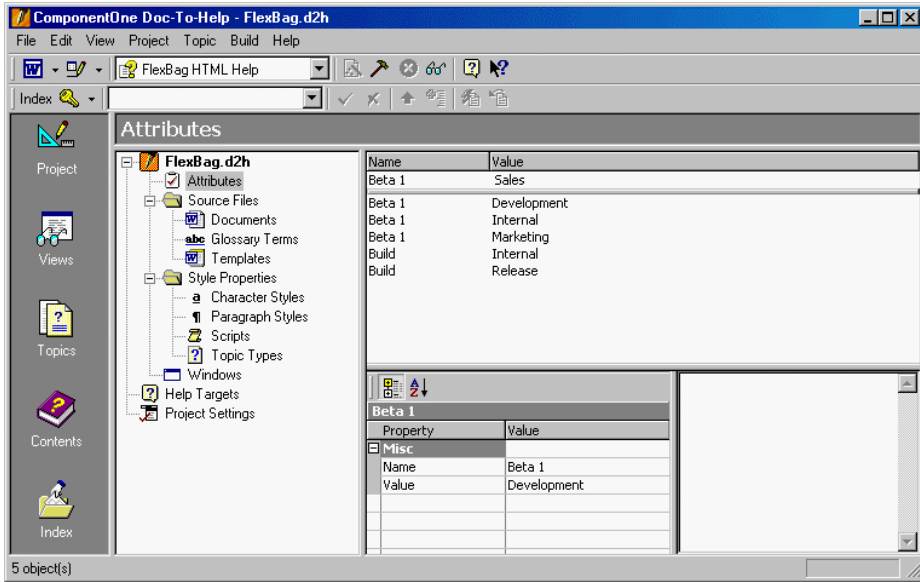
This sentence is assigned the **Administrator** attribute.

By highlighting the **Attributes** item on the project editor tree list, you are presented with a list of the default attributes in the right pane. By choosing any one of the default attributes, the lower left pane displays the property settings for that attribute.

Adding a New Attribute

To add a new attribute to the default list:

- Highlight the **Attributes** item on the topic icon project editor tree list.
- Right-click anywhere in the attributes list pane on the right and choose **New** from the shortcut menu.
- Note that the focus of your cursor has moved to the top of the attribute list window.
- Type the name of the new attribute and press **Enter**.
- Enter the value for the attribute and press **Enter**.
- Note that the new attribute has been added to the list.
- To add another value to the new attribute, simply perform steps 1 through 4 using the same attribute name but a new value. In this way you can have several attributes with the same name but different values.

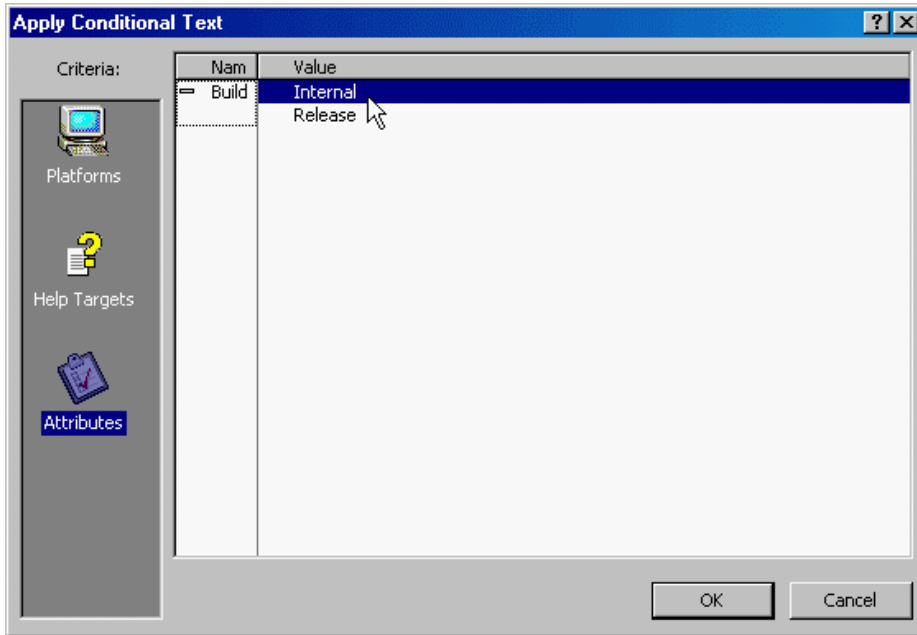


Alternatively, you can add a new attribute by clicking the **Project** menu, choosing **New**, then clicking **Attribute**.

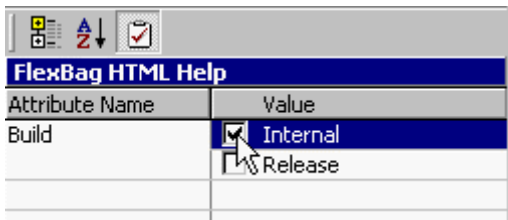
Inserting Conditional Text Using Attributes

In some cases you may have two Help files using the same Help target, but with somewhat different text. To do this, you can set the conditional text by attribute.

- Highlight the text that you wish to limit by attribute and click the **Apply Conditional Text** button.
- In the **Criteria:** pane, choose the **Attributes** icon.
- From the **Value** column, choose the attribute that you want to associate with the highlighted text.
- Click **OK**.




- Save and close the Word document.
- Select the **Help Targets** item in the left hand pane of the Project Editor.
- Choose the Help target you wish to assign with the attribute from the right pane.
- Click the **View Attribute** icon in the Help target properties pane.
- Choose the checkbox for the attribute value you want to associate with the Help target.

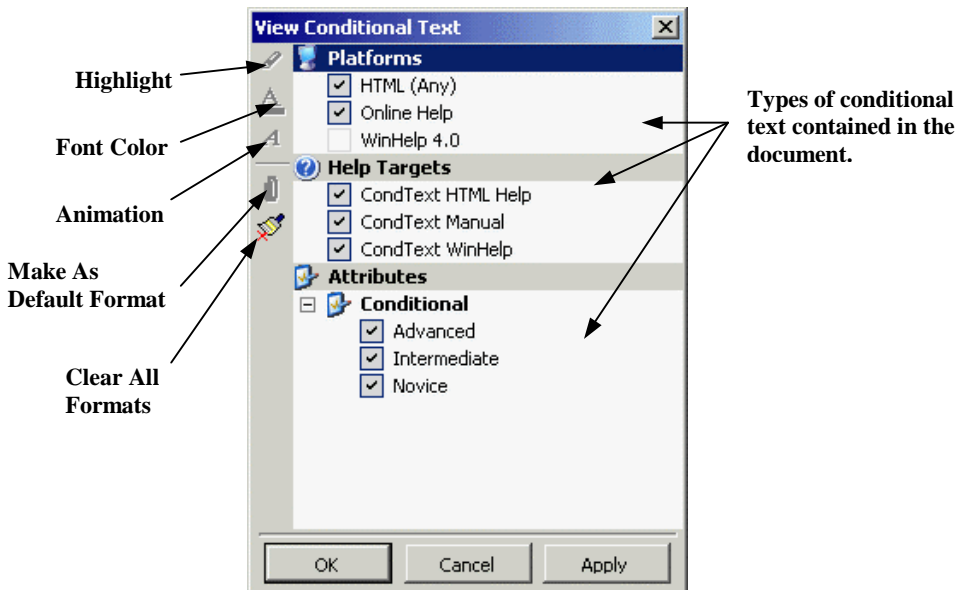


- Click the Make Target button to build the Help target.

Using the View Conditional Text Dialog

In some cases, you may wish to limit the types of conditional text you see in your source documents, or modify the text color of a specific type of conditional text. Doc-To-Help 7.5 provides you with a feature which allows you to determine which types of conditional text will be visible in your source documents. By simply clicking the **View Conditional**

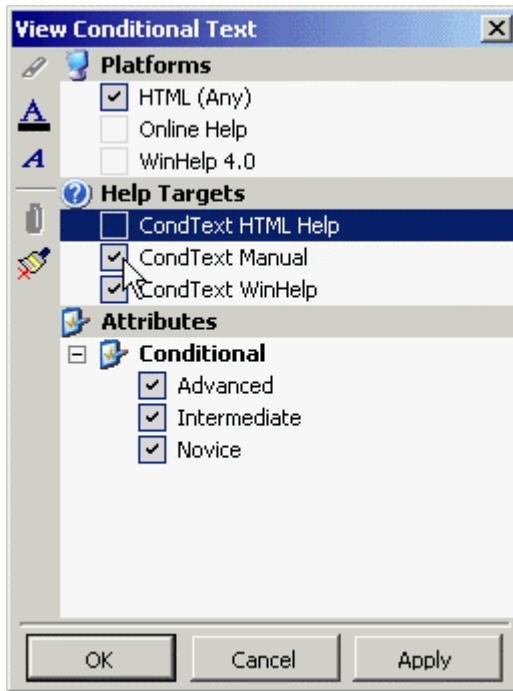
Text button , a dialog provides you with the ability to select specific conditional text for viewing, modify conditional text colors, add animation and set the default conditional text features. To close the View Conditional Text dialog, simply click the **OK** or **Cancel** buttons.



Note: Only checkboxes for the types of conditional text found within the current source document will appear in the View Conditional Text dialog.

Selecting Conditional Text Views

The View Conditional Text dialog allows you to select which types of conditional text you view in your source documents. This can make complex source documents, with many types of conditional text, easier to read by filtering out those types that you do not need to see. By clearing any of the Conditional Text checkboxes, you can filter which types of conditional text are shown in the currently opened document.



To filter the conditional text in a document:

- Open the document you want to filter.
- Click the View Conditional Text button.
- The View Conditional Text dialog opens. The dialog contains check boxes for each type of conditional text formatted within the document.
- Clear (or check) the check boxes for the conditional text you wish to filter.
- Click the **Apply** or **OK** button after you are done with your selections.

Doc-To-Help filters your conditional text so that only the types you selected are visible.

Note: To see the filtering, you must turn off your view hidden text option in Word.

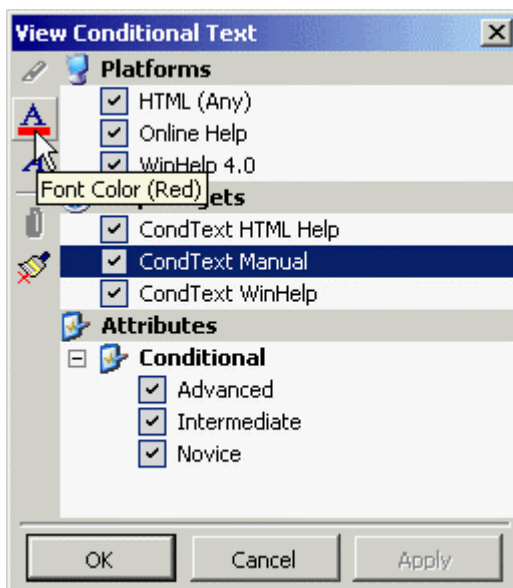
Customizing the Conditional Text Font

The default font color for all conditional text in Doc-To-Help is black. In some cases, where you have a wide range of conditional text types, it may be more convenient to define the text with different colors or even use animation. The View Conditional Text

dialog contained within Doc-To-Help uses features inherent in Microsoft Word to allow you to modify the text color and use animation to define various conditional text types.

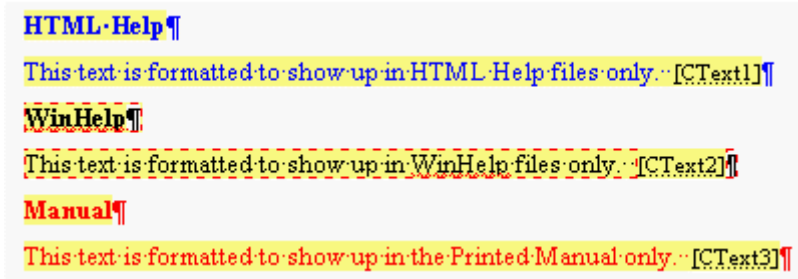
To modify the text definition for conditional text:

- Open the document you want to customize.
- Click the **View Conditional Text** button.
- The View Conditional Text dialog opens.
- Select the conditional text type you want to modify, then click the **Font Color** or **Animation** icon.



- Select the color or animation type you wish to apply from the appropriate dialog.
- Click the **Apply** or **OK** button after you are done with your selections.

In this way, you can modify the text color or animation of your conditional text as shown in the example below.

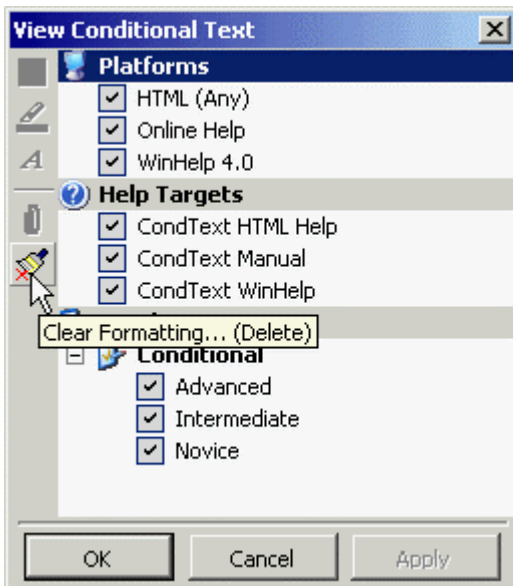


Note: Modifications to the conditional text affect the current file only. The font changes do not affect other source documents within the Help project.

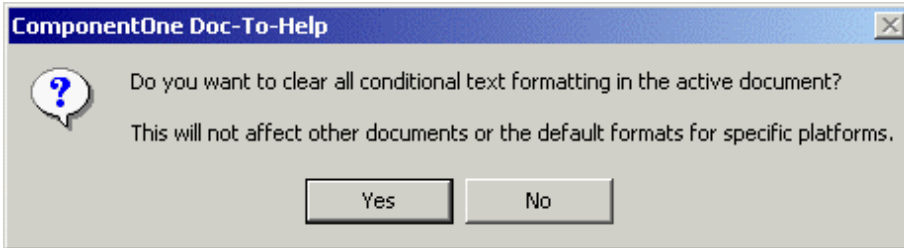
Clearing All Conditional Text Formats

In some cases you may wish to clear the formats you have made to the conditional text in your document. To do this, you simply need to click the **Clear All Formats** icon in the View Conditional Text dialog.

- Click the View Conditional Text button.
- The View Conditional Text dialog opens.
- Click the **Clear Formatting** icon.



- Click the **Yes** button to remove all conditional text formatting.

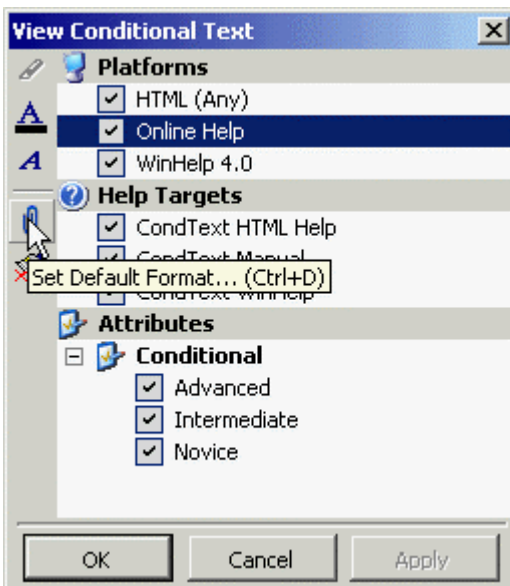


Note: Clearing modifications to the conditional text affect the current document only.

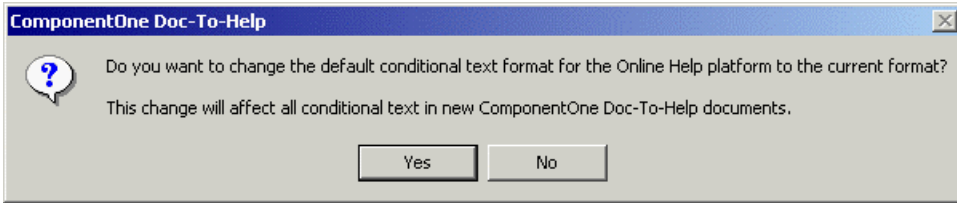
Modifying Conditional Text Default Format

The View Conditional Text dialog allows you to customize your conditional text platform styles then set them as the default for all future documents. After you have customized the conditional text for a specific platform, perform the following steps to set the customization as your default for that platform.

- Click the View Conditional Text button.
- The View Conditional Text dialog opens.
- Select the platform conditional text type you want to set as default, then click the **Set Default Format** icon.



- Click the Yes button to change the default conditional text for that platform.



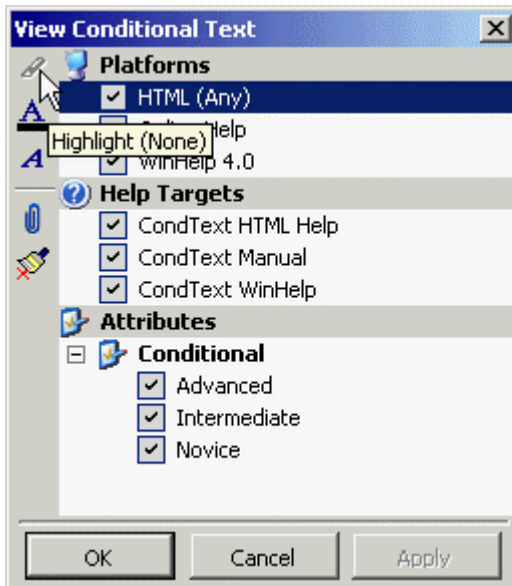
Note: Changes to the conditional text default settings are performed globally, affecting all future documents produced with Doc-To-Help. These changes will not affect the settings for existing documents including the document that is currently active.

Highlighting Conditional Text (Word XP)

If you are using Microsoft Windows XP to edit your source documents, you will notice that the comment text (and therefore Doc-To-Help conditional text) no longer utilizes the yellow background color. The View Conditional Text dialog allows you to insert a background color behind your conditional text as desired.

To add highlight to your conditional text:

- Open the document you want to customize.
- Click the View Conditional Text button.
The View Conditional Text dialog opens.
- Select the conditional text type you want to modify, then click the Highlight icon.



- Select which color you want as your background.
- Click the **Apply** or **OK** buttons after you are done with your selections.

Doc-To-Help inserts the background color into any conditional text of the type you selected.

ca to. This format is continuously helpful to readers.

This text will only appear in online help. |

Comment: Platform: Online H

This text will only appear in a printed manual. |

Comment: Platform: Printed M

Note: This feature is only functional with Microsoft Word XP.

Using Modular Help

Modular Help development breaks the creation of a large documentation project down into a set of smaller component pieces. Each component—containing, perhaps, one chapter of the overall project—becomes a separate document and Help file. The components are linked or merged together into a seamless single system, giving the end user appearance and functionality indistinguishable from a single large Help file.

In the context of ComponentOne Doc-To-Help 7.5, each component project in a modular Help system is a separate Doc-To-Help project, with its own folder, document file(s), and so forth, creating its own Help file. A separate hub project links the components together.

When you decide to take a modular approach with a Help system, you don't want to lose any functionality. You want to be able to generate the same high quality WinHelp and HTML Help systems as you would with a single project. Following the instructions in this chapter will ensure this. Before using modular Help, be aware that modular Help is Help target specific and you cannot produce a printed manual from a modular Help project.

Note: Do not confuse modular development with a Doc-To-Help multifile project. In a multifile project, multiple .DOC Word document files are compiled together into one Help file. In a modular project, there are different source documents, different Doc-To-Help projects, and different Help files, which are then made to appear to the user as one file.

Managing Your Modular Help System

Before beginning a modular Help project (or any project of a similar size and scope, for that matter), it's wise to take a step back and consider what the entire project will entail. With this perspective, you'll be better able to plan the project.

What's in a Modular Help System

Your modular Help system will consist of a number of modules or components, created from a number of Doc-To-Help projects. In addition to these component projects, the linking or merging of components in a modular Help project is done using what we shall call the hub project (generally one for each platform), which are primarily responsible for tying the modules together into an apparent whole.

Module Files:

- For WinHelp, each component is a separate WinHelp .HLP file and a WinHelp .CNT contents file. Each matching set of .HLP and .CNT files is the product of a separate Doc-To-Help project.
- For compiled HTML Help, each component is a separate HTML Help .CHM file. Just like with WinHelp, each .CHM file is the product of a separate Doc-To-Help project.

Hub Files

- For WinHelp, the hub is the WinHelp .HLP file and the WinHelp .CNT contents file.
- For compiled HTML Help, the hub is another compiled HTML Help .CHM file.

Note: When working with HTML Help hub files, there is a limitation to the naming convention that is unrelated to Doc-To-Help. Using spaces when naming the hub .CHM files will disable the next/previous functionality of the modular Help system. It is therefore advisable to eliminate spacing in your .CHM file names wherever possible. If your .D2H project file contains spaces (and, by default your .CHM file also contains spaces) simply remove the spaces from the BaseName property of the Help target.

In the examples in this documentation, we often describe the files as the “hub” project or “hub” Help file. You do not have to name your project “hub”, and in fact probably shouldn't. A better choice would be a descriptive name that reflects the content and purpose of your Help system.

Similarly, in the module examples, we describe files as “module” document or “module” Help file. Again, except for illustrative purposes in discussions of modular Help development, these are poor choices for file names—a descriptive and specific name is much better.

File Organization

There is no single right way to organize projects and files when you are developing modular Help systems. However, a good system of organization makes management easier and production less error-prone. The following suggestions are based on techniques and standards that have proven themselves again and again.

Create a folder for the development files for the entire system. Give each component project its own subfolder underneath the overall folder. The hub document should also have a separate subfolder.

If a team of several authors will be working on the system, you will probably want to put the project on a network drive. If you are working alone, you can put it on a local drive, but in either case, be sure it gets backed up regularly.

Your project will need a folder in which you can collect, collate, and test the total Help system. This folder needs only the files which will be distributed to end users, such as the .HLP, .CNT or .CHM files. It does not need the component project folders or their contents.

If end users on your network need to access your modular Help system, it's best to have another folder somewhere with the user-accessed copy of the Help. This allows you to develop and test the Help system, and your users to access the information they need, without interfering with one another.

Project Standards

In order for your modular project to mesh seamlessly, it's important that all the components adhere to a set of standards. Among the things you should consider standardizing are:

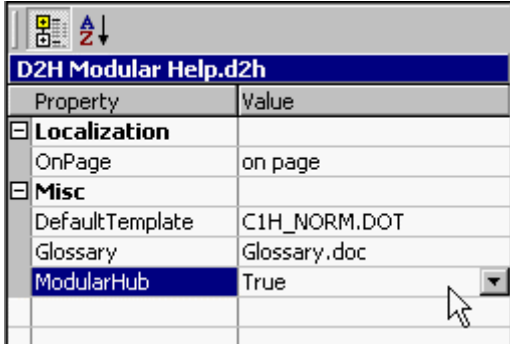
- Which templates your documents and help files are based on
- Which styles are used, for what, and how
- Style customizations
- Typographical conventions
- Help window designs and usage
- Indexing
- Frequency and type of hypertext jumps
- Use of graphics
- Tone and style of writing

You should also ensure that, if you are using a centralized glossary, authors do not attempt to add terms to their local glossary, but instead place their terms in the central glossary.

Creating a Modular Help Project for WinHelp

1. Start by creating your “module” Help projects as you normally would.
2. Create a separate project to use as the “hub” Help project.
3. Highlight the **Project Settings** item in the left pane of the project editor.

4. Set the ModularHub property to True.



5. Open the "hub" project document and create placeholder topics for the each of the "module" Help files you wish to associate with the hub.

The placeholder topic text is generally formatted as Heading 1 topics, but you can use other heading levels as desired.

6. Click the **Make Target** button to build the "hub" WinHelp file.
7. Click the **Topics** Icon.
8. Highlight one of the placeholder topics and enter the corresponding "module" hlp Help file to the right of the MergeFile property.
9. Enter the corresponding "module" .cnt Help file to the right of the MergeContents property.

Note that you can use the MergeTitle property to differentiate between like named Help topics.

The screenshot shows a window titled 'Xarray' with a table of properties. The table has two columns: 'Property' and 'Value'. The properties are grouped into sections: Behavior, Display, Misc, and Modular. The 'Modular' section is expanded, showing 'MergeContents' with value 'Xarray.cnt', 'MergeFile' with value 'Xarray.hlp', and 'MergeTitle' with an empty value.

Property	Value
Behavior	
CancelKey	False
CancelTag	False
TopicType	(Inherit from Style)
Display	
ContentsTitle	Xarray
DisplayTitle	Xarray
Misc	
Comments	
ID	5
Parent	0
Title	Xarray
Modular	
MergeContents	Xarray.cnt
MergeFile	Xarray.hlp
MergeTitle	

10. Build the “hub” project again.
11. Copy the “module” Help files (.hlp and .cnt) into the “hub” Help file output directory.
12. Click the “hub” Help file **View Target** button.

Note that the contents pane includes all the “hub” Help files.

13. Click the **Index** tab in the Help file.

Note that the index list has been merged to include index keywords from all the “module” Help files.

Note: A full rebuild (Rebuild Target command) deletes all existing files in the Help file output directory. If you do a full rebuild, be sure to copy the “module” Help files back into the “hub” output directory.

Modifying the Context IDs in your Modular Projects

To avoid numbering conflicts when using automatic context IDs, customize the context ID numbers in each “module” project with the MapNumberOffset property. For more information, see Context Sensitive Help (page 186) and Customizing your Automatically Generated Context ID Numbers (page 189).

WinHelp Module Glossaries

Doc-To-Help includes a glossary feature that automatically generates popup definitions throughout your Help, based on glossary definitions. If you would like to use this feature with modular Help, each component Help project (hub and module projects) will need to have a duplicate glossary.doc as part of the project. To eliminate the duplicate glossaries contained in the “module” Help projects, from the “hub” modular help table of contents, perform the following procedure on all **module** Help projects:

- Open one of the “module” projects.
- Select the **Project** icon.
- In the left pane, select **Help Targets**.
- From the right pane, select the Help Target you wish to modify.
- From the property pane, set the SkipGlossary property to **True** and press Enter.
- Click the **Make Target** button.
- Copy the compiled “module” Help files (.hlp and .cnt) into the “hub” Help file output directory, overwriting the previous version.
- Perform the same steps to remove the glossary contents from all of the “module” Help projects.

For more information, see *Modifying the Glossary* (page 266).

Referencing Absent CNT Files

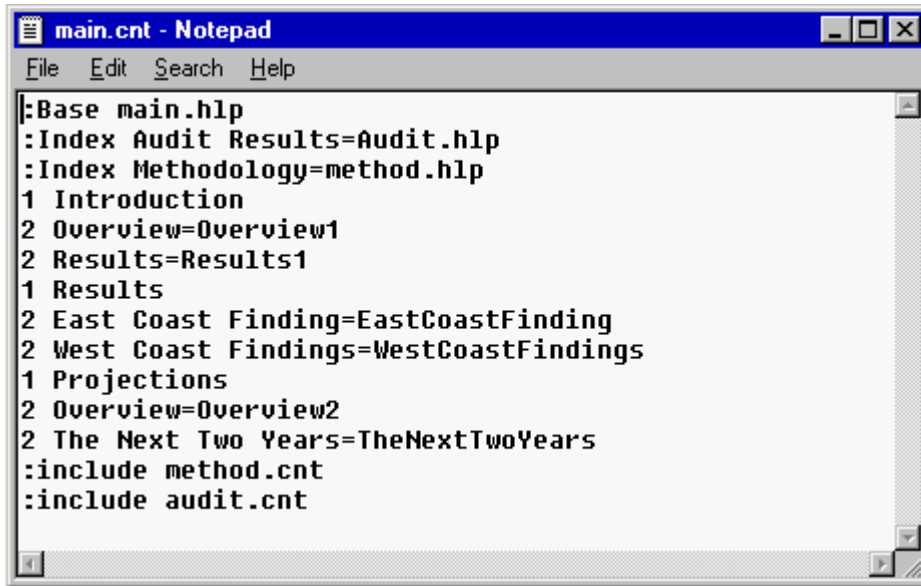
If your hub .CNT file contains references to “module” .CNT files that are not found on the user's computer (either in the current folder or in the WinHelp folder), don't worry, WinHelp won't display any error messages and everything will work fine. The headings and topics referenced by the missing .CNT files won't be displayed.

The same is true for files included in the index. If one of the Help files you specify is not found on the user's computer, its keywords are simply not included, without any errors displayed to the user.

This means you can include references to files that don't yet exist, but will exist when you send an updated version of your Help system or if someone purchases an additional module to your program. For example, let's say a user purchases a program from you. Your program's main .CNT file references ADDON.CNT which was not included with your program. Several months later, the user purchases the add-on module which comes with ADDON.CNT and ADDON.HLP. Assuming your add-on's setup program regenerates the main Help file, ADDON.CNT and ADDON.HLP will automatically become part of the Help system.

What's in a CNT File

A simple .CNT file that references other Help files and .CNT files is shown below.



```

main.cnt - Notepad
File Edit Search Help
:Base main.hlp
:Index Audit Results=Audit.hlp
:Index Methodology=method.hlp
1 Introduction
2 Overview=Overview1
2 Results=Results1
1 Results
2 East Coast Finding=EastCoastFinding
2 West Coast Findings=WestCoastFindings
1 Projections
2 Overview=Overview2
2 The Next Two Years=TheNextTwoYears
:include method.cnt
:include audit.cnt

```

The line

```
: Base mai n. hlp
```

indicates that MAIN.HLP is the main Help file. The lines

```
: I ndex Audi t Resul ts=audi t. hlp
: I ndex Methodol ogy=method. hlp
```

indicate that AUDIT.HLP and METHOD.HLP should be included in the keyword search, full-text search and in K-Links. The lines

```
1 I ntroducti on
2 Overvi ew=Overvi ew1
2 Resul ts=Resul ts1
1 Resul ts
. . .
2 The Next Two Years=TheNextTwoYears
```

determine the outline that appears in the Contents tab. The lines

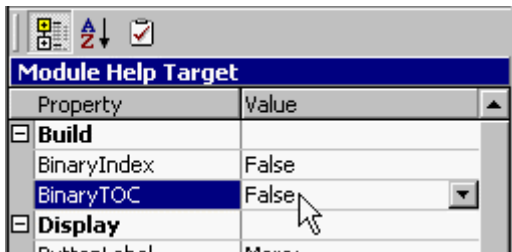
```
: i ncl ude method. cnt
: i ncl ude audi t. cnt
```

indicate that METHOD.CNT and AUDIT.CNT should be appended to the current outline.

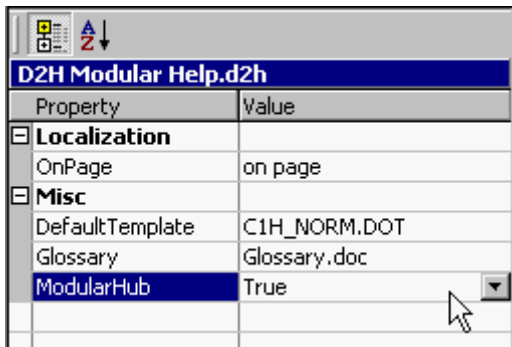
For a modular system, the component Contents files (generated automatically by Doc-To-Help during the Build Online Help process) contain the topic lines that determine the Contents tab outline. The hub Contents file contain :Include and :Index statements.

Creating a Modular Help Project for HTML Help

- Start by creating your “module” Help projects as you normally would.
- Set the BinaryTOC and the BinaryIndex properties of the “module” Help targets to False.



- Create a separate project to use as the “hub” Help project.
- Highlight the **Project Settings** item in the left hand pane of the project editor.
- Set the ModularHub property to True.

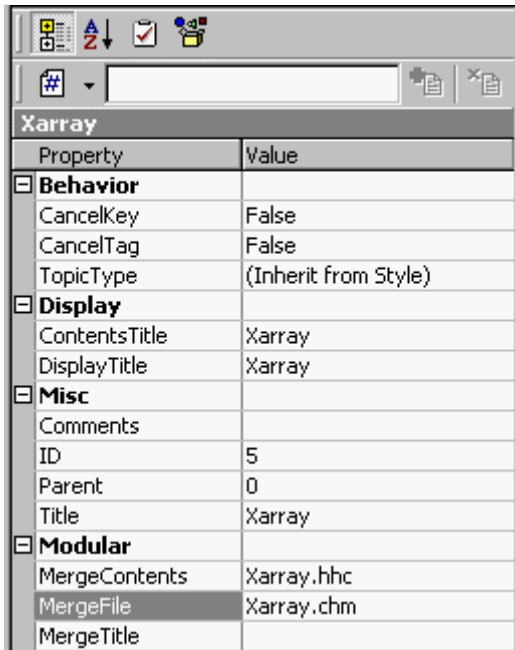


- Open the “hub” project document and create placeholder topics for the each of the “module” Help files you wish to associate with the hub.

The placeholder topic text is generally formatted as Heading 1 topics, but you can use other heading levels as desired.

- Click the **Make Target** button to build the “hub” HTML Help file.
- In the “hub” project, choose the **Help Target** item from the left pane and the **HTML Help** target from the right pane of the project editor.

- Set the BinaryTOC property to False and the BinaryIndex property to True.
- Click the project editor **Topics** Icon.
- Highlight one of the placeholder topics and enter the corresponding “module” Help file chm to the right of the MergeFile property. Enter the corresponding “module” Help file hhc to the right of the MergeContents property. (The hhc file name is normally the same as the chm file name.)



Property	Value
Behavior	
CancelKey	False
CancelTag	False
TopicType	(Inherit from Style)
Display	
ContentsTitle	Xarray
DisplayTitle	Xarray
Misc	
Comments	
ID	5
Parent	0
Title	Xarray
Modular	
MergeContents	Xarray.hhc
MergeFile	Xarray.chm
MergeTitle	

- Build the “hub” project again.
- Copy the “module” Help files (.chm) into the “hub” Help file output directory.
- Note that you do not have to copy the .hhc files into the “hub” Help file output directory.
- Click the “hub” Help file **View Target** button.

The contents pane includes all the “hub” Help files. Click the **Index** tab; notice that the index list has been merged to include index keywords from all the “module” Help files.

Note: A full rebuild (Rebuild Target command) overwrites all existing files in the Help file output directory. If you do a full rebuild, be sure to copy the “module” Help files back into the “hub” output directory.

Modifying the Context IDs in your Modular Projects

To avoid numbering conflicts when using automatic context IDs, customize the context ID numbers in each “module” project with the `MapNumberOffset` property. For more information, see [Context Sensitive Help](#) (page 186) and [Customizing your Automatically Generated Context ID Numbers](#) (page 189).

HTML Help Module Glossaries

Doc-To-Help includes a glossary feature that automatically generates popup definitions throughout your Help, based on glossary definitions. If you would like to use this feature with modular Help, each component Help project (hub and module projects) will need to have a duplicate `glossary.doc` as part of the project. To eliminate the duplicate glossaries contained in the “module” Help projects, from the “hub” modular help table of contents, perform the following procedure on all **module** Help projects:

Open one of the “module” projects.

1. Select the **Project** icon.
2. In the left pane, select **Help Targets**.
3. From the right pane, select the Help Target you wish to modify.
4. From the property pane, set the `SkipGlossary` property to **True** and press Enter.
5. Click the **Make Target** button.
6. Copy the compiled “module” Help file (.chm) into the “hub” Help file output directory, overwriting the previous version.
7. Perform the same steps to remove the glossary contents from all of the “module” Help projects.

For more information, see [Modifying the Glossary](#) (page 266).

HTML Help Hub Project

The hub project in a modular HTML Help system is a Doc-To-Help project resulting in a compiled HTML Help (.CHM) file that ties the whole system together through special entries in the contents (.HHC) and project (.HHP) files.

We recommend that you keep an HTML Help hub project quite small and restrict it to information that does not change often, such as a welcome screen, directions for using the Help, or company contact information.

Referencing Module Files

A modular HTML Help system's hub file uses two mechanisms to reference the components: Merge tags in the Contents (.HHC) file to combine the components' tables of contents with that of the hub; and [Merge Files] entries in the project (.HHP) file to include the components in the hub HTML Help file's Index and Find features.

Merge Tags

A typical Merge tag looks like this:

```
<OBJECT type="text/si temap">  
<param name="Merge" value="File.chm: /File.hhc">  
</OBJECT>
```

HTML Help Contents files can be merged even if the HTML Help files are not compiled .CHM files.

[MERGE FILES]

Syntax of the [MERGE FILES] section is very simple; just list the .CHM files, as shown here:

```
[MERGE FILES]  
Al pha.chm  
Bravo.chm  
Charl ie.chm
```

Support for the [MERGE FILES] section is limited to compiled HTML Help (.CHM) files.

Scripting Techniques

This chapter describes common operations necessary to write code modules using VBScript that automate indexing and hyperlinks in ComponentOne Doc-To-Help 7.5 projects.

For complete VBScript documentation, visit the Microsoft Scripting Technologies site at <http://msdn.microsoft.com/scripting>.

What are Scripts?

Doc-To-Help scripts are code modules written in the VBScript. They can be used to modify the behavior of paragraph and character styles during compilation. If you are familiar with event-driven languages such as Visual Basic, you can think of a script as an "event handler" for a style. Just as the Click event is fired when the user clicks a button in a Visual Basic application, a script is executed whenever Doc-To-Help encounters a topic or a hot spot defined by a scripted style.

You can perform the following operations in script code:

- Create top-level and secondary index keywords.
- Create named topic groups.
- Assign index keywords and groups to topics.
- Assign context-sensitive Help IDs to topics.
- Modify the link tags of topics and hot spots.

You can perform all of these operations in Doc-To-Help's authoring environment using techniques outlined elsewhere. If the notion of writing code seems daunting, then you are not required to use scripts at all. However, a simple script can save you many hours of tedious labor. This is especially true in cases where the title of a topic or the text of a hot spot can be split apart or rearranged to yield a textual match or a different focal point for an index entry. For example, given a set of topic titles of the form *Adjective Noun*, you could write a script to parse the title and associate three index keywords with each topic:

```
Adj ecti ve noun
nouns
nouns, Adj ecti ve
```

Note that you can control both the capitalization and the number (singular or plural) to conform to standard indexing conventions. You can also employ secondary keywords to improve the readability of the index:

```
Adj ecti ve noun  
nouns  
  Adj ecti ve
```

Suppose further that the topics are organized into several documents, with each document describing the features of a particular *Widget*. With just a few more lines of code, you could extend the index profile for each topic as follows:

```
Adj ecti ve noun  
Adj ecti ve noun, of Wi dget  
nouns  
nouns, Adj ecti ve  
nouns, of Wi dget  
Wi dget  
Wi dget, Adj ecti ve noun  
Wi dget, nouns of
```

If secondary keywords are used, the reader sees:

```
Adj ecti ve noun  
  of Wi dget  
nouns  
  Adj ecti ve  
  of Wi dget  
Wi dget  
  Adj ecti ve noun  
  nouns of
```

If you have to document hundreds of nouns for dozens of widgets, then maybe writing a script isn't such a crazy idea after all!

Scripts are Subroutines

When you create a script named *xyz*, Doc-To-Help automatically creates a subroutine declaration for it, resulting in the following VBScript code:

```
Sub xyz()  
' Body of xyz scri pt  
End Sub
```

This has the following implications:

- Script names must be legal procedure names in VBScript. That is, they must start with a letter and contain only letters, numbers, and underscores.
- You cannot define other subroutines or functions within the body of a Doc-To-Help non-global script, as VBScript does not support nested procedures.

```
Func ti on pi () ' Not a val id Doc-To-Hel p scri pt  
pi = 3. 14159  
End Func ti on
```

- You can use the `Exit Sub` statement to exit a script.

```
If expression Then
    MsgBox "Exiting script"
    Exit Sub
End If
```

- You can use the `Call` statement to execute another script.

```
Call xyz
xyz ' Call keyword is optional
```

Unlike other Doc-To-Help objects such as styles, scripts have no properties that you can set in the authoring environment (other than the code itself)

Note: There is one exception to the rule that script names must be legal VBScript procedure names. If a script name starts with a character that is not a letter, Doc-To-Help treats it as a global module. This means that it can contain subroutines and functions that can be called from other scripts. For more information, see [Creating a Global Script Module \(page 333\)](#).

Scripts Access the Doc-To-Help Object Model

Since scripts are subroutines without arguments, your code must communicate with the Doc-To-Help objects. Whenever a script runs, there is always one default global object named **Project**. The term *global* means that the object is accessible to all scripts. The term *default* means that you do not have to refer to it explicitly by name. That is, the following statements are equivalent:

```
Project.Print "The current Help target is " &
Project.Platform
Print "The current Help target is " & Target
```

The **Project** object has three collections that represent the individual items in a project. The most important of these are the following:

Topics	All topics in the project.
Keywords	All top-level index keywords in the project.
Groups	All named topic groups in the project.

In addition to its collections, the **Project** object also has two fundamental properties, each of which returns an object that is central to the compilation process:

ActiveTopic	Returns a Topic object representing paragraph style formatting. This property can be used only in scripts assigned to a paragraph style.
--------------------	---

ActiveLink	Returns a Link object representing character style formatting. This property can be used only in scripts assigned to a character style.
-------------------	--

Scripts are Extensions of Styles

In order for a script to be executed during compilation, you must assign it to a style. Scripts assigned to paragraph styles handle topics; scripts assigned to character styles handle topic references or hot spots.

If you assign an empty script to a style, the behavior of the style during compilation is the same. The behavior of your style only changes if the code or properties of the Help system is changed.

For both paragraph and character styles, you can use the following properties to override or cancel the default behavior as specified in the **Styles** tab:

Tag	Sets or returns the default link tag.
Key	Sets or returns the default index keyword.
Cancel	Cancels both the default link tag and the default index keyword when set to <i>True</i> .

These three properties apply to the **Topic** object returned by the **ActiveTopic** property as well as the **Link** object returned by the **ActiveLink** property.

Working with Scripts

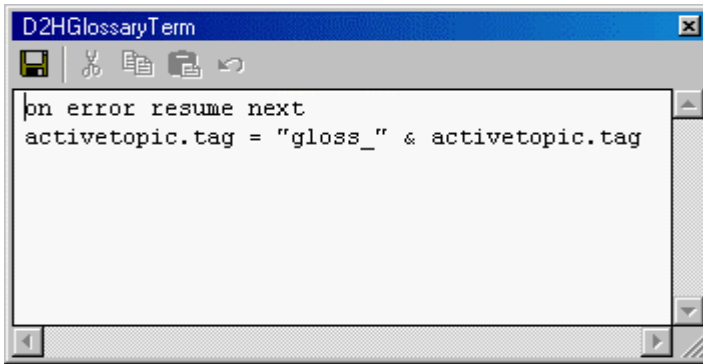
This section describes how to create, edit, run, and debug scripts.

Creating a New Script

To create a new script:

1. Highlight the **Script** item on the project editor tree list.
2. Right-click anywhere in the script list pane on the right and choose **New** from the shortcut menu.
3. Note that the focus of your cursor has moved to the top of the script list window.
4. Type the name of the new script and press **Enter**.
5. Note that the new script name has been added to the list.
6. To activate the editable code textbox, highlight the new script name in the **Upper Right Pane**, then click the box located to the right of the Code

property box in the **Properties Pane**. In this textbox you can enter a new script code or edit an existing one.



Alternatively, you can add a new script by clicking the **Project** menu, choosing **New**, then clicking **Script**.

Creating a Global Script Module

If you have multiple scripts that perform similar tasks, you can simplify your code by moving the common statements to a separate procedure, then calling the procedure directly from the original scripts. Normally, scripts cannot contain subroutines or functions, but if a script is designated as a global script it can contain them.

You can designate a script as a global module by choosing a name that does not begin with a letter, such as "(General)". For more specific instructions, see [Creating a New Script](#) (page 332.)

Since a script specified as a global script is treated as a repository for multiple `Sub` and `Function` declarations, Doc-To-Help will not wrap the body of the script with `Sub` and `End Sub` statements.

You cannot assign a global script module to a style.

Editing Scripts

Highlight the **Script** item on the project editor tree list.

1. Select the name of the script to be edited from the right pane.
2. To activate the editable code textbox, click the box located to the right of the Code property box. Edit the code within the text box as desired.
3. Note that your changes affect the selected script instantly and need not be saved explicitly. However, the full effect of your changes is not realized until you build the Help target.

For details on how to write scripts, see [Common Script Operations](#) (page 335).

Assigning a Script to a Style

In order for a script to be executed during compilation, you must assign it to a style as follows:

1. Highlight the **Paragraph Styles** or **Character Styles** item on the project editor tree list.
2. Select the name of the style to be edited from the right pane.
3. In the properties pane for the style, click the box to the right of the Script property and choose the appropriate script from the drop-down.

Note: You can assign one script to multiple styles. A script must be created in the project before it can be assigned to a style.

Running a Standalone Script

- In the left hand pane of the project editor, highlight the **Scripts** item.
- In the right hand pane of the project editor, right-click on the script you wish to run and select **Run** from the shortcut menu.

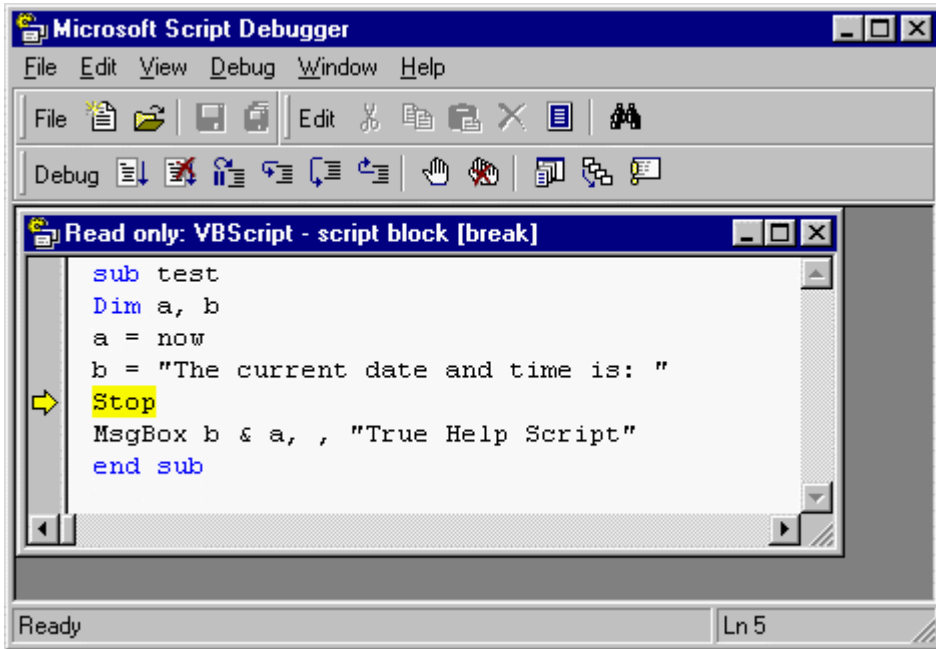
Debugging a Script

The following programming constructs are always available for displaying intermediate results or interacting with a running script:

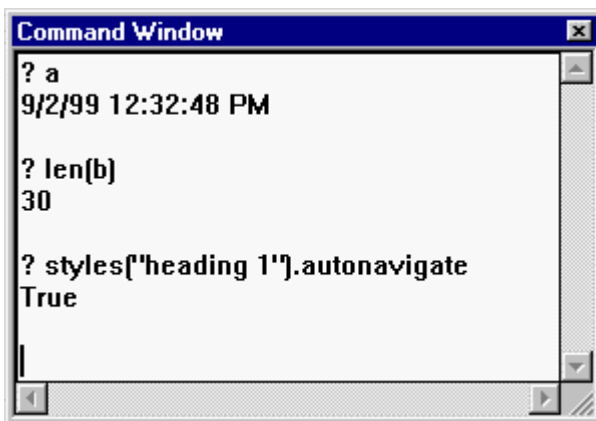
Print	This Doc-To-Help method writes the value of a variable or constant to the build output window.
MsgBox	This VBScript function displays a message in a modal dialog box and optionally prompts for a response using command buttons.
InputBox	This VBScript function displays a message in a modal dialog box and prompts for a character string.

If you have the Microsoft Script Debugger installed, you can also use the `Stop` statement to halt execution and open a separate window for interactive debugging.

Doc-To-Help setup does not install the Microsoft Script Debugger, but you can download a free copy from the Microsoft Web site at <http://msdn.microsoft.com/scripting>.



You can use the debugger's command window to display the variable values, execute intrinsic VBScript functions, or examine the object model of Doc-To-Help.



Common Script Operations

This section demonstrates some typical uses of scripts, particularly for indexing, and provides tips for handling errors.

Changing automatic link tags in a script

In order for a topic to be a destination for a jump or popup link, it must have a unique link tag that identifies it. Typically, you do this by enabling automatic link tags for an active paragraph style. For specific instructions, see [Adding a Link Tag to a Topic](#) (page 178).

Automatic link tags are based on the topic title, but are normalized so that they do not contain any illegal characters (as restricted by the Windows Help compiler):

- Spaces, hyphens, and periods are converted to underscores.
- Letters, numbers, and underscores are unchanged.
- All other characters are removed.

When a script assigned to a paragraph style executes at compile time, the following expression returns the normalized link tag generated by Doc-To-Help:

```
ActiveTopic.Tag
```

However, you can change the default behavior by setting the **Tag** property to a different value. For instance, if you have several documents with like-named topics, you can generate a unique link tag for each topic by appending the document name:

```
Dim name, pos
name = ActiveTopic.Document.Name
pos = InStr(1, name, ".")
ActiveTopic.Tag = ActiveTopic.Tag + "_" + Left(name, pos - 1)
```

Note the use of the intrinsic VBScript functions `InStr` and `Left` to extract the root filename, minus the extension. The variable `pos` receives the character position of the period in the filename, and the `Left` function returns a string containing all characters to the left of the period.

To prevent a link tag from being generated, set the **Tag** property to an empty string:

```
ActiveTopic.Tag = ""
```

For more information, see [Linking Related Topics](#) (page 30).

Changing automatic index keywords in a script

For reference material, the topic title often serves as a convenient index. Typically, you index reference topics by enabling automatic index keywords for an active paragraph style. For specific instructions, see [Creating Index Keywords Automatically Using Styles](#) (page 221).

When a script assigned to a paragraph style executes at compile time, the following expression returns the name of the automatic index keyword generated by Doc-To-Help:

ActiveTopic.Key

However, you can change the default behavior by setting the **Key** property to a different value. For instance, you can convert the keyword to lower case with the following statement, which uses the intrinsic VBScript function `LCase`:

```
ActiveTopic.Key = LCase(ActiveTopic.Key)
```

To prevent an index keyword from being generated, set the **Key** property to an empty string:

```
ActiveTopic.Key = ""
```

Creating index keywords in a script

To create top-level index keywords, use the **Add** method of the **Keywords** collection:

```
Keywords.Add "animal"  
Keywords.Add "vegetable"  
Keywords.Add "mireal"
```

Keywords created in this manner are not associated with any topics. This is analogous to creating top-level keywords in the **Index** toolbar.

However, if you use the **Keywords** collection of the active topic, and not of the project, then the keywords are created and associated with the active topic:

```
ActiveTopic.Keywords.Add "animal"  
ActiveTopic.Keywords.Add "vegetable"  
ActiveTopic.Keywords.Add "mireal"
```

This is analogous to entering keywords in the **Index** tab of the topic properties pane. Note that you can simplify the script and make it more efficient by assigning the **Keywords** collection of the **ActiveTopic** object to a variable:

```
Dim keys  
Set keys = ActiveTopic.Keywords  
keys.Add "animal"  
keys.Add "vegetable"  
keys.Add "mireal"
```

Creating secondary index keywords in a script

To create secondary index keywords, use the **Add** method of the **Keywords** collection and specify the parent keyword as the second argument. The following example creates a top-level keyword and then creates a secondary keyword as its child:

```
Dim key  
Set key = Keywords.Add("mammals")  
Keywords.Add "Hominid", key
```

Note the use of the variable `key` as a placeholder for the top-level **Keyword** object. Keywords created in this manner are not associated with any topics. This is analogous to creating secondary keywords in the **Index** toolbar.

However, if you use the **Keywords** collection of the active topic, and not of the project, then both top-level and secondary keywords are created and associated with the active topic:

```
Dim key
Set key = ActiveTopic.Keywords.Add("mammals")
ActiveTopic.Keywords.Add "Homo sapiens", key
```

This is analogous to using the **Index** command of the **Topic** menu twice, once for each keyword. Note that you can simplify the script and make it more efficient by assigning the **Keywords** collection of the **ActiveTopic** object to a variable:

```
Dim key, keys
Set keys = ActiveTopic.Keywords
Set key = keys.add("mammals")
keys.add "Homo sapiens", key
```

Creating topic groups in a script

To create named groups, use the **Add** method of the **Groups** collection:

```
Groups.Add "novice"
Groups.Add "intermediate"
Groups.Add "expert"
```

Groups created in this manner are not associated with any topics. This is analogous to creating groups in the **Index** toolbar.

However, if you use the **Groups** collection of the active topic, and not of the project, then the groups are created and associated with the active topic:

```
ActiveTopic.Groups.Add "novice"
ActiveTopic.Groups.Add "intermediate"
ActiveTopic.Groups.Add "expert"
```

This is analogous to entering groups in the **Index** tab of the topic properties pane. Note that you can simplify the script and make it more efficient by assigning the **Groups** collection of the **ActiveTopic** object to a variable:

```
Dim grps
Set grps = ActiveTopic.Groups
grps.Add "novice"
grps.Add "intermediate"
grps.Add "expert"
```

Handling error conditions in a script

When writing scripts, you need to be aware that some operations may fail. Fortunately, you can use the built-in error handling features of VBScript to take the appropriate action.

When working with collections, a common error is attempting to access an element that does not exist. Consider the following standalone script, which tries to assign a bogus keyword to a variable:

```
Dim k
Set k = Keywords("@#!%$")
Print TypeName(k)
```

Running this script produces the following error message:

```
Script error on line 2, column 1:
Keyword not found: '@#!%$'
```

Since all script errors are fatal, the `Print` statement never executes. However, if an `On Error` statement is inserted before line 2, the script executes in its entirety:

```
Dim k
On Error Resume Next
Set k = Keywords("@#!%$")
Print TypeName(k)
```

This time, the script prints the word `Empty` to the output window, since the variable `k` is still uninitialized. Technically, the `Set` statement still results in an error, but the `On Error` directive causes execution to continue with the statement immediately following the one that caused the error.

Another common error is attempting to add an element to a collection that already exists. Consider the following standalone script, which adds a new group to the project:

```
Groups.Add "qwerty"
```

The first time this script is run, it executes silently. However, running it again produces the following error message:

```
Script error on line 1, column 1:
Group already exists: 'qwerty'
```

Again, the solution is to use the `On Error` statement:

```
On Error Resume Next
Groups.Add "qwerty"
```

VBScript also provides an `Err` object that you can use to determine whether the last statement was successful. The following example demonstrates how to use the `Err` object to write code that always operates on a specified member of a collection, whether or not it already exists:

```
Dim k
On Error Resume Next
Set k = Keywords("primary")
If Err <> 0 Then ' keyword not found
    Set k = Keywords.Add("primary")
    Err.Clear
End If
' Add a secondary keyword under "primary"
Keywords.Add "secondary", k
```

The `Set` statement in the third line attempts to access a specific keyword. If it exists, the default property of the `Err` object returns zero, and execution continues with the last line of the script. If the keyword does not exist, the script creates it using the `Add` method. In either case, when the last line of the script executes, the variable `k` contains the appropriate object.

Finally, note that the scope of the `On Error` statement is limited to the procedure in which it appears. Consider the following function, which resides in a global script module:

```
Function GetKeyword(Name)
    On Error Resume Next
    Set GetKeyword = Keywords(Name)
    If Err <> 0 Then ' keyword not found
        Set GetKeyword = Keywords.Add(Name)
        Err.Clear
    End If
End Function
```

The `GetKeyword` function generalizes the technique employed in the previous example. It accepts a string and unconditionally returns a top-level keyword having the same name. If necessary, the function creates a new keyword; otherwise, it returns an existing one. The following example demonstrates how to call `GetKeyword` from another script:

```
Dim k
Set k = GetKeyword("primary")
Keywords.Add "secondary", k
```

If the specified secondary keyword already exists, the third line results in an unhandled error, even though the `GetKeyword` function uses the `On Error` statement. In order to trap all errors, an additional `On Error` statement is required:

```
Dim k
Set k = GetKeyword("primary")
On Error Resume Next
Keywords.Add "secondary", k
```

Documenter for .NET



Documenter for .NET creates and maintains documentation for one or more .NET assemblies. All you have to do is select the assemblies you want to document and **Documenter** will automatically generate MSDN-style reference documentation, complete with formatting and references. Additionally, Documenter for .NET generates a Doc-To-Help project with the reference document. This project can be used to build any Help target supported by Doc-To-Help.

Documenter uses reflection to collect information about the classes in your assemblies. Because the information is generated directly from the assemblies, it is always accurate and up-to-date. *Even without source code* comments, Documenter builds a complete reference document with all types and members present.

Documenter can also use Xml documentation files created by the C# compiler. The XML documentation is created automatically by the compiler using comments embedded in the source code. Because this type of documentation is written along with the code, it is easy to keep up-to-date. Also, the compiler can check any cross-references automatically and spot documentation errors at compile time. For details on Xml documentation, please refer to MSDN (see the XML Documentation Tutorial).

Documenter is fully integrated with **Doc-To-Help**. Because of this, you can leverage all the power of **Doc-To-Help** to enhance the automatically generated documentation. You can, add custom narrative sections, cross-references, indices, Help targets, styles, and more.

Documenter for .NET is available in the Professional version only.

Generation Speed and Supported Word Versions

Generation speed in Documenter for .NET is dramatically optimized - up to 20 times faster - with Word 2003. The previous Word versions, Word 2000 and Word XP, are also fully supported with exactly the same document generated. However, the generated document is slower with old Word versions. This dramatic speed enhancement in Word 2003 is due to using the new XML features (internally in Documenter and therefore transparent to the user) that are only available starting with Word 2003.

Please note that only Generate New Document is optimized with using Word 2003 XML features. Regenerate Existing Document has the same speed with Word 2003 as with

earlier Word versions. However, Regenerate Existing Document is usually very fast as it is, without additional optimization.

Documenter for .NET Guided Tour

This section walks you through the process of documenting an existing assembly. The components for this tutorial are located in *C:\Program Files\ComponentOne\DocToHelp7\Tutorial\BarCode*.

\BarCode Assembly\Original	Contains the .DLL and .XML files used for the original build of the tutorial project.
\BarCode Assembly\Updated	Contains the .DLL and .XML files used for the update build of the tutorial project.
\BarCode Assembly\Document	Contains the CustomTopics.doc used to add additional an additional document during the tutorial.
\BarCode Project	Contains the completed Documenter project example.

Note: The Documenter for .NET Guided Tour assumes familiarity with .NET programming.

Selecting the Assemblies to Document

Start by selecting the assembly (or assemblies) you want to document.

Even without source code comments, Documenter produces complete reference document with all types and members present. Brief descriptions are included if they are supplied in the Description Attribute in the assembly.

If the assembly contains Xml documentation comments, make sure the corresponding Xml documentation file is up-to-date by rebuilding the assembly.

For this tutorial, we will use the **C1.Win.C1BarCode.Dll** assembly, a simple .NET assembly that contains a single WinForms control. The assembly is located in *C:\Program Files\ComponentOne\DocToHelp7\Tutorial\BarCode\BarCode Assembly*.

Using XML Comments

Xml comments are optional, but they are a great feature when used with **Documenter**. They are easy to add and maintain, and the compiler helps ensure the documentation is

complete and accurate. Xml comments encourage you to write and update the documentation in a consistent manner, while you are writing or modifying the code.

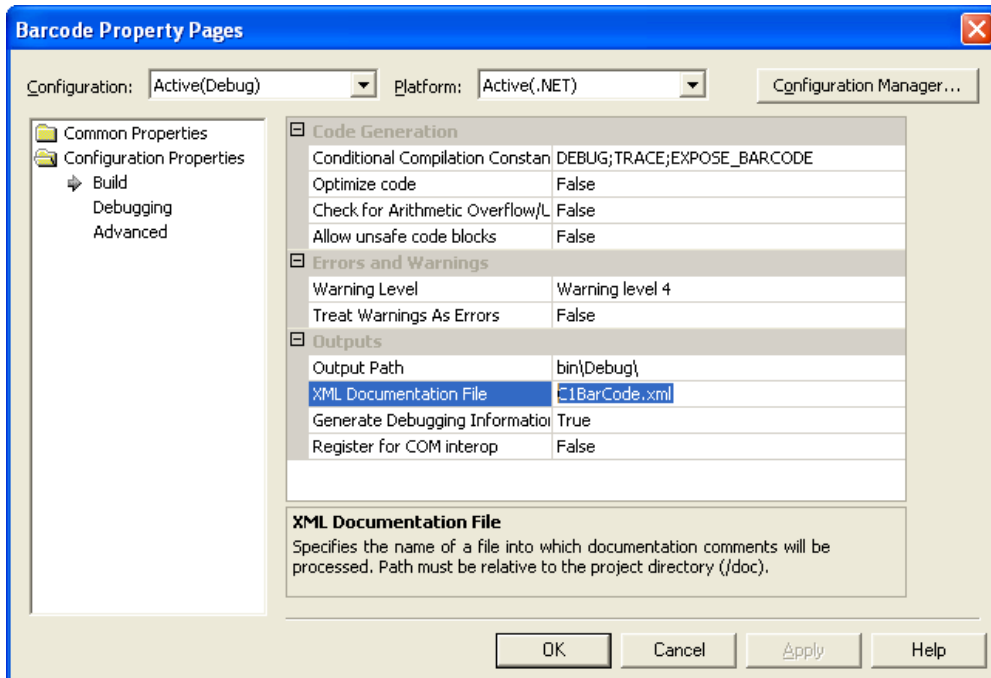
The **C1Barcode** assembly contains Xml comments, as shown in the code snippets below:

```
namespace Cl.Win.C1Barcode
{
    /// <summary>
    /// Specifies the type of barcode to be generated by the C1Barcode control.
    /// </summary>
    /// <remarks>
    /// The CodeTypeEnum enumeration represents the different types of barcode that
    /// can
    /// be generated by the C1Barcode control. Some types can be used to represent
    /// alphanumeric values, others can only represent numeric values.
    /// </remarks>
    public enum CodeTypeEnum
    {
        /// <summary>
        /// Code 39 is an alpha-numeric encoding also known as 3 of 9 and LOGMARS.
        /// This was the first alphanumeric symbology developed, and is one of the
        /// most widely used encodings.
        /// </summary>
        Code39,

        /// <summary>
        /// Code 93 is an alpha-numeric encoding that is slightly denser than code 39.
        /// </summary>
        Code93,
        ...
    }
}
```

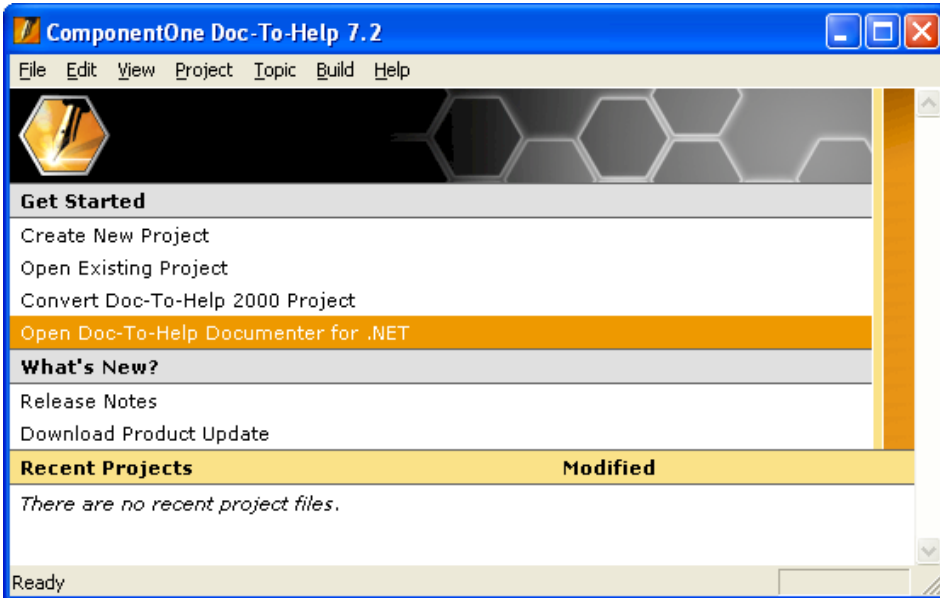
```
/// <summary>
/// Gets or sets the value that is encoded as a barcode image.
/// </summary>
/// <remarks>
/// <para>Some encodings support alpha-numeric values of any length. Others
/// are limited to numeric values or to a specific set of characters.
/// See the <see cref="Cl.Win.ClBarcode.CodeTypeEnum"/> enumeration for
/// details. </para>
/// <para>If you use characters that are invalid for the current encoding, the
/// control will display a blank image.</para>
/// </remarks>
/// <example>This sample shows how to use the Text property to encode
/// the value "123456" as a Code39-encoded barcode.
/// <code>
///     clbarcode1.CodeType = CodeTypeEnum.Code39;
///     clbarcode1.Text = "123456";
///     pictureBox1.Image = clbarcode1.Image;
/// </code>
/// </example>
[
Description("Gets or sets the value that is encoded as a barcode image."),
DefaultValue("")
]
override public string Text
{
    get { return base.Text; }
    set
    {
        base.Text = value;
        SetDirty();
    }
}
```

To generate the XML documentation file, configure your project as shown in the picture below. The C# compiler will parse the Xml comments and generate the Xml documentation file when you build the project. The compiler will even issue warnings about undocumented members and broken cross-references, so you can check that the documentation is complete.



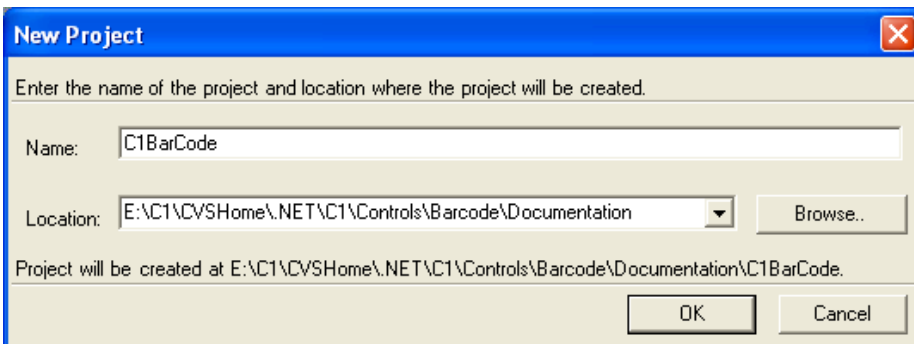
Creating a New Doc-To-Help Documenter Project

- Create a directory named **Source** in a convenient location. This folder is used to store your .DLL and .XML files for the purposes of this tutorial.
- Copy the **C1.Win.C1Barcode.DLL** file and **C1Barcode.XML** file from *C:\Program Files\ComponentOne\DocToHelp7\Tutorial\BarCode\BarCode Assembly\Original* to your **Source** directory.
- Start **ComponentOne Doc-To-Help** and select the **Open Doc-To-Help Documenter for .NET** option from the main screen.



This will bring up the **Documenter** start dialog, where you can select to create a new project or open an existing one.

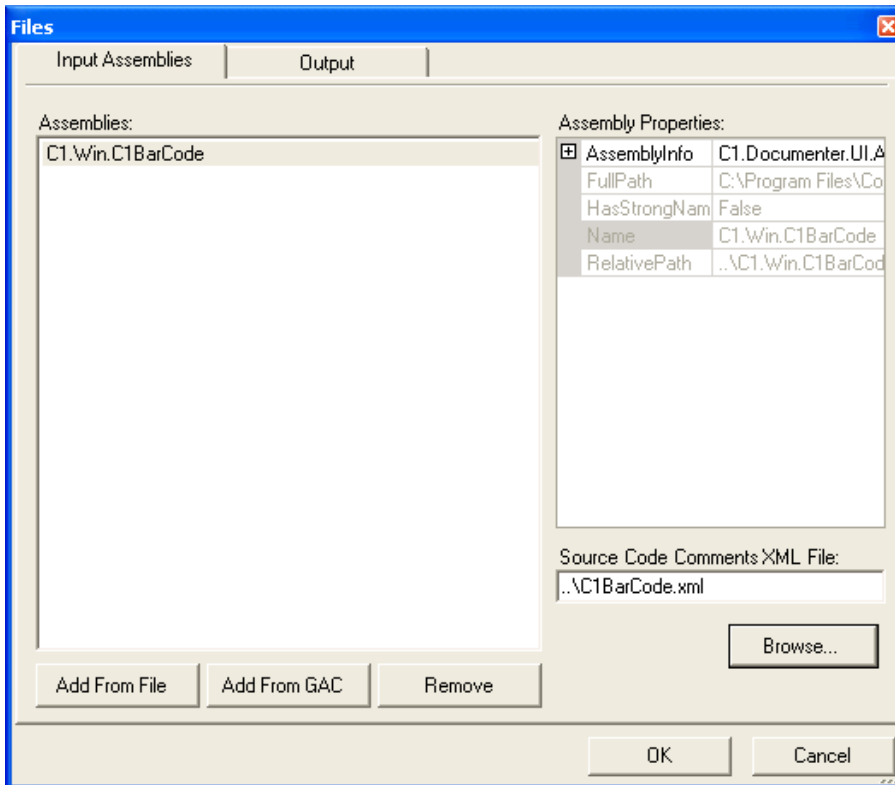
- Click the New Project button, then select a name and location for the new documentation project.



- Click **OK** and the new project will be created at the location you specified.

A dialog will prompt you for the assemblies you want to document and the location of the optional Xml documentation file created by the C# compiler.

By default, the output document file uses the same name as the project. If you want to modify the name of the output document, select the **Output** tab and modify the document name in the **Generated Document | Name** textbox.

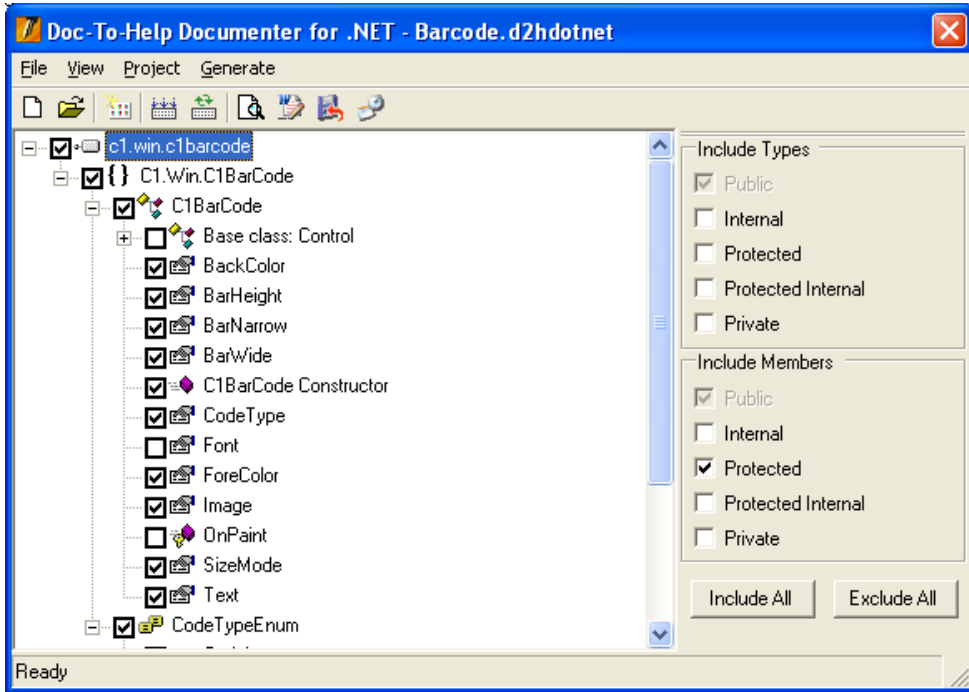


- Click **Add From File** and select **C1.Win.C1Barcode.dll** from your **Source** directory.

Note that you have the option to add from the Global Assemblies Cache (GAC). By clicking **Add From GAC**, Documenter exposes all assemblies stored in the local GAC. "Strongly Named" assemblies must reside in the GAC to be documented.

- Click **Browse** and select **C1Barcode.xml** from your **Source** directory
- After selecting the assembly and .XML file, click **OK**.

Documenter opens a second dialog that allows you to specify which elements you want to include in the documentation.



Note that in this case, we unchecked the node that contains the base class. That was done because `C1Barcode` inherits from the .NET `Control` class, and that is documented in MSDN. We will only document the members that are declared in our own assembly.

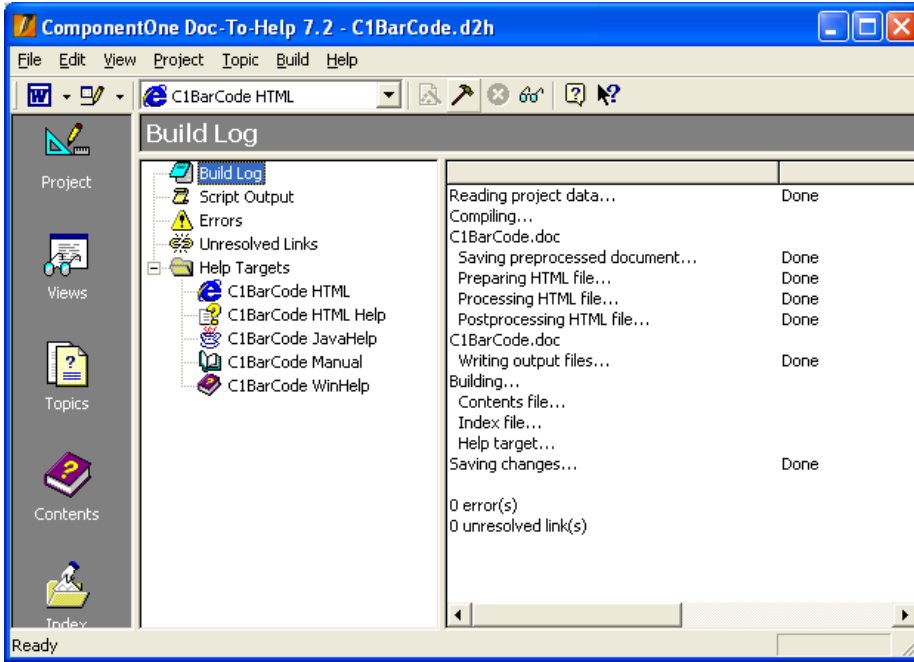
- Once you have selected the members you want to document, choose the **Generate | Generate New Document** menu option. **Doc-To-Help** creates the Doc-To-Help project files.
- When this is done (it should take about a minute), click **Close**.

The Doc-To-Help **Project Editor** opens, with the new project loaded.

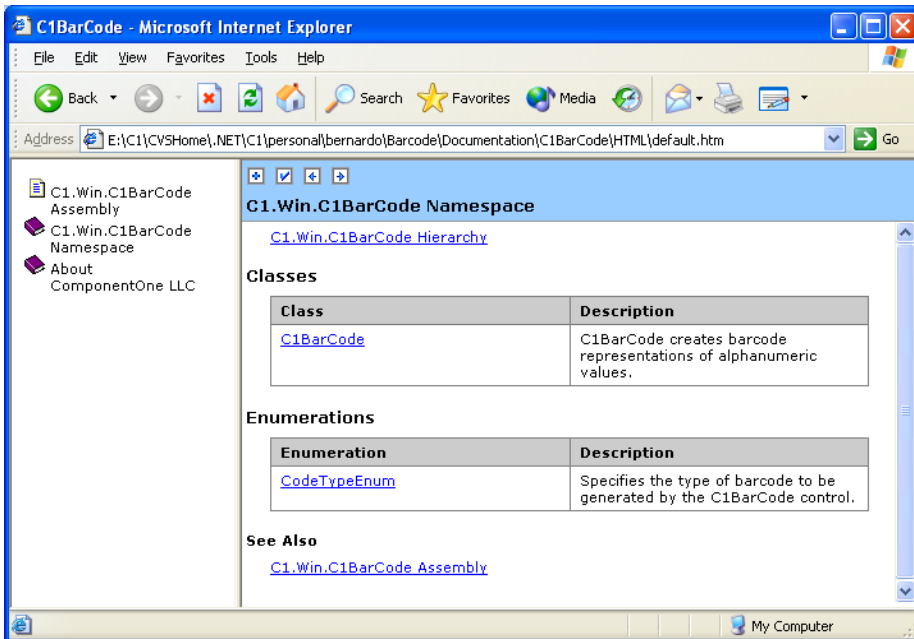
Building the Documentation

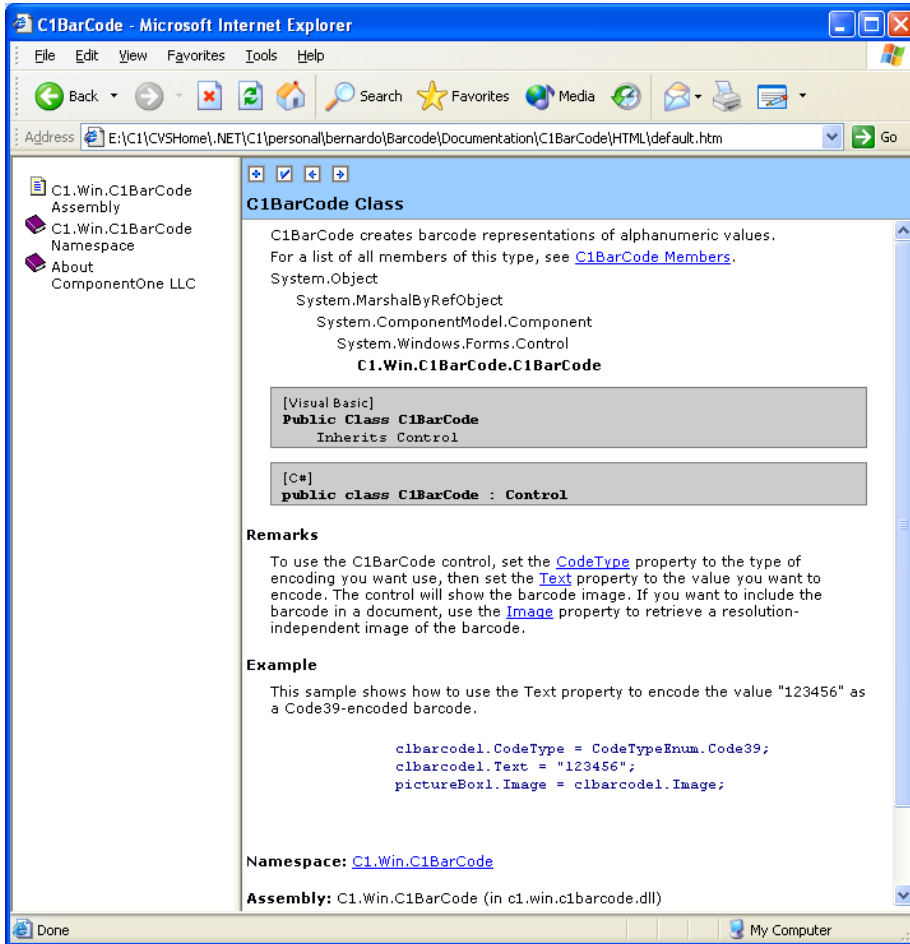
At this point, you can use all the regular **Doc-To-Help** commands to work on the project.

- Select the **HTMLHelp** target on the project toolbar, then use the **Build | Build Target** menu option to build the documentation.



- Select the **Build | View Target** menu to preview the documentation. Notice how it is formatted and cross-referenced following the Visual Studio standard:





The reference part of the documentation is ready. If you modify your assembly by adding new classes, properties, methods, etc, simply repeat the process and tell the **Documenter** to update the help project.

Note: If you are creating HTMLHelp documentation, be aware that the project path should not contain any folders with periods in their names. For example, if your project is in a folder called "c:\help projects\.**net**\c1barcode", then **Doc-To-Help** will build the project as usual, but the Microsoft help compiler will not create the chm help file. This is a limitation of the help compiler, not related to the **Documenter** or **Doc-To-Help**. To work around this problem, you have to rename the folder that contains the period or move the project to a new folder.

Add Narrative Topics to the Documentation

Most documentation projects consist of more than just a reference section. There are also custom topics such as tutorials, how-to sections, and frequently asked questions. You may also want to add boilerplate items such as company information, copyright, and licensing. These topics are not embedded in assemblies, and must be added manually.

You can make these additions using the regular **Doc-To-Help** features. When your assembly changes and you regenerate the reference part of the documentation, **Documenter** will detect and preserve your changes. This is one of the most important features in **Documenter**: you can integrate the reference and narrative portions of the documentation and always keep it always up-to-date and accurate with minimum effort.

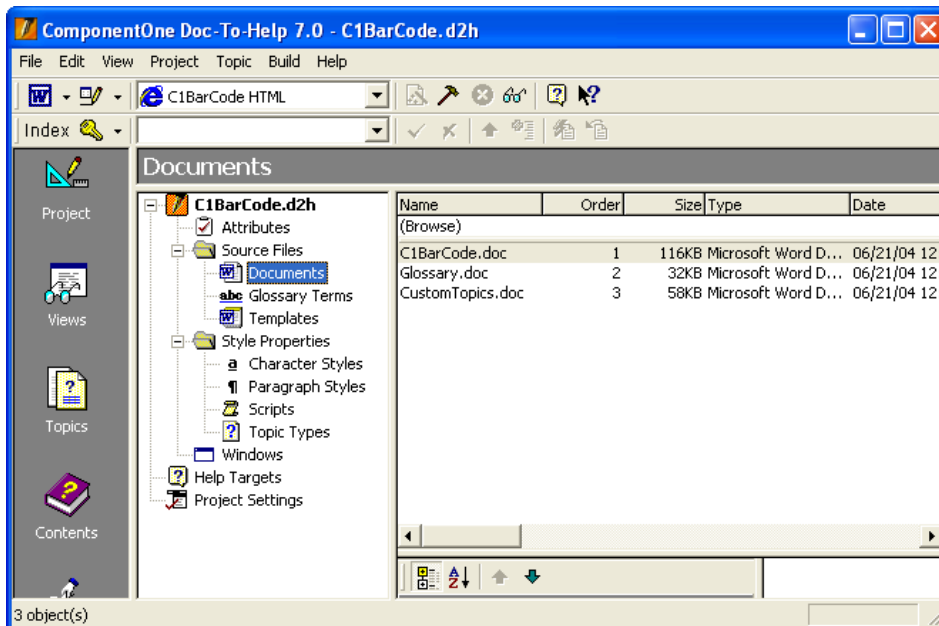
Adding a New Document to the Documenter Project

Let's add a new document containing legal and licensing information. This type of information is not product-specific. It is usually written by lawyers and included in every product.

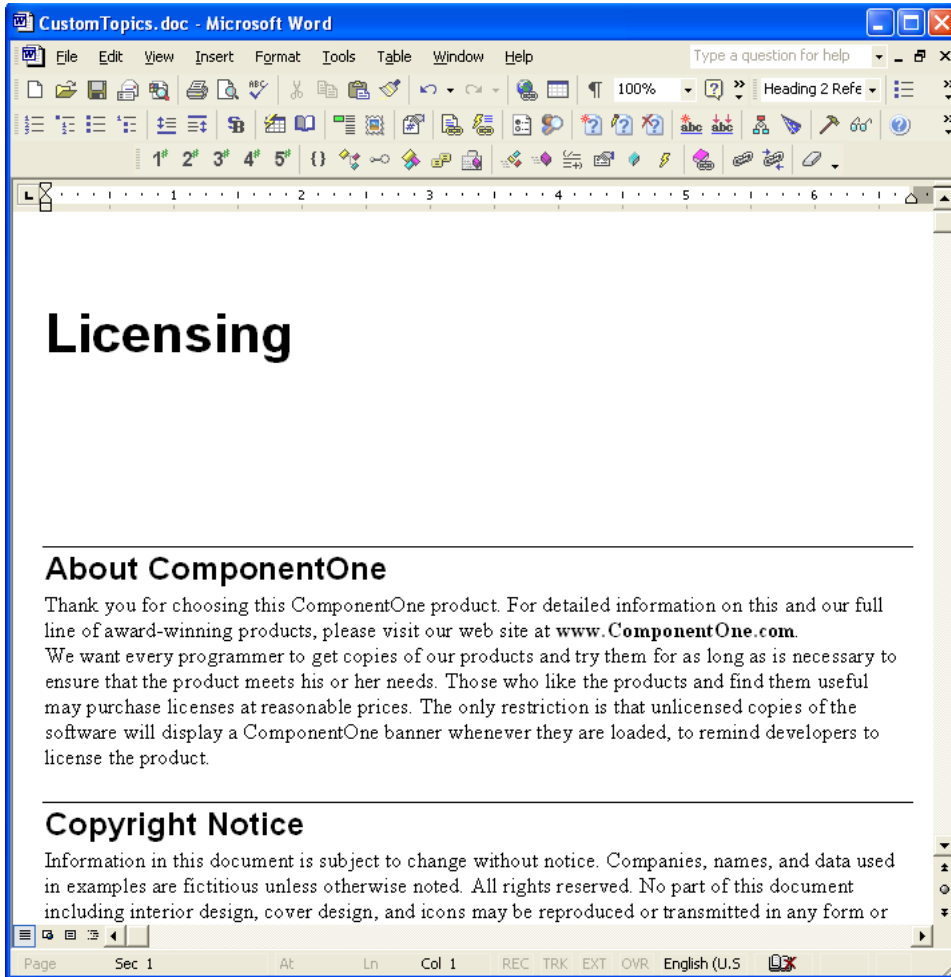
To add the new document, select the **Project | Add Document** menu.

- Browse for **CustomTopics.doc**, then click **OK**.

The **CustomTopics.doc** is added to your project.



If you examine the **CustomTopics.doc**, you will note that it already has a title formatted using the default styles and formatting used by **Doc-To-Help**.



Because CustomTopics.doc contains only general information, you can include it in all your help projects. If you want to change the information in that file (to introduce a new licensing policy for example), all you need to do is update CustomTopics.doc file and rebuild your help projects.

Adding Content to the Documenter Produced Document

- You can also add custom content to the files that were generated by the **Documenter**.
- Going back to our example, open **C1BarCode.doc** and add a paragraph below the sample code under the **C1BarCode Class** topic.

Example

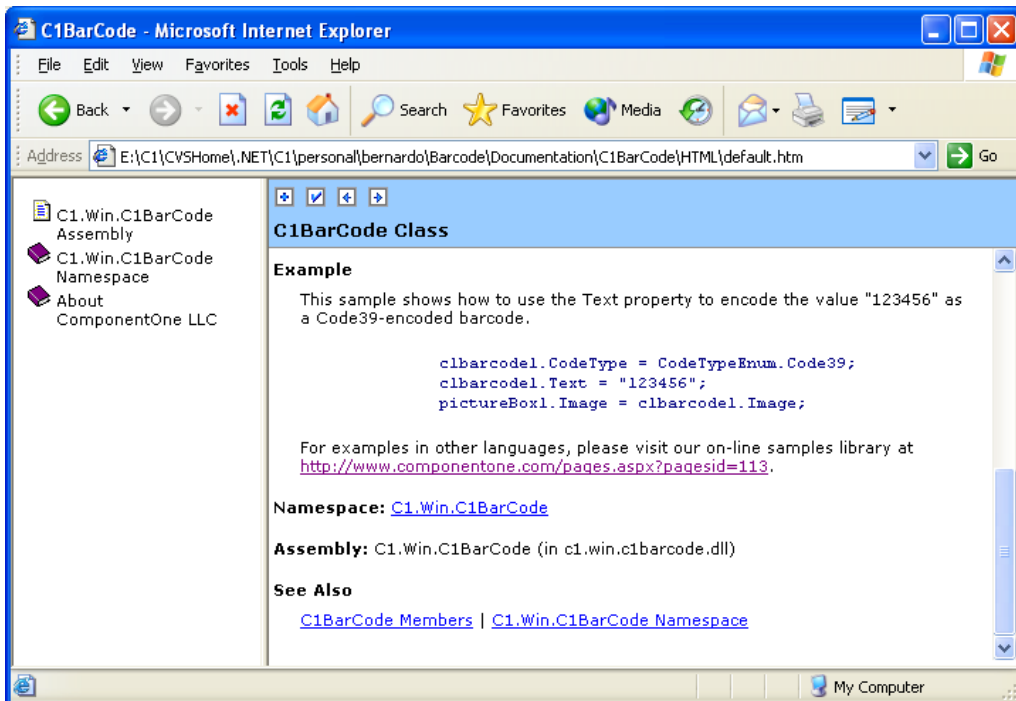
This sample shows how to use the Text property to encode the value "123456" as a Code39-encoded barcode.

```
clbarcode1.CodeType = CodeTypeEnum.Code39;
clbarcode1.Text = "123456";
pictureBox1.Image = clbarcode1.Image;
```

For examples in other languages, please visit our on-line samples library at <http://www.componentone.com/pages.aspx?pagesid=113>.

- Save and close **C1Barcode.doc**.
- Rebuild your Help project.

Note that the new content has been added to your Help file.



Note: Documenter is able to preserve the changes you make by tagging automatically generated paragraphs with a hidden header. You can add and remove as many

paragraphs as you like, but you should not make changes to the automatically generated paragraphs. If you do, your changes will be lost.

Updating the Assembly and the Documentation

Software evolves. As you release new versions of your assemblies, with new properties, events, and methods, keeping the documentation up-to-date can be a challenge.

Documenter addresses this by allowing you to regenerate the documentation while keeping the changes you made to the automatically generated files.

To demonstrate this, we will use a new version of the **C1BarCode** assembly and update the documentation to reflect the changes. The new **C1BarCode** has an additional property called **ShowText**. The new assembly is located in *C:\Program Files\ComponentOne\DocToHelp7\Tutorial\BarCode\BarCode Assembly\Updated*.

- Copy (and overwrite the existing) **C1.Win.C1BarCode.DLL** file and **C1BarCode.XML** file from *C:\Program Files\ComponentOne\DocToHelp7\Tutorial\BarCode\BarCode Assembly\Updated* your **Source** directory.
- Open Doc-To-Help.
- From the main screen, start the **Documenter**, then load the **C1BarCode** project by clicking the **Load Project** button.

Notice the **Documenter** remembers where to find the assembly, the Xml help file, and which elements to include in the documentation. All that information is stored as part of the project.

- Select Generate | Regenerate Existing Document.

The Documenter will repeat the process of analyzing the assembly and Xml help file, and will merge the changes with the existing documentation.

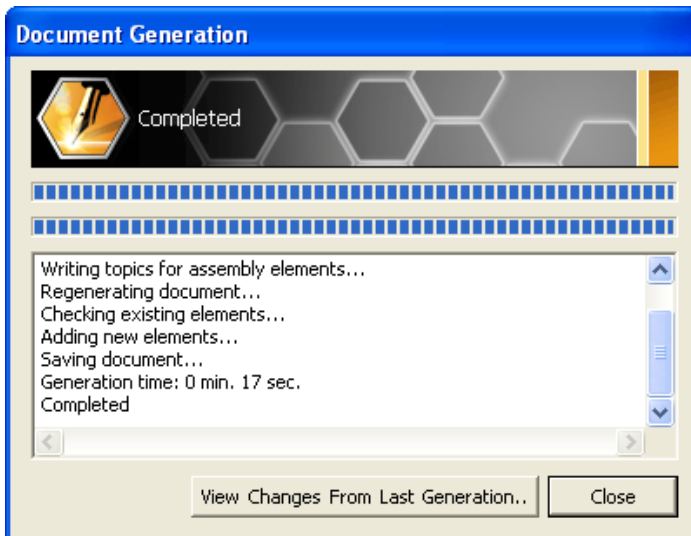
A Note About Regeneration:

Regeneration (rebuilding the automated documentation) removes generated topics and paragraphs that are no longer relevant (part of the assembly), adds new topics and paragraphs as needed, and modifies existing generated paragraphs if they need modification because of assembly changes. Documenter will preserve your changes: user added paragraphs will remain in their places or they will be moved to different places, if assembly change necessitates the move. Documenter tries to do it in the best possible way, but sometimes there can be user-added paragraphs that cannot be placed in the document after regeneration. Such paragraphs will be lost, removed from the document. For example, when you have a user added paragraph describing a method that is no longer present in the assembly. In this case, your paragraph becomes orphaned, so it is removed from the document.

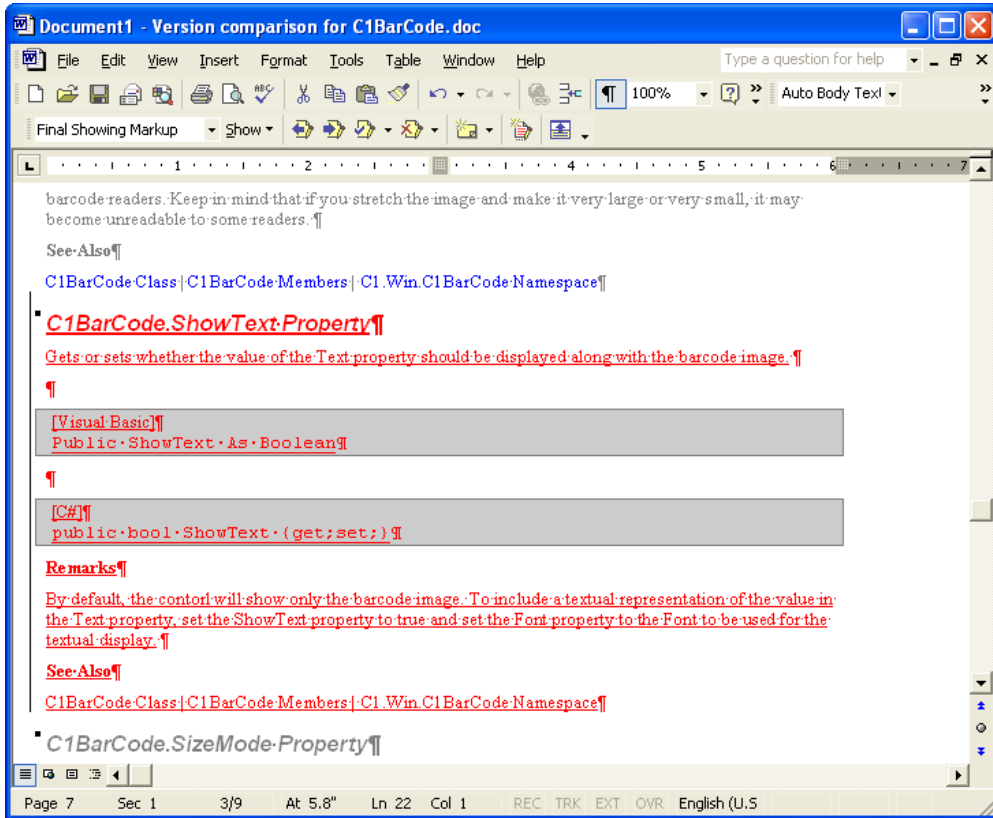
After each regeneration, you should press **View Changes From Last Generation** to examine the changes done by Documenter in regeneration and fix the issues if you find there. When fixing any issues, make sure you: only add and delete paragraphs, don't modify generated paragraphs, do not use styles starting with "Auto" in your added paragraphs.

In addition to preserving user-added paragraphs, Documenter "preserves" deletions made by the user, when it regenerates the document. This means that if a generated paragraph is deleted by the user before the regeneration, it won't appear in the regenerated document as well. Sometimes (not often) you may need to suppress this behavior, for example, when you deleted some auto-generated paragraphs by mistake and want the Documenter to generate them again, to restore them for you. To do this, you can check the checkbox **Recreate generated paragraphs that you deleted from the document** in the confirmation dialog shown when you start regeneration. But bear in mind that in this case Documenter will ignore all your previous deletions, that is, it will recreate all paragraphs that you deleted if they are still relevant in the current assembly contents.

Click the **View Changes From Last Generation** button.

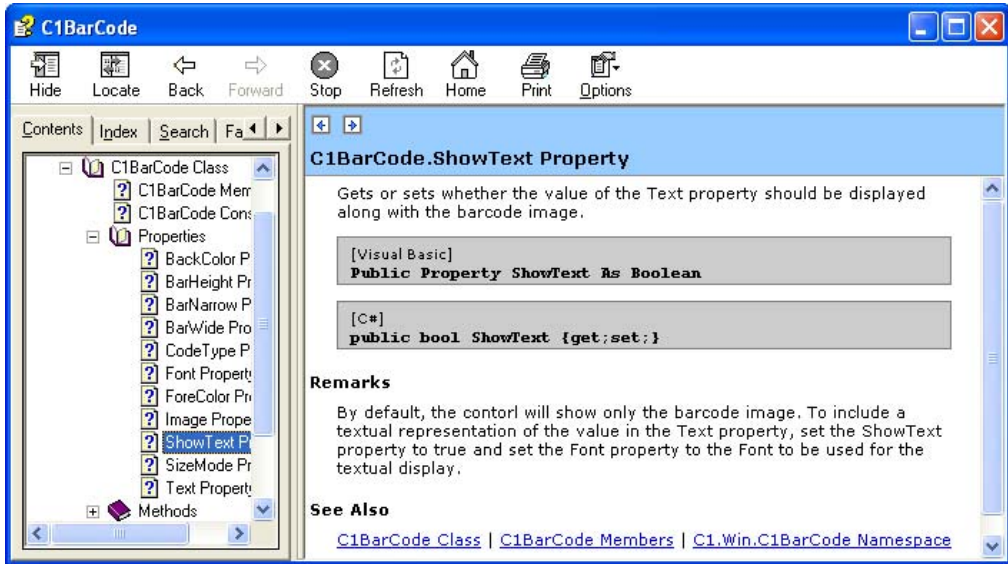


Note that the added property is highlighted in red.

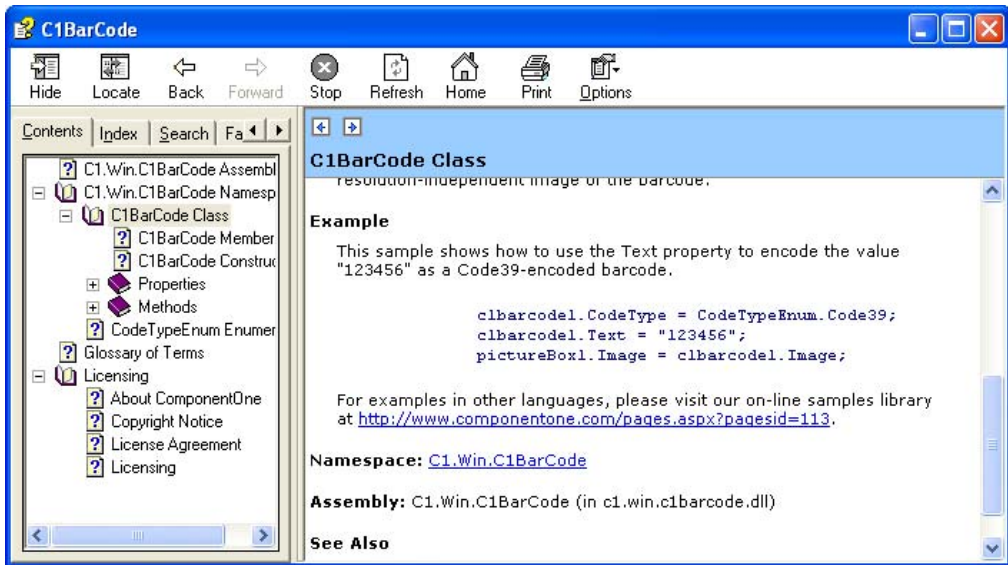


- Close the Word document displaying the changes and click the **Close** button on the **Document Generation** dialog.
- Rebuild the help project.

Notice the new topics that were added for the new properties, which were also included in the documentation outline.



Notice also how the changes that you made in the previous step were preserved.



Documenter Controls and Toolbars

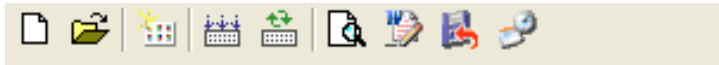
Documenter for .NET uses a set of control icons when producing and updating the automatically generated source document. Additionally, **Documenter** displays a Word toolbar for easy text formatting and style defined link control. For more information on










using styles to create automatic links, see [Using Documenter Styles to Create Links](#) (page 371).

The following two topics give you a brief overview of the functions related to each icon.

Exploring the Documenter Controls

The **Documenter** controls are used while defining and building the automated source documentation produced for Doc-To-Help. These controls are located at the top of the Documenter user interface. To learn more about these controls, see [Documenter for .NET Guided Tour](#) (page 342.)
























	New Project	Start a new Documenter project.
	Open Project	Open an existing Documenter project.
	Project Files	View and edit the assembly files for the current project.
	Generate New Document	Generate a new Documenter source document.
	Regenerate Existing Document	Regenerate an existing Documenter source document. For information on regenerating a document, see "A Note About Regeneration" in the Updating the Assembly and the Documentation (page 354)topic.
	View Document	View the current Documenter source document.
	Document Changes in Last Generation	View a document showing the changes from the last document generation.
	Backup Document	Backup your current source document (before regeneration).
	Document Backup Versions	View the document backup versions.

Exploring the Documenter Style Toolbar

The **Documenter** toolbar allows you to easily apply a link style, making link creation a matter of two clicks. Select a word or phrase and click a tool button to format it with a

link style. Doc-To-Help takes care of the rest when you build a Help target. For more information on the link styles, see *Using Documenter Styles to Create Links* (page 371).

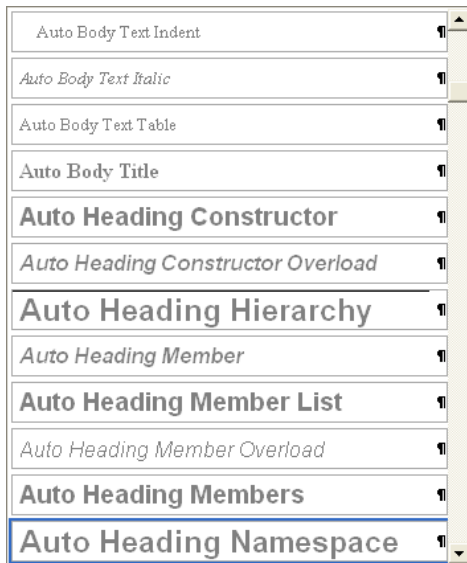


	Heading 1 Reference	Format text with Heading 1 Reference style.
	Heading 2 Reference	Format text with Heading 2 Reference style.
	Heading 3 Reference	Format text with Heading 3 Reference style.
	Heading 4 Reference	Format text with Heading 4 Reference style.
	Heading 5 Reference	Format text with Heading 5 Reference style.
	Link Namespace	Format text with Link Namespace style.
	Link Class	Format text with Link Class style.
	Link Interface	Format text with Link Interface style.
	Link Structure	Format text with Link Structure style.
	Link Enumeration	Format text with Link Enumeration style.
	Link Delegate	Format text with Link Delegate style.
	Link Constructor	Format text with Link Constructor style.
	Link Method	Format text with Link Method style.
	Link Operator	Format text with Link Operator style.
	Link Property	Format text with Link Property style.
	Link Field	Format text with Link Field style.
	Link Event	Format text with Link Event style.
	Link Topic	Format text with Link Topic style.
	Link Tag	Format text with Link Tag style.
	Show Hide Tag of Link	Show or Hide the tag segment of the link.
	Default Paragraph Font Style	Change text to the default paragraph style.

Examining the Automatically Generated Document and Project

Once Documenter produces a reference document, it is a good idea to take a look at it to see just what is included. This topic describes the major automation features included in the output document.

After opening the document, place your cursor on any heading text and click the style dropdown. Note that all of the paragraphs are formatted with a style starting with **Auto**. Documenter uses these styles during the regeneration process to distinguish auto-generated paragraphs from those added by the user.



Note: It is the users responsibility to avoid using styles starting with **Auto** in user added paragraphs. You can delete any auto-generated paragraphs you don't want to appear in the document, but cannot modify auto-generated paragraphs. More exactly, Documenter will overwrite modifications you perform to auto-generated paragraphs if it changes those paragraphs during regeneration. You are also not allowed to break generated paragraphs by adding your paragraph in the middle of an auto-generated paragraph. You must always add your paragraphs before or after an auto-generated paragraph.

Place your cursor on any link text (blue text) and click the style dropdown. Note that all of the topic links are formatted with styles starting with **Link**. These styles, and the related toolbar and a menu items (under the **Formatting** menu) are used to format link styles and to perform other frequently needed actions. These controls are provided in the `C1H_dotnet_src.dot` author template. Corresponding target templates for building

different Doc-To-Help targets are **C1H_dotnet_html.dot** (HTML-based targets), **C1H_dotnet_hlp.dot** (WinHelp), **C1H_dotnet_prn.dot** (Manual).

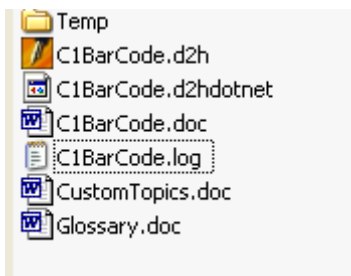


Every paragraph in the generated document starts with a hidden text that looks like this:

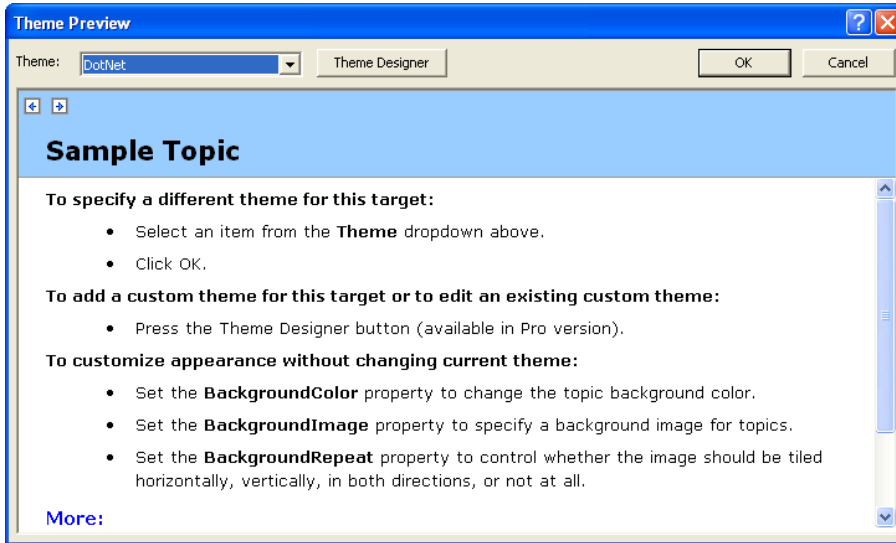
```
D2HDCmtr_: Types_Type: TargetAssembly: TargetAssembly. AFirstClassDerived%
```

This is the "block key". Documenter uses it for internal purposes, to make document regeneration possible. Normally, the block key should be ignored by the user. Just hide hidden text to remove it from the screen.

In addition to the .doc file, Documenter also creates a .log file in the project directory each time it generates or regenerates a document. This file must be kept unchanged in the project directory. Documenter needs it to be able to regenerate the document.




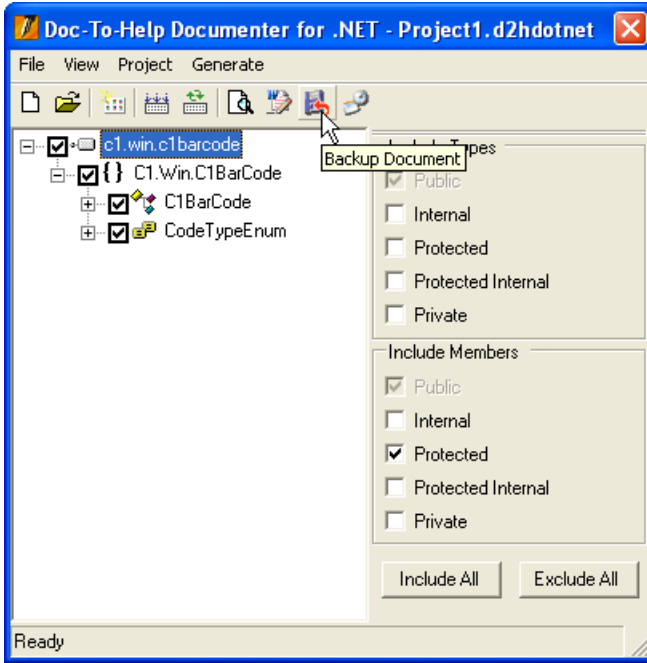
The targets in the D2H project have their Theme property set to a special theme ".NET". You can use D2H Theme Designer to modify that theme or to create a new theme and use it in the D2H project.




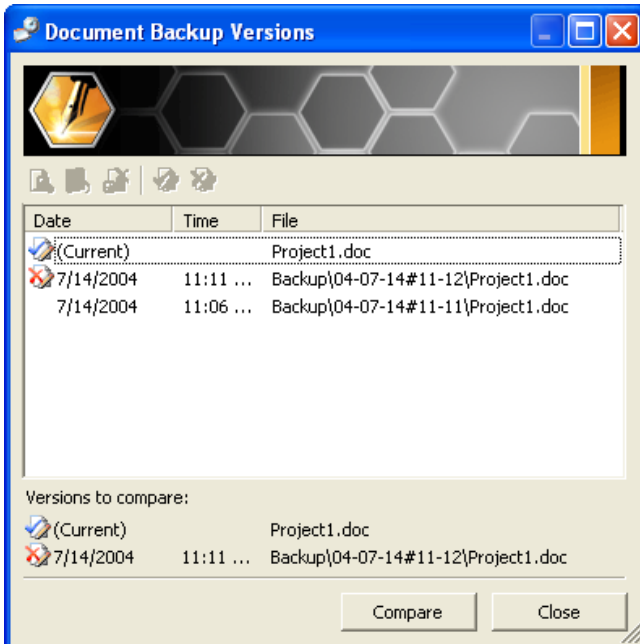
Adding Narrative to a Documenter Project

Documenter allows you to add narrative topics to the documentation and keeps those sections intact when your component evolves and you rebuild the documentation (by selecting the updated assemblies). This is a unique feature that provides seamless integration between the reference and narrative parts of the documentation.

Another important feature of **Documenter** is the ability to archive and compare versions of your automatically produced documentation. By simply clicking the **Backup Document** button , you can store a "snapshot" of your document before rebuilding with an updated .NET assembly.



By clicking the **Document Backup Version** button , you can view and compare your stored back up versions.



In addition to adding, retaining and archiving the narrative text in your Documenter produced reference sections; you can insert additional Word documents into your Doc-To-Help project. Any Word document added to a Documenter project has complete access to all of the Documenter features, including the topic link styles. For more information on using the topic link styles, see [Adding Topic Links to a Documenter Project](#) (page 364) and [Using Documenter Styles to Create Links](#) (page 371).

Adding Topic Links to a Documenter Project

Unlike other documentation tools, Doc-To-Help Documenter for .NET automates the entire process of creating program documentation, not just the reference materials, but the entire document set. Other documentation tools help to generate the reference sections of your documentation only. But quality software documentation contains additional documentation (narrative sections, conceptual sections, tutorials, etc.) in addition to reference materials. Documenter takes the next step by helping you produce these additional documents (which frequently have multiple links to the reference materials) by automating the topic link process.

By using the new link styles, creating topic links is a simple matter of formatting text. Because narrative sections usually have numerous links to the reference sections of the documentation, using the link style feature makes it easy for you to add topic jumps as you write, without worrying about dialog boxes, selecting topics, etc.

To add a link to the topic below:

C1Barcode.BackColor Property

Gets or sets the background color for the control.

- Select the location and text for the link.

C1Barcode.BarNarrow Property

Gets or sets the width of the narrow bars in screen pixels.

For more information on the colors, see [C1Barcode.BackColor](#).

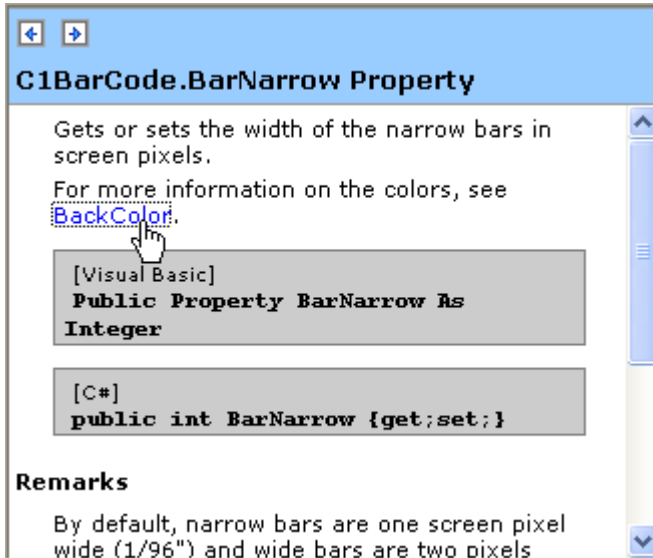
- Click the **Link Property** style button.



In this case, we selected the **Link Property** button because the topic we are linking to describes a property. If we were linking to a topic that describes a method, we would select **Link Method**, for a delegate, we would select **Link Delegate**, etc.

- Build the Help target.

Note that the link is in the Help target, minus the **C1Barcode**, which is stripped out by Documenter.



For more information on Link Styles, see Using Documenter Styles to Create Links (page 371).

Using the Link Tag Style

In some instances, it may be necessary to insert a topic link using text that does not correspond to the topic text. In this case, you can use the **Link Tag** style.

To add a link to the topic below:



- Select the location and text for the link.

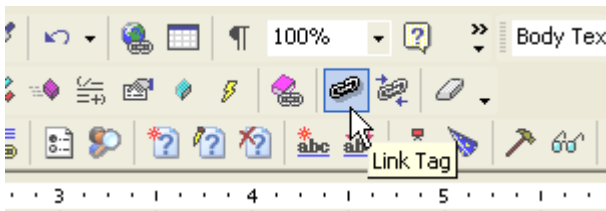
To learn more about the C1Barcode Hierarchy, click [here@C1 Win C1BarCode Hierarchy](#).

You will note that we have specially formatted this text to use the Link Tag style. The formatting is:

`j ump text@j ump topi c`

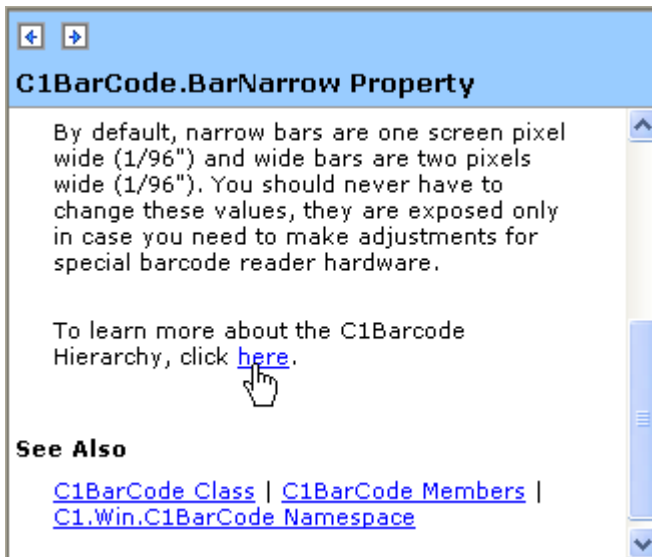
In this case **Here** is the jump text, then the @ symbol (without spaces), then **C1 WinC1BarCode Hierarchy** is the jump topic. (Replace all "dots" (periods) with spaces.)

- Click the **Link Tag** style button.



- Build the Help target.

Note that the link is in the Help target, minus **@C1 WinC1BarCode Hierarchy**, which is stripped out by Documenter.



In cases where you want see how your document will look without the links, select the link tag and click the **Show/Hide Tag of Link**  on the Doc-To-Help Documenter for

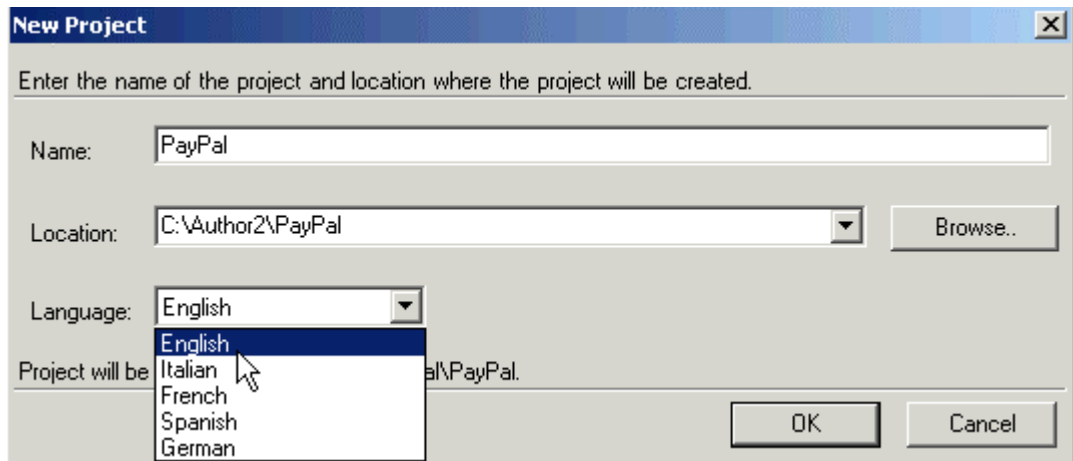
.NET toolbar. To make the tag part reappear, press the **Show/Hide Tag of Link** button again.

- To hide or show the tag parts for all Link Tag links in the document, press **Show/Hide Tag of Link** while holding the **Ctrl** key.

For more information on Link Styles, see Using Docucenter Styles to Create Links (page 371).

Multilanguage Support in Docucenter for .NET

Docucenter for .NET can generate documentation in English, French, German, Spanish, Italian and Russian. And with custom localization, you can generate output in any other language, see Customization and Localization of Docucenter for .NET (page 367). When creating a Docucenter project (or updating from a previous Docucenter version), you can choose the language of your project. The figure below illustrates this.

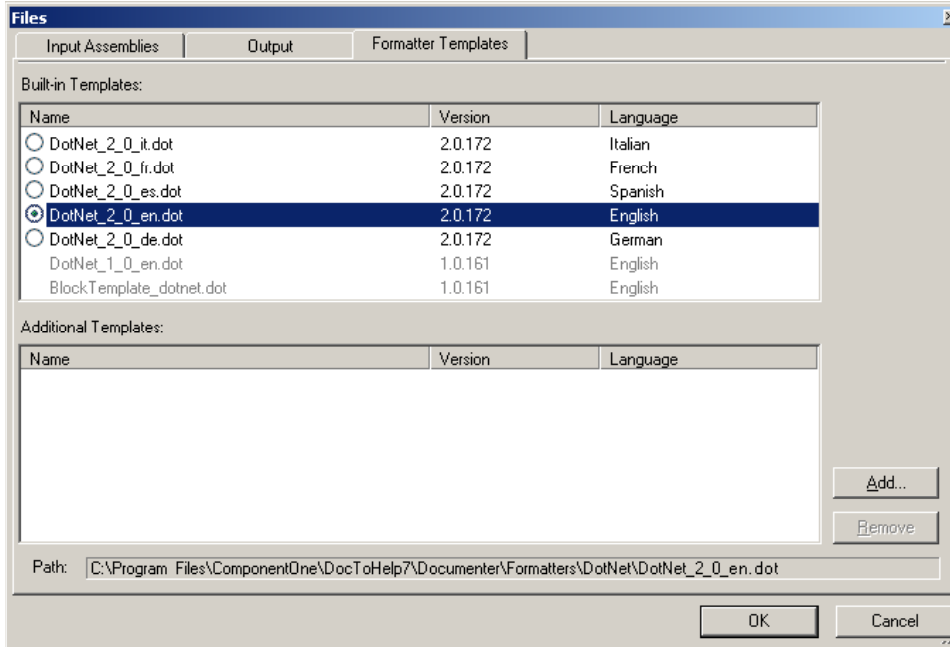


Customization and Localization of Docucenter for .NET

You can customize Docucenter for .NET output by modifying special templates called *formatter templates*. You can change and add texts, formatting, graphics, etc. You can also use this customization feature to localize the output so it is produced in a language not covered by the built-in choice of languages (see Multilanguage Support in Docucenter for .NET (page 367).)

Understanding the Formatter Templates

Documenter for .NET uses a “formatter template” to generate the document. The formatter template used in a project is specified in the project Files dialog on the Formatter Templates tab.



Documenter for .NET comes with built-in formatter templates:

- **Formatter Version 1.0 (English only):** this template is enabled only in projects created with the older Documenter version.
- **Formatter Version 2.0 (all supported languages, each language in a separate template):** this template is enabled in projects created with this formatter version, version 2.0.

In addition to the built-in templates, you can create your own formatter templates to customize **Documenter** output. You can start with one of the built-in templates and modify it as necessary.

Do not change a built-in template. Always copy it to a new file and rename it.

Documenter addresses formatter templates by file name alone. The file names (without path) of all formatter templates on your computer must be unique. When you have created a new formatter template (or received it from someone else), you must add it to the Documenter in order to use it. This is done in the Files dialog’s Formatter Templates tab with the Add button adding templates to the Additional Templates list. This list is global to all Documenter projects on your computer. If you want to copy a Documenter

project that uses a custom formatter template to another computer, you must copy the template with the project and then add it to the Files dialog on the destination computer.

Understanding Formatter Template “Blocks”

A Formatter template consists of multiple “blocks”. Each block is marked with a bookmark (the bookmark contains the entire block). Bookmark names must be the same as in the original built-in template. A customized template must contain all bookmarks (blocks) present in the original. Bookmark names have the prefix “D2HDcmtr_” and the remainder of the name represents the part of the output (usually one paragraph) generated using this block of the template.

To see the blocks with their bookmarks, check the Bookmarks check box on the View tab of the Word Tools | Options dialog. Use the Bookmark dialog (Insert | Bookmark in Word menu) if you need to locate a bookmark.

Customizing Formatter Template

ShowExample Button:

The Formatter template has a Formatter Help toolbar that contains the ShowExample button. By using this button, you can easily find out, for any block, the place in the generated document that this particular block generates. Position on the block you are interested in and press the ShowExample button. This will show a sample generated project in a separate Word window with selection highlighting text generated by this block.

Variables:

Most blocks contain “variables”. Variables are enclosed in angle brackets (see figure).

<Name>

When generating the output, Documenter substitutes values for these variables. **Do not change the variables in the formatter template. They are hardcoded in the software; changing them will disrupt generation.**

You may delete a variable if you need to (although it is rarely necessary), but do not change them. All constant words, phrases and other content that may need to be changed to customize or localize a template is in the constant parts of the blocks, not in the variables. The styles with which variables are formatted in the template are also essential for proper generation. You can modify style attributes (font, etc.), but do not change the name of a style with which a variable is formatted. All other styles (in the constant parts of the blocks) can be used without restrictions. However, you must ensure that all paragraph styles used in the formatter template start with “Auto” because all generated content must be formatted with “Auto” styles to distinguish it from paragraphs added by the user

Localizing Formatter Template

If you customize a formatter template in order to localize it, translate it to a language other than languages built-in in the Documenter, then you should consider using the `FormatterLocalizer` utility supplied with Documenter for .NET. Using this utility, localization is much easier than changing a formatter template by hand. This utility helps you to specify and maintain a list of translations for English phrases, and it automatically produces a localized formatter template using these translations.

The `FormatterLocalizer` utility and a document explaining how to use it are located in the Doc-To-Help installation directory in `Documenter\Customization\FormatterLocalizer`.

Read-made translations for all built-in languages are provided with the utility.

Restrictions in Changing Texts in Formatter Template

Changing texts in a formatter template, you must be aware that some texts are used to create links, so they can not always be changed arbitrarily without breaking the links. This mostly concerns texts used for topic titles. Some blocks are interdependent in the sense that changing a block you must change its dependent blocks correspondingly, to avoid breaking links. Some (not many) blocks can only be changed if you make corresponding changes in the scripts in the Doc-To-Help project. For example, if you change title “MyProperty Property” to “MyProperty (property)”, then links for MyProperty in the text will be broken, because the scripts in the Doc-To-Help project assume a property topic title has a certain format.

For the complete list of rules and restrictions in changing formatter template texts, with complete lists of dependent blocks, see `Documenter\Customization\Restrictions.doc` in the Doc-To-Help installation directory.

Formatter Versions in Documenter for .NET

When a new major version of Documenter for .NET is released, it usually has new features and improved formatting that you want to use in your projects. This happens transparently when you create a new project with the new version, you just benefit from new features and enhancements. However, there are some issues in upgrading an existing project to a new major Documenter version.

Upgrading

When you open an old project with the new **Documenter** version, you will be prompted to upgrade by creating a new project.

Normally, the upgrade is chosen, in which case a new project will be created with all your settings copied to the new project. The new project belongs to the new formatter version. Retain the old project for backup purposes.

The only time you would consider not upgrading to the new formatter version is when you have already entered many paragraphs of your own text to the generated document and do not want to repeat that in the new version. Because you have to generate the project from scratch after upgrading the project, your changes will not appear in the document until you re-add them. If your project does not contain manually-added content or if you do not mind re-adding the material, the upgrade is always recommended.

Should you choose not to upgrade, **Documenter** will continue working on your project just as the previous version did. **Documenter** supports several formatter versions, including the previous version allowing you to continue working on an old document without upgrading if you so choose.

Viewing a Formatter Version of Your Project

To view the formatter version of your project, open Files dialog | Formatter Templates tab and look at the selected formatter template. It shows Version in a three-number format: <major version>.<minor version>.<build number>. Formatters with different major version numbers are incompatible. However, both are supported by the same **Documenter** version. Incompatibility refers only to manually-added content in a document created with an old formatter. The manually-added content will not be automatically retained by the newer formatter (Regenerate Existing Document is disabled). Minor version and build number increase do not break formatter compatibility.

Using Documenter Styles to Create Links

ComponentOne Documenter for .NET automatically generates links in reference documentation using the Doc-To-Help character style mechanism. This allows the links to be specified as simple text in the document. The fact that a text is actually a link is determined by formatting the text with a pre-defined style. There is no additional information stored anywhere in the document or in the Doc-To-Help project that is needed to create the link. When compiling the help target, Doc-To-Help recognizes the link style and creates the required connection automatically. You can see links created by Documenter for .NET in the generated document as words and phrases formatting with styles such as Link Class, Link Method, Link Property, etc.

Documenter for .NET does not require your participation in or even knowledge of the process of creating links in the text it generates. However, you will need link styles if you want to use one of the most powerful features of Documenter for .NET, the ability to add your paragraphs, topics and whole documents and be able to easily specify links from your added contents to Documenter generated contents.

To create a link from your text to a generated topic (or to an added topic), all you need to do is to format a word or a phrase in your text (usually it is a name of a property,

method, class, etc.) with an appropriate style. You don't need to search for a topic in some huge list box, or even ensure that your link text corresponds exactly to the target topic name. For example, to create a link from the text **MyProperty** to the topic **MyClass.MyProperty Property**, you simply format the word **MyProperty** in your text with **Link Property** style. Given that such links are usually very numerous in narrative topics, you can see how Documenter for .NET makes the task of writing narrative documentation much easier. It completely automates creating reference parts, and for narrative parts it automates the mundane task of creating links. It frees you up to do what is most important, write the narrative itself.


If you use the correct style and, optionally, specify your preferences using one of a few special characters described in Special Characters in Link Text (page 383), Doc-To-Help will automatically determine the link target topic and will optionally modify the link text according to your preferences. All you need to do to create a link is choose the correct link style and apply it to the text. Doc-To-Help makes it even easier for you providing a special toolbar so you can easily apply a link style, making creating a link really a matter of two clicks, double-click a word, say, a method name, to select it, and click a toolbar button to format it with a link style. For more information, see Documenter Controls and Toolbars (page 357.)


The following topics provide more information on the **Documenter for .NET** link styles and link functionality.

The Link Tag Style

Link Tag style is used to link to any topic regardless of the link text. The target topic title is specified in the second part of the link text, called the *tag part*, invisible in the help file.

Using this style, link text must consist of two parts separated with '@'. The first part is the link text as it will appear in the help file. The second part is the topic title specifying the link's target topic. Doc-To-Help removes the second part when it builds the help file.

To create a link using this style, select the link text and press **Link Tag**  on the Doc-To-Help Documenter for .NET toolbar. Then add '@' to the end and type the full topic title. Or you can first type '@' in the end and paste the topic title and then select the whole text and press the **Link Tag** toolbar button.

If you want to see how your document will look without the links, select the link tag and click the **Show/Hide Tag of Link**  on the Doc-To-Help Documenter for .NET toolbar. To make the tag part reappear, press the **Show/Hide Tag of Link** button again.


To hide or show the tag parts for all Link Tag links in the document, press **Show/Hide Tag of Link** while holding the **Ctrl** key.

The following table gives you some examples of using the Link Tag style:

In document	In help file	Links to topic
This is a link to FileInfo@FileInfo Class	This is a link to FileInfo	FileInfo Class
Another link to FileInfo@FileInfo Class	Another link to FileInfo	FileInfo Class
public FileInfo CopyTo(string);@FileInfo.CopyTo Method (String)	public FileInfo CopyTo(string);	FileInfo.CopyTo Method (String)

The Link Topic Style

Link Topic style is used to link to the topic whose title is an exact textual match with the link text.

To create a link using this style, select the link text and press  **Link Topic** on the Doc-To-Help Documenter for .NET toolbar.


For this style, Doc-To-Help applies only the nested type separator substitution to the link text, see Special Characters in Link Text (page 383). Other special characters are not applicable to this style.

The following table gives you some examples of using the Link Topic style:

In document	In help file	Links to topic
FileInfo Class	FileInfo Class	FileInfo Class
System.Data Namespace	System.Data Namespace	System.Data Namespace
IResourceReader Members	IResourceReader Members	IResourceReader Members
Environment+SpecialFolder Enumeration	Environment.SpecialFolder Enumeration	Environment.SpecialFolder Enumeration

The Link Namespace Style

Link Namespace style is used to link the text representing a namespace name to the topic describing that namespace.

To create a link using this style, select the link text and press **Link Namespace**  on the Doc-To-Help Documenter for .NET toolbar.


For this style, Doc-To-Help compiler will add the word **Namespace** to the text and establish a link to the topic with the title coinciding with the resulting text.

The following table illustrates the usage of the Link Namespace style:

In document	In help file	Links to topic
System	System	System Namespace
System.Data	System.Data	System.Data Namespace

The Link Class Style

Link Class style is used to link a class name to the topic describing that class.

To create a link using this style, select the link text and press **Link Class**  on the Doc-To-Help Documenter for .NET toolbar.

For this style, the Doc-To-Help compiler will apply the nested type separator substitution to the link text, if applicable, see Special Characters in Link Text (page 383), add the word **Class** to the text, and establish a link to the topic with the title coinciding with the resulting text.


By default, Doc-To-Help removes the qualifying namespace, if present, from the class name in the link text. You can change this behavior by using the special characters, exclamation sign and parenthesis. See Special Characters in Link Text (page 383).

The following table demonstrates the usage of the Link Class style:

In document	In help file	Links to topic
FileInfo	FileInfo	FileInfo Class
System.IO.FileInfo	FileInfo	System.IO.FileInfo Class
!System.IO.FileInfo	System.IO.FileInfo	System.IO.FileInfo Class
(System.IO.)FileInfo	System.IO.FileInfo	FileInfo Class

The Link Interface Style

Link Interface style is used to link an interface name to the topic describing that interface.

To create a link using this style, select the link text and press **Link Interface**  on the Doc-To-Help Documenter for .NET toolbar.

For this style, the Doc-To-Help compiler will apply the nested type separator substitution to the link text, if applicable, see Special Characters in Link Text (page 383), add the word **Interface** to the text, and establish a link to the topic with the title coinciding with the resulting text.


By default, Doc-To-Help removes the qualifying namespace, if present, from the interface name in the link text. You can change this behavior by using the special characters, exclamation sign and parenthesis. See [Special Characters in Link Text](#) (page 383).

The following table demonstrates the usage of the Link Interface style:

In document	In help file	Links to topic
IResourceReader	IResourceReader	IResourceReader Interface
System.Resources.IResourceReader	IResourceReader	System.Resources.IResourceReader Interface
!System.Resources.IResourceReader	System.Resources.IResourceReader	System.Resources.IResourceReader Interface
(System.Resources.)IResourceReader	System.Resources.IResourceReader	IResourceReader Interface

The Link Structure Style

Link Structure style is used to link a structure name to the topic describing that structure.

To create a link using this style, select the link text and press **Link Structure**  on the Doc-To-Help Documenter for .NET toolbar.

For this style, the Doc-To-Help compiler will apply the nested type separator substitution to the link text, if applicable, see [Special Characters in Link Text](#) (page 383), add the word **Structure** to the text, and establish a link to the topic with the title coinciding with the resulting text.

By default, Doc-To-Help removes qualifying namespace, if present, from the structure name in the link text. You can change this behavior by using the special characters, exclamation sign and parenthesis. See [Special Characters in Link Text](#) (page 383).

The following table demonstrates the usage of the Link Structure style:

In document	In help file	Links to topic
Point	Point	Point Structure
System.Drawing.Point	Point	System.Drawing.Point Structure
!System.Drawing.Point	System.Drawing.Point	System.Drawing.Point Structure
(System.Drawing.)Point	System.Drawing.Point	Point Structure

The Link Enumeration Style

Link Enumeration style is used to link an enumeration name to the topic describing that enumeration.

To create a link using this style, select the link text and press **Link Enumeration**  on the Doc-To-Help Documenter for .NET toolbar.

For this style, the Doc-To-Help compiler will apply the nested type separator substitution to the link text, if applicable, see [Special Characters in Link Text](#) (page 383), add the word **Enumeration** to the text, and establish a link to the topic with the title coinciding with the resulting text.


By default, Doc-To-Help removes qualifying namespace, if present, from the enumeration name in the link text. You can change this behavior by using the special characters, exclamation sign and parenthesis. See [Special Characters in Link Text](#) (page 383).

The following table demonstrates the usage of the Link Enumeration style:

In document	In help file	Links to topic
TraceMode	TraceMode	TraceMode Enumeration
System.Web.TraceMode	TraceMode	System.Web.TraceMode Enumeration
!System.Web.TraceMode	System.Web.TraceMode	System.Web.TraceMode Enumeration
(System.Web.)TraceMode	System.Web.TraceMode	TraceMode Enumeration

The Link Delegate Style

Link Delegate style is used to link a delegate name to the topic describing that delegate.

To create a link using this style, select the link text and press **Link Delegate**  on the Doc-To-Help Documenter for .NET toolbar.

For this style, the Doc-To-Help compiler will apply the nested type separator substitution to the link text, if applicable, see [Special Characters in Link Text](#) (page 383), add the word **Delegate** to the text, and establish a link to the topic with the title coinciding with the resulting text.

By default, Doc-To-Help removes qualifying namespace, if present, from the delegate name in the link text. You can change this behavior by using the special characters, exclamation sign and parenthesis. See [Special Characters in Link Text](#) (page 383).

The following table demonstrates the usage of the Link Delegate style:

In document	In help file	Links to topic
TypeFilter	TypeFilter	TypeFilter Delegate
System.Reflection.TypeFilter	TypeFilter	System.Reflection.TypeFilter Delegate
!System.Reflection.TypeFilter	System.Reflection.TypeFilter	System.Reflection.TypeFilter Delegate
(System.Reflection.)TypeFilter	System.Reflection.TypeFilter	TypeFilter Delegate

The Link Type Style

Link Type style is used to link a type name to the topic describing that type. A type can be a class, an enumeration, an interface, a structure or a delegate.

Note: Using this style is not necessary and not recommended. You can always use a specialized style instead, one of the following: Link Class, Link Enumeration, Link Interface, Link Structure, or Link Delegate. The Link Type style is used by Documenter for .NET for internal purposes, to simplify the process of generating the document, and is documented here only for completeness.

The link text starts with a special character followed by colon. The special character specifies the kind of the type:

C - Class

D - Delegate

E - Enumeration

I - Interface

S - Structure

For this style, the Doc-To-Help compiler removes the special character and colon. The target topic title is determined by the remainder of the link text with the added type kind name, which is one of the words **Class**, **Delegate**, **Enumeration**, **Interface** or **Structure**.

To create a link using this style, prepend the text with one of the special characters (uppercase) and colon, select the text including the special character and format it with the **Link Type** style using standard Microsoft Word style formatting.

For this style, Doc-To-Help applies all transformations determined by the special characters '!', '()' and '+', as for the corresponding specialized styles for type links, see Special Characters in Link Text (page 383).

The following table demonstrates the usage of the Link Type style:


In document	In help file	Links to topic
C:FileInfo	FileInfo	FileInfo Class
I:IResourceReader	IresourceReader	IResourceReader Interface
S:Point	Point	Point Structure
E:TraceMode	TaceMode	TraceMode Enumeration
D:TypeFilter	TypeFilter	TypeFilter Delegate

The following table demonstrates using special characters with the Link Type style:

In document	In help file	Links to topic
C:FileInfo	FileInfo	FileInfo Class
C:System.IO.FileInfo	FileInfo	System.IO.FileInfo Class
C:!System.IO.FileInfo	System.IO.FileInfo	System.IO.FileInfo Class
C:(System.IO.)FileInfo	System.IO.FileInfo	FileInfo Class

The Link Field Style

Link Field style is used to link a field name to the topic describing that field.

To create a link using this style, select the link text and press **Link Field**  on the Doc-To-Help Documenter for .NET toolbar.

For this style, the Doc-To-Help compiler will apply the nested type separator substitution to the link text, if applicable, see Special Characters in Link Text (page 383), add the word **Field** to the text, and establish a link to the topic with the title coinciding with the resulting text.

By default, Doc-To-Help removes the name of the type containing the field, if the type name is present. You can change this behavior by using the special characters, exclamation sign and parenthesis. See Special Characters in Link Text (page 383).


The following table demonstrates the usage of the Link Field style:

In document	In help file	Links to topic
Timeout.Infinite	Infinite	Timeout.Infinite Field
!Timeout.Infinite	Timeout.Infinite	Timeout.Infinite Field

In document	In help file	Links to topic
System.Threading.Timeout.Infinite	Infinite	System.Threading.Timeout.Infinite Field
!System.Threading.Timeout.Infinite	System.Threading.Timeout.Infinite	System.Threading.Timeout.Infinite Field
(System.Threading.)Timeout.Infinite	System.Threading.Timeout.Infinite	Timeout.Infinite Field

The Link Event Style

Link Event style is used to link an event name to the topic describing that event.

To create a link using this style, select the link text and press **Link Event**  on the Doc-To-Help Documenter for .NET toolbar.

For this style, the Doc-To-Help compiler will apply the nested type separator substitution to the link text, if applicable, see Special Characters in Link Text (page 383), add the word **Event** to the text, and establish a link to the topic with the title coinciding with the resulting text.


By default, Doc-To-Help removes the name of the type containing the event, if the type name is present. You can change this behavior by using the special characters, exclamation sign and parenthesis. See Special Characters in Link Text (page 383).

The following table demonstrates the usage of the Link Event style:

In document	In help file	Links to topic
Form.Load	Load	Form.Load Event
!Form.Load	Form.Load	Form.Load Event
System.Windows.Forms.Form.Load	Load	System.Windows.Forms.Form.Load Event
!System.Windows.Forms.Form.Load	System.Windows.Forms.Form.Load	System.Windows.Forms.Form.Load Event
(System.Windows.Forms.)Form.Load	System.Windows.Forms.Form.Load	Form.Load Event

The Link Property Style

Link Property style is used to link a property name to the topic describing that property.

To create a link using this style, select the link text and press **Link Property**  on the Doc-To-Help Documenter for .NET toolbar.

For this style, the Doc-To-Help compiler will apply the nested type separator substitution to the link text, if applicable, see Special Characters in Link Text (page 383), add the word **Property** to the text, and establish a link to the topic with the title coinciding with the resulting text.


By default, Doc-To-Help removes the name of the type containing the property, if the type name is present. You can change this behavior by using the special characters, exclamation sign and parenthesis. See Special Characters in Link Text (page 383). You can also use the argument list for property overloads with arguments, see Argument List in Links (page 384).

The following table demonstrates the usage of the Link Property style:

In document	In help file	Links to topic
FileInfo.Exists	Exists	FileInfo.Exists Property
!FileInfo.Exists	FileInfo.Exists	FileInfo.Exists Property
System.IO.FileInfo.Exists	Exists	System.IO.FileInfo.Exists Property
!System.IO.FileInfo.Exists	System.IO.FileInfo.Exists	System.IO.FileInfo.Exists Property
(System.IO.)FileInfo.Exists	System.IO.FileInfo.Exists	FileInfo.Exists Property

The Link Method Style

Link Method style is used to link a method name to the topic describing that method.

To create a link using this style, select the link text and press **Link Method**  on the Doc-To-Help Documenter for .NET toolbar.

For this style, the Doc-To-Help compiler will apply the nested type separator substitution to the link text, if applicable, see Special Characters in Link Text (page 383), add the word **Method** to the text, and establish a link to the topic with the title coinciding with the resulting text.

By default, Doc-To-Help removes the name of the type containing the method, if the type name is present. You can change this behavior by using the special characters, exclamation sign and parenthesis. See Special Characters in Link Text (page 383). You can also use the argument list for method overloads, see Argument List in Links (page 384).


The following table demonstrates the usage of the Link Method style:

In document	In help file	Links to topic
File.Delete	Delete	File.Delete Method
!File.Delete	File.Delete	File.Delete Method
System.IO.File.Delete	Delete	System.IO.File.Delete Method
!System.IO.File.Delete	System.IO.File.Delete	System.IO.File.Delete Method
(System.IO.)File.Delete	System.IO.File.Delete	File.Delete Method

The Link Constructor Style

Link Constructor style is used to link a constructor name to the topic describing that constructor.

For this style, link text must consist of a type name followed by the word **Constructor**.

To create a link using this style, select the link text and press **Link Constructor**  on the Doc-To-Help Documenter for .NET toolbar.

By default, Doc-To-Help applies the nested type separator substitution to the link text, if applicable, see Special Characters in Link Text (page 383). It also removes qualifying namespace, if present, from the class name in the link text. You can cancel removing the namespace by using the special characters, exclamation sign and parenthesis. See Special Characters in Link Text (page 383). You can also use the argument list for constructor overloads, see Argument List in Links (page 384).


The following table demonstrates the usage of the Link Constructor style:

In document	In help file	Links to topic
FileInfo Constructor	FileInfo Constructor	FileInfo Constructor
System.IO.FileInfo Constructor	FileInfo Constructor	System.IO.FileInfo Constructor
!System.IO.FileInfo Constructor	System.IO.FileInfo Constructor	System.IO.FileInfo Constructor
(System.IO.)FileInfo Constructor	System.IO.FileInfo Constructor	FileInfo Constructor

The Link Operator Style

Link Operator style is used to link an operator name to the topic describing that operator.

For this style, link text must consist of an operator name followed by the word **Operator** or **Conversion** defining the kind of the operator. The operator name must exactly correspond to the operator name in the title of the topic describing the operator.

To create a link using this style, select the link text and press **Link Operator**  on the Doc-To-Help Documenter for .NET toolbar.

By default, Doc-To-Help applies the nested type separator substitution to the link text, if applicable, see Special Characters in Link Text (page 383). It also removes the name of the type containing the operator, if the type name is present. You can change this behavior by using the special characters, exclamation sign and parenthesis. See Special Characters in Link Text (page 383). You can also use the argument list for operator overloads, see Argument List in Links (page 384).

The following table demonstrates the usage of the Link Operator style for regular operators:

In document	In help file	Links to topic
Point.Addition Operator	Addition Operator	Point.Addition Operator
!Point.Addition Operator	Point.Addition Operator	Point.Addition Operator
System.Drawing.Point.Addition Operator	Addition Operator	System.Drawing.Point.Addition Operator
!System.Drawing.Point.Addition Operator	System.Drawing.Point.Addition Operator	System.Drawing.Point.Addition Operator
(System.Drawing.)Point.Addition Operator	System.Drawing.Point.Addition Operator	Point.Addition Operator

The following table demonstrates the usage of the Link Operator style for conversion operators:

In document	In help file	Links to topic
Point.Point to Size Conversion	Point to Size Conversion	Point.Point to Size Conversion

In document	In help file	Links to topic
!Point.Point to Size Conversion	Point.Point to Size Conversion	Point.Point to Size Conversion
System.Drawing.Point.Point to Size Conversion	Point to Size Conversion	System.Drawing.Point.Point to Size Conversion
!System.Drawing.Point.Point to Size Conversion	System.Drawing.Point.Point to Size Conversion	System.Drawing.Point.Point to Size Conversion
(System.Drawing.)Point.Point to Size Conversion	System.Drawing.Point.Point to Size Conversion	Point.Point to Size Conversion

Special Characters in Link Text

There are three special characters that you can use in link texts, in addition to link styles, to control the process of creating links: '!' (exclamation sign), '()' (parenthesis), '+' (plus). These characters have special meanings to the Doc-To-Help compiler (more exactly, to the scripts in the Doc-To-Help project created by Documenter for .NET). They are removed or replaced in the resulting help file.

By default, Doc-To-Help removes the qualifying namespace, if present, from type names in links to types. It also removes the qualifying type name from member names in links to members. So, for example, a link **System.Windows.Forms.CheckBox** formatted with Link Class style will become simply **CheckBox** in the help file. And a link **DataSet.Clone** formatted with Link Method style will become simply **Clone** in the help file. This is done so you can use a namespace or type name to qualify your link so it points to the correct topic. For example, a class **CheckBox** also exists in the namespace **System.Web.UI.WebControls**, and many different classes have a **Clone** method, so you may need to qualify those links to ensure uniqueness, but you usually don't want the qualifier to appear in the help file text.

However, sometimes you may want the fully qualified name to appear in the link text. In that case, use one of the special characters, either exclamation sign or parenthesis.

An exclamation sign '!' at the start of the link text instructs Doc-To-Help to leave the full name as is, not to remove the namespace or type name qualifier. Special character '!' is supported in all link styles except **Link Namespace**, **Link Topic** and **Link Tag** where it is not applicable.

Enclosing the qualifier (including the dot in the end) in parenthesis has a similar effect. It also stops the qualifier from being removed from the link text. But, unlike the exclamation sign, parentheses specify a different link target topic: the link points to a topic whose name does not contain the qualifier.

The following table illustrates the use of special characters '!' and '()' (link text in the document is formatted with Link Method style):

In document	In help file	Links to topic
File.Delete	Delete	File.Delete Method
System.IO.File.Delete	Delete	System.IO.File.Delete Method
!System.IO.File.Delete	System.IO.File.Delete	System.IO.File.Delete Method
(System.IO.)File.Delete	System.IO.File.Delete	File.Delete Method

The special character '+' (plus) is used only in nested type names (nested types are classes, enumerations and other types defined inside a class). It must be used in every nested type name in any link, wherever a type name can occur (that includes links to nested types, links to members of nested types if they are qualified with a type name, and links to method overloads that have nested type names in their argument list). It separates the type name from the parent type name. The normal separator is dot ('.'), and indeed Doc-To-Help replaces '+' with '.' in the help file text, but '+' is necessary to tell Doc-To-Help where the parent type name ends and the type name itself starts.

This behavior, called the *nested type separator substitution*, is supported in all link types except **Link Namespace** and **Link Tag** where it has no meaning.

The following table illustrates the use of the special character '+' for a **SpecialFolder** enumeration nested in the Environment class (all examples are from Microsoft .NET Framework). It also demonstrates the other two special characters, '!' and '()'. Link text in the document is formatted with Link Enumeration style.

In document	In help file	Links to topic
Environment+SpecialFolder	Environment.SpecialFolder	Environment.SpecialFolder Enumeration
System.Environment+SpecialFolder	Environment.SpecialFolder	System.Environment.SpecialFolder Enumeration
!System.Environment+SpecialFolder	System.Environment.SpecialFolder	System.Environment.SpecialFolder Enumeration
(System.)Environment+SpecialFolder	System.Environment.SpecialFolder	Environment.SpecialFolder Enumeration

Argument List in Links

The following link styles can contain an argument list, in case it is necessary to distinguish between overloads with different arguments:

- Link Method
- Link Property
- Link Operator
- Link Constructor

If there are nested types in the argument lists, they undergo nested type separator substitution, see Special Characters in Link Text (page 383,) that is, '+' is replaced with '.'. This is the only transformation applied to types in argument lists. Any types that are not nested remain unchanged. It means that the argument list in the link must exactly correspond to the argument list in the target topic title (except for the '+' sign in nested types, if nested types are present).

Important: Argument lists must be separated with a space from the preceding member name in the link.

The following table illustrates the use of argument lists in List Method links:

In document	In help file	Links to topic
File.Create (String, System.Int32)	Create (String, System.Int32)	File.Create Method (String, System.Int32)
System.IO.File.Create (String, System.Int32)	Create (String, System.Int32)	System.IO.File.Create Method (String, System.Int32)
!System.IO.File.Create (String, System.Int32)	System.IO.File.Create (String, System.Int32)	System.IO.File.Create Method (String, Int32)
(System.IO.)File.Create (String, System.Int32)	System.IO.File.Create (String, System.Int32)	File.Create Method (String, System.Int32)

Link Colors and Appearance

By default, links use the color blue and underline. You can change this, as with many other appearance and formatting style attributes, by modifying the Help target template used by Doc-To-Help. These templates are installed by Doc-To-Help in the Microsoft Word template directory (to locate that directory, use Word menu **Tools | Options | File Locations | User Templates**).

To see what template is used in building a help target, select the target in the Doc-To-Help project and look at the value of the **Template** property in the Help Targets property pane. For example, to modify the color or other appearance attributes of the links using the Link Class style in HTML Help, open the C1H_dotnet_html.dot template in Microsoft Word, find the Link Class style and modify it. You can do this with any style in any help

target template. Here is the list of standard Documenter for .NET target templates for different targets:

- C1H_dotnet_hlp.dot - WinHelp target
- C1H_dotnet_prn.dot - Manual target
- C1H_dotnet_html.dot - HTML-based targets (HTML Help, HTML, Help 2.0, JavaHelp)

Blue underlined links are an MSDN standard, so it was chosen as default in Documenter for .NET. But you can change the colors as described above, or you can make links look like standard links in the corresponding target. To use the standard link appearance instead of the one specified by styles in target templates, set the **AffectsAppearance** property of the link styles to **False**. You can locate that property in Doc-To-Help on the Project icon properties pane.

For HTML-based targets viewed with Internet Explorer (or standalone Microsoft viewers, such as Help 2.0 or HTML Help), standard link colors are determined in Internet Explorer settings **Internet Options | General | Colors | Links**. For WinHelp, the standard link color is green with an underline.

Using Natural Search for Doc-To-Help



ComponentOne Natural Search provides a natural language search capability for both WinHelp (.hlp) and compiled HTML Help (.chm) files. When Doc-To-Help 7.5 Professional builds a help file, Natural Search automatically generates an index file (.smi). When the end-user types a question and initiates a search, the Natural Search runtime checks this index to generate a set of answers that match the question asked.

If Natural Search has been installed, Doc-To-Help 7.5 Professional displays additional properties for the WinHelp, HTML Help, and HTML 4.0 targets that enable and control the search features. By setting the SearchEnabled property to True, Natural Search automatically generates index and configuration files for the project in the same directory as the finished Help file.

Note: Although the HTML 4.0 target exposes the SearchEnabled property, it does not incorporate Natural Search into the user interface of the generated Help system itself. This option is provided for importing HTML pages into the **ComponentOne Response** product.

Natural Search Properties

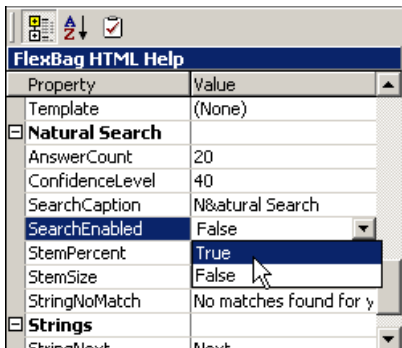
AnswerCount	Determines the maximum number of answers to be returned when a natural language search is performed.
ConfidenceLevel	Sets the minimum score percentage that controls which answers will be displayed when a natural language search is performed.
SearchCaption	Specifies the text added to the tab caption (HTML Help) or button caption (WinHelp) when ComponentOne Natural Search is enabled.
SearchEnabled	Controls whether ComponentOne Natural Search is enabled for a Help target, document, or topic.

StemPercent	Determines the number of characters (as a percentage of the total) that are added to the root word to create a new word group when building natural language search indexes.
StemSize	Determines the number of characters (as a percentage of the total) that are added to the root word to create a new word group when building natural language search indexes.
StringNoMatch	Specifies the text used in the message box when ComponentOne Natural Search does not find a match for a question.

Enabling Natural Search

To enable Natural Search for HTML Help 1.x (.chm) and WinHelp 4.0 (.hlp) targets:

- Select the desired Help target from the Project Editor.
- In the property pane, set the SearchEnabled property to **True**.



Note: Make sure the GenerateProjectFile property is also set to True. If you have the GenerateProjectFile property set to False, you will need to manually edit the .hlp or .hlp file to enable Natural Search. For more information, see [Manually Adding Natural Search Support](#) (page 393).

Defining Your Natural Search Content

In addition to enabling the Natural Search functions, the SearchEnabled property allows you to define which topics and documents you want included in the search index. By default, all documents and topics are included in the Natural Search index.

To exclude a document from the Natural Search index:

- Select the desired Help target from the Project Editor.
- In the Icon Bar, click **Project**.
- From the left pane, select **Documents**.
- Select the document you want to exclude from the right pane.
- In the property pane, set the SearchEnabled property to **False**.

Name	Order	Size	Type	Date
(Browse)				
FlexBag.doc	1	46KB	Microsoft Word D...	02/11/03 10:30:56

Property	Value
Misc	
AdjustLeftIndent	True
Name	FlexBag.doc
Order	1
SplitFile	True
SplitFileWithHHCtrl	False
TextPopups	False
Natural Search	
SearchEnabled	True
	True
	False

SearchEnabled Property

Controls whether ComponentOne Natural Search is enabled for a Help target, document, or topic.

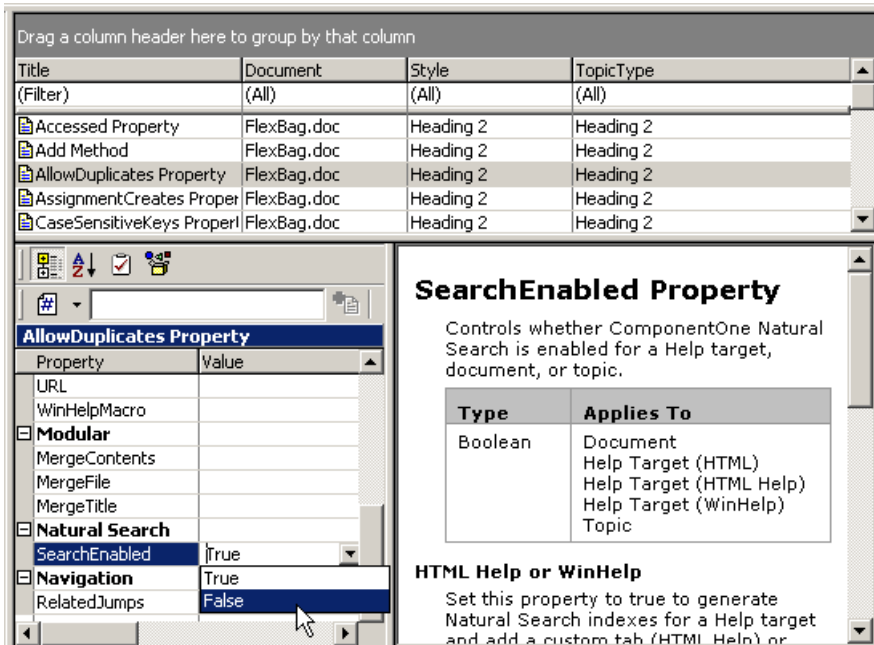
Type	Applies To
Boolean	Document Help Target (HTML) Help Target (HTML Help) Help Target (WinHelp) Topic

HTML Help or WinHelp

Set this property to true to generate Natural Search indexes for a Help target and add a custom tab (HTML Help) or

To exclude a topic from the Natural Search index:

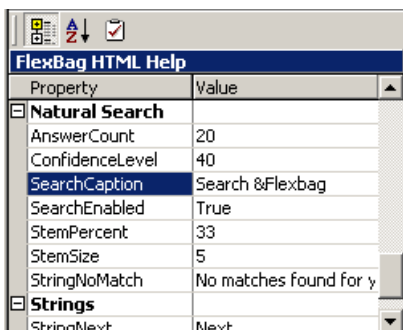
- Select the desired Help target from the Project Editor.
- In the Icon Bar, click **Topics**.
- From the right pane, select the topic you wish to exclude.
- In the property pane, set the SearchEnabled property to **False**.



Modifying the Natural Search Tab Caption

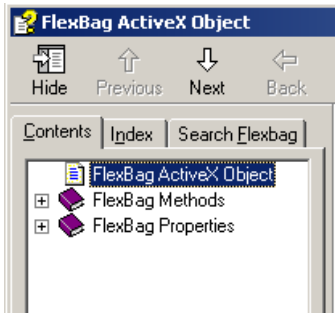
By default, the caption for the Natural Search tab is "Natural Search". You can modify the caption by changing the SearchCaption property.

- Select the desired Help target from the Project Editor.
- In the property pane, select the SearchCaption property and insert the desired text into the right cell.



- Before building your Help target, make sure that the SearchEnabled property is set to True.

After building the Help target, the Natural Search tab displays your SearchCaption property text.

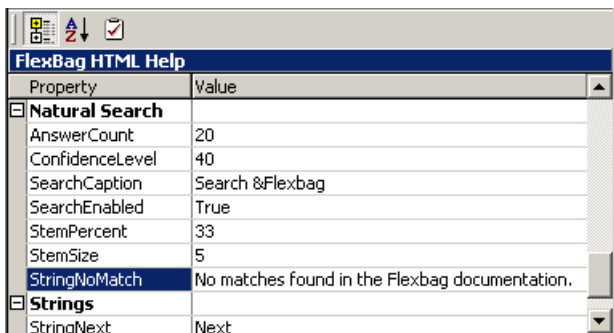


Note: Placing an ampersand in front of any letter in the caption defines that letter as a shortcut key. Your reader can then access the Natural Search tab by simply pressing Alt key plus the defined letter. (In the example above, it would be **Alt+F**.)

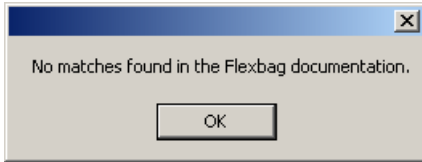
Defining the "No Match" Message

Whenever a user performs a search without a successful match, Natural Search delivers a No Match message. By default, the No Match is "**No matches found for your request.**". You can modify this message by changing the StringNoMatch property.

- Select the desired Help target from the Project Editor.
- In the property pane, select the StringNoMatch property and insert the desired text into the right cell.



After building the Help target, the Natural Search tab displays your StringNoMatch property text when a search is performed with no matches.



Defining the Number of Natural Search Returns

Doc-To-Help Natural Search provides two properties to allow you to define the number of search returns that your user receives. The `AnswerCount` property defines the maximum number of answers that Natural Search is returned to the user. The `ConfidenceLevel` property defines lowest score (in terms of percentage) that is returned to the user.

Defining the Maximum Number of Returns

- Select the desired Help target from the Project Editor.
- In the property pane, set the `AnswerCount` property to whatever number you wish to define.
- By default, this property is set to 20 returns.

Defining the Minimum Score for Returns

- Select the desired Help target from the Project Editor.
- In the property pane, set the `ConfidenceLevel` property to whatever percentage score you wish to define.
- By default this property is set to 40%.

Note: Matching answers that have a lower score than the value of the `ConfidenceLevel` property will be discarded.

Modifying the Search Root Word

Doc-To-Help Natural Search provides two properties to allow you to modify the search root words in the index.

The `Stemsize` property defines the maximum number of characters used as a root word in the natural search index. The default value of this property is 5. Highly technical documentation, where longer search terms are common, may benefit from a longer stem size.

The ConfidenceLevel property defines the number of characters (as a percentage of the total) that are added to the root word to create a new word group when building natural language search indexes. The default value for this property is 33. That is, 1 character is added to the root word for every 3 additional characters in the word. Some international languages, where lengthy suffixes are common, may benefit from a higher percentage.

To modify either of these properties:

- Select the desired Help target from the Project Editor.
- In the property pane, select the appropriate property and set it to the desired number.

Manually Adding Natural Search Support

The HHP file for HTML Help projects and the HPJ file for WinHelp projects can be manually edited to include Natural Search support if the **GenerateProjectFile** option is set to false.

Adding Support to a HTML Help 1.x Target

To manually modify a .HPP file to include Natural Search, you need to make the following modifications:

In the [OPTIONS] section, add the following line:

```
Custom tab="Natural Search", SmartEng.SmartSearchPane
```

In the [WINDOWS] section, modify the line for the main window (or \$global_main) by adding 0x8000 (hexidecimal) to the 10th item in the list and by appending "0,,0,[Natural Search,SmartEng.SmartSearchPane]" to the end of the line.

For example, if the line is

```
$global_main="VSA", "VSA. hhc", "VSA. hhk", , , , , , 0x2120, , 0x3006  
, [200, 100, 1050, 700], 0x0, 0x0, , , 0, ,
```

it should be changed to

```
$global_main="VSA", "VSA. hhc", "VSA. hhk", , , , , , 0xa120, , 0x3006  
, [200, 100, 1050, 700], 0x0, 0x0, , , 0, , 0, [Natural  
Search, SmartEng.SmartSearchPane]
```

Adding Support to a WinHelp Target

To manually modify a .HPJ file to include Natural Search, you need to make the following modifications:

Add the WinHelp macro for the **Natural Search** button using the following code:

```
[CONFIG]
RegisterRoutine('SmaRTEng.dll', 'SmaRTLlookupEx', 'SSSU'): CreateButton('smartBtn', '&atural Search', 'SmaRTLlookupEx(qchPath, ', 'Natural Search', hwndApp)')
```

Altering the Natural Search Noise Word List

The noise word list is a simple text file. Each individual noise word appears on a separate line in the file. The initial list is in alphabetical order but there is no requirement that they be added that way.

The default versions of the configuration and noise word list files are installed in the Natural Search default directory (C:\Program Files\ComponentOne\Natural Search\Default\).

Distributing Natural Search with your Help Files

There are four elements that need to be installed to enable Natural Search to work correctly with your Help file.

For each help target, Natural Search will generate three files: The **Index** (.smi), the **Noise Word List** (.snz), and the **Configuration** file (.cfg). These files will have the same name as the Help file, and should be installed in the same directory as the .chm or .hlp file.

In addition, the Natural Search Engine needs to be installed. Natural Search provides a setup.exe program, located in the "Redist" folder of the Natural Search installation directory (C:\Program Files\ComponentOne\Natural Search\Redist\). This installs the Natural Search runtime (SmaRTEng.dll) and configures the necessary system information.

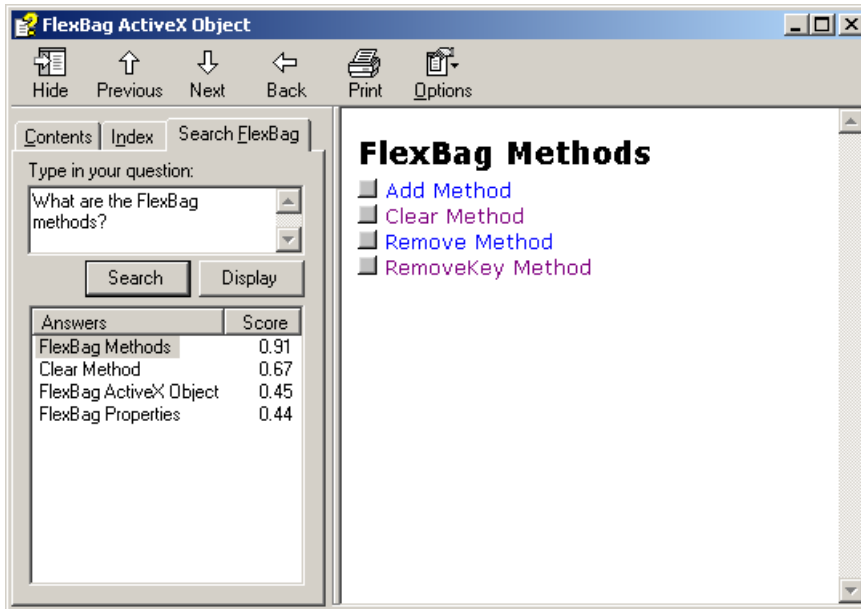
Note: The **Index** (.smi) files for WinHelp and HTML Help 1.x files are not interchangeable and should be installed with their appropriate help target.

Searching a Help File Using Natural Search

If the Natural Search SearchEnabled property was enabled for the Help target, the resulting WinHelp or HTML Help file will include the Natural Search interface for natural language search.

Searching an HTML Help File

Natural Search adds a tab to the navigation pane of the HTML Help window. Selecting this tab displays the search screen where the user can enter a question.

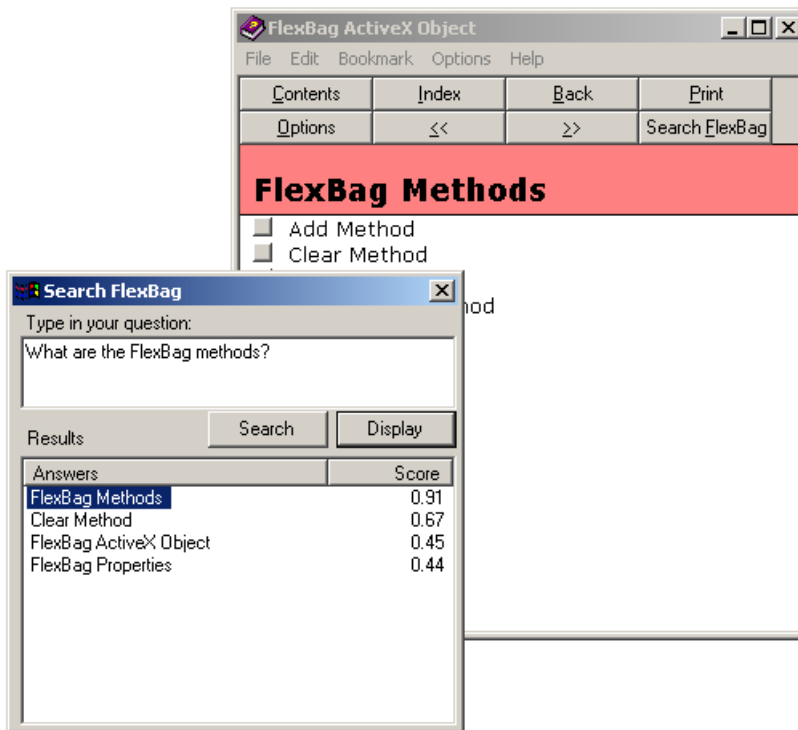


To Search an HTML Help file with Natural Search

- Click the Natural Search tab in the navigation pane.
- Enter a question in the text field at the top of the screen and press **Search**.
- All matching topics are displayed in the **Results** list.
- Double-click a topic and click **Display** to navigate to that topic in the main Help window.

Searching a WinHelp File

Unlike the HTML Help file, the WinHelp interface does not provide additional tabs. Instead, **Natural Search** is added as a navigation button to the top of the help window. Clicking the **Natural Search** button opens a dialog box where the user can enter a question.



To Search a WinHelp file with Natural Search

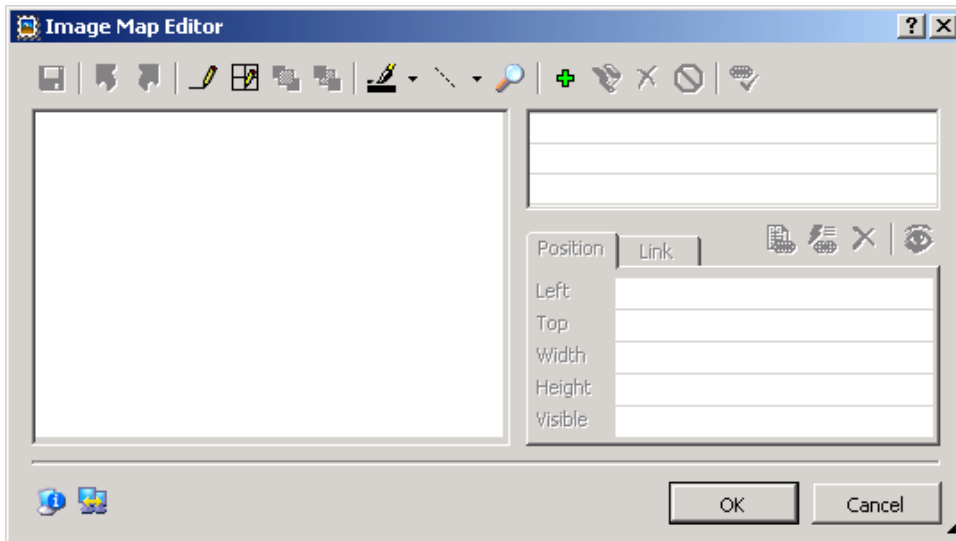
- Click the **Natural Search** button in the WinHelp window.
- A dialog box is displayed.
- Enter the question in the text field at the top of the dialog box and press **Search**.
- All matching topics are displayed in the **Results** list.
- Double-click a topic and click **Display** to navigate to that topic in the main WinHelp window.

Using the Image Map Editor



Image Map Editor

The ComponentOne **Image Map Editor** allows you to place hot spot topic links within a graphic. By simply selecting a graphic and clicking the Image Map Editor icon, you can define any number of hot spots to topics within your Help project.


















In the online Help, the image above is defined as an image map. If you cursor over the tool bar or the lower left side position area, the cursor changes to a hand, signifying a link. If you hold the cursor in that spot, an author defined label opens, giving you information on the link.






If you click on the link, the defined topic replaces the current topic in the Help pane. You can also define the link so that it opens in a separate window.

Exploring the Image Map Toolbar Controls

	Save	Save defined topic links.
	Undo	Undo your last action.
	Redo	Redo your last action.
	Draw (Insert)	Defines the hot spot area.
	Split	Splits the graphic into a defined number of hot spot areas.
	Bring To Front	Brings a hot spot to the front.
	Send To Back	Sends a hot spot to the back.
	Line Color	Defines the line color of the hot spot area.
	Line Style	Defines the line style of the hot spot area.
	Zoom Mode	Zooms in on the image.
	New	Creates a new hot spot.
	Rename	Renames a hot spot.
	Delete	Deletes a hot spot.
	Clear	Clears all hot spots from an image.
	Verify Links	Verifies that the hot spot links are correct.

Exploring the Topic Link Toolbar Controls



	Topic Link	Assigns a topic link to a hot spot.
	Dynamic Link	Assigns a dynamic link to a hot spot.
	Delete Link	Deletes a link from a hot spot.

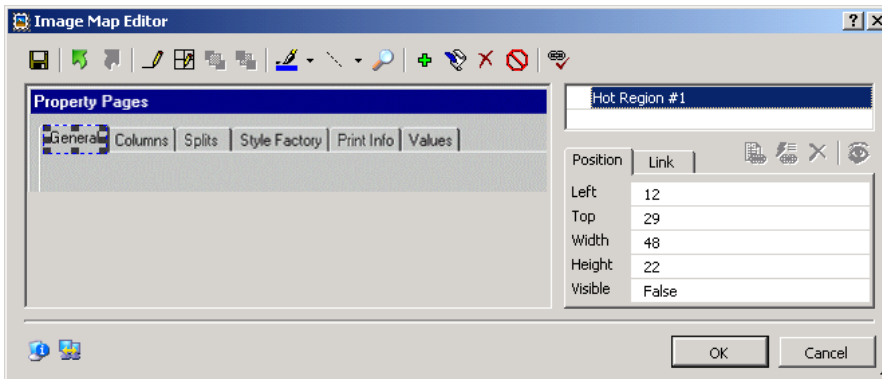
**View Topic**

Views the topic that is linked to the hot spot.

Creating Image Hot Spots

In many cases, it is more effective to create topic links using image hot spots rather than text links. In this way, the user can get information on a specific aspect of an interface or jump to a description of a specific tab by simply clicking on that area of the image in the Help file.

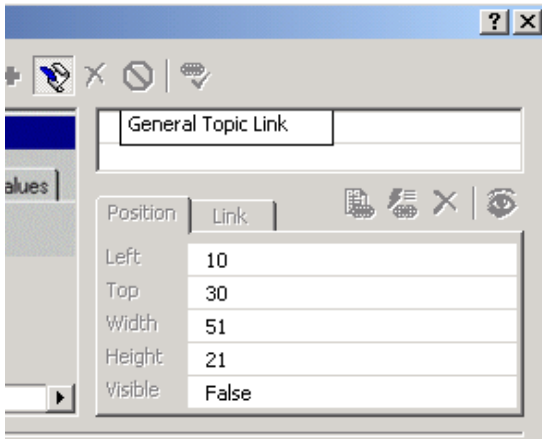
1. Select an image in your source document.
2. Click the **Image Map Editor** icon .
3. Click the **Draw (Insert)** icon .
4. Define the size of your hot spot by dragging the box over the image in the editor.



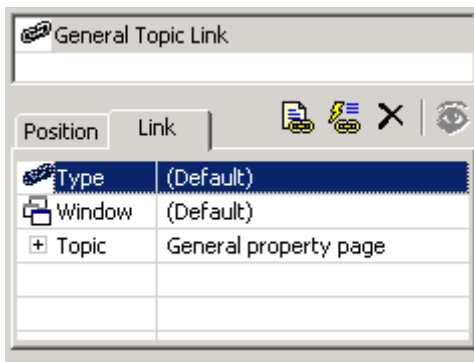
Note that the size properties (Left, Top, Width and Height) have been defined by the size of the box you created.

5. Rename your link by clicking the **Rename** icon  and typing a new name.

This name appears as a label when you cursor over the link area.



6. Select the **Link** tab.
7. Define your topic or dynamic topic by selecting the appropriate icon.



8. Continue to insert additional hot spots as needed.
9. Click **OK** to save your changes and close the Image Map Editor.
10. Build your Help target.

Modifying the Size of an Image Hot Spot

There are two methods for modifying the size of an Image Hot Spot.

Dragging the Hot Spot Box

- Select the image you want to modify and click the **Image Map Editor** icon.
- Select the Hot Spot region box you wish to modify.
- Stretch the box to the size you want.

- Click **OK** to save the changes.

Using the Size and Position Properties

- Select the image you want to modify and click the **Image Map Editor** icon.
- Select the Hot Spot region box you wish to modify.
- On the **Position** tab, modify the **Left**, **Top**, **Width** and **Height** properties.
- Click **OK** to save the changes.

The Modular TOC Utility



One of the major problems with modular HTML Help is that the "child" modules do not contain the entire table of contents. Generally, this limitation is not an obstacle because the "parent" or hub Help file has the full table of contents and, in most cases, this is the Help file that is exposed to the end-user. Where the incomplete TOC can become an issue is during the implementation of "F1" or context sensitive Help. In this case, the child Help file may be called , leaving the end-user with no way to navigate through the entire Help contents.

The ComponentOne **Modular TOC Utility** removes the table of contents limitation by automatically adding the full table of contents to each Help file in a modular project. The Modular TOC Utility is a stand-alone tool that can be used on any set of modular Help files. By simply pointing the utility at each of the component .HHP files a fully functional table of contents is incorporated into each modular Help file.

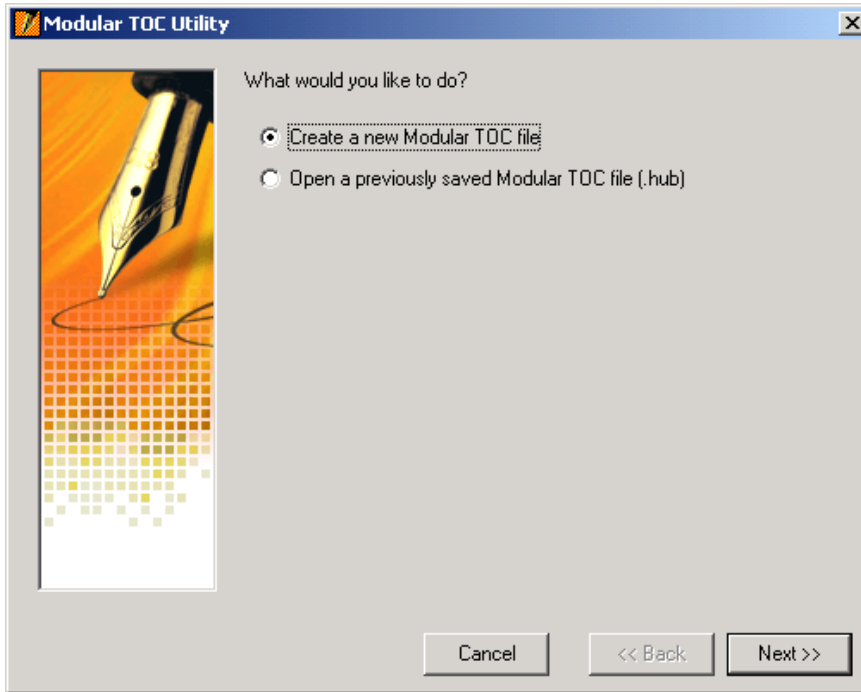
For more information on building a modular Help system, see Using Modular Help (page 317).

Using the Modular TOC Utility

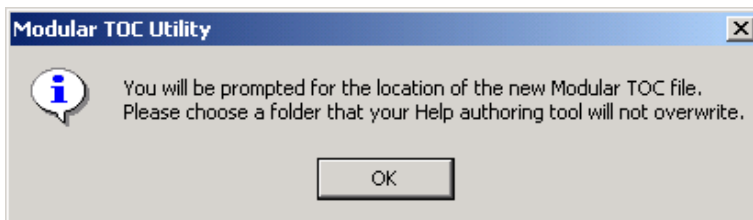
Make sure that you have a fully defined modular Help system before continuing with the following steps.

1. Start the Modular TOC Utility by selecting **Start | ComponentOne | ComponentOne Doc-To-Help 7.5 | Modular TOC Utility**.

The **Modular TOC Utility** opens.



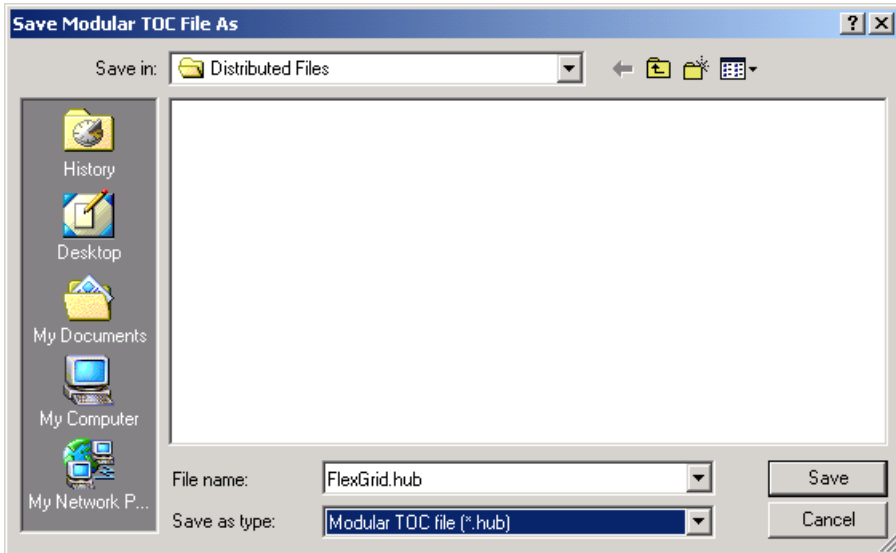
2. Select the Create a new Modular TOC file option and click **Next>>**.
3. After reading the informational dialog, click **OK**.



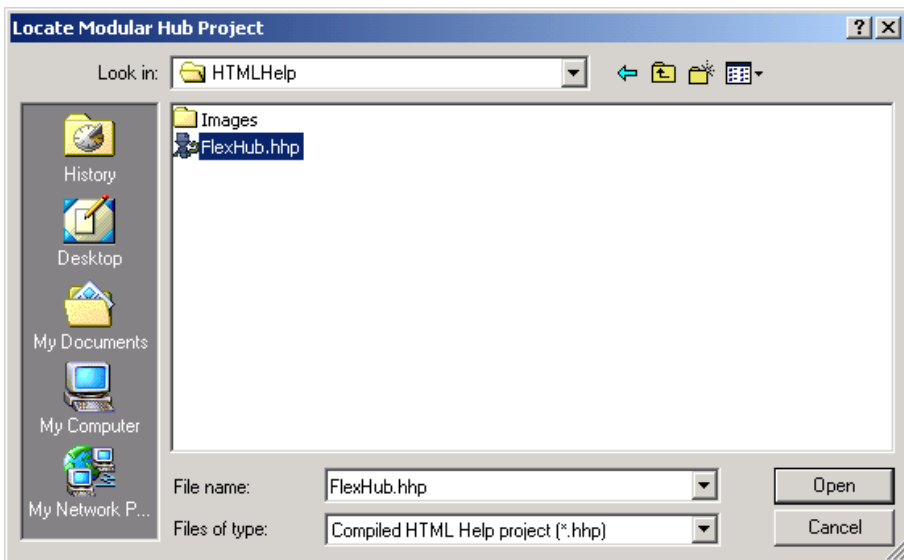
4. Browse to the directory where you want to store your .HUB file.

Note: The Modular TOC Utility will overwrite duplicate .HUB files.

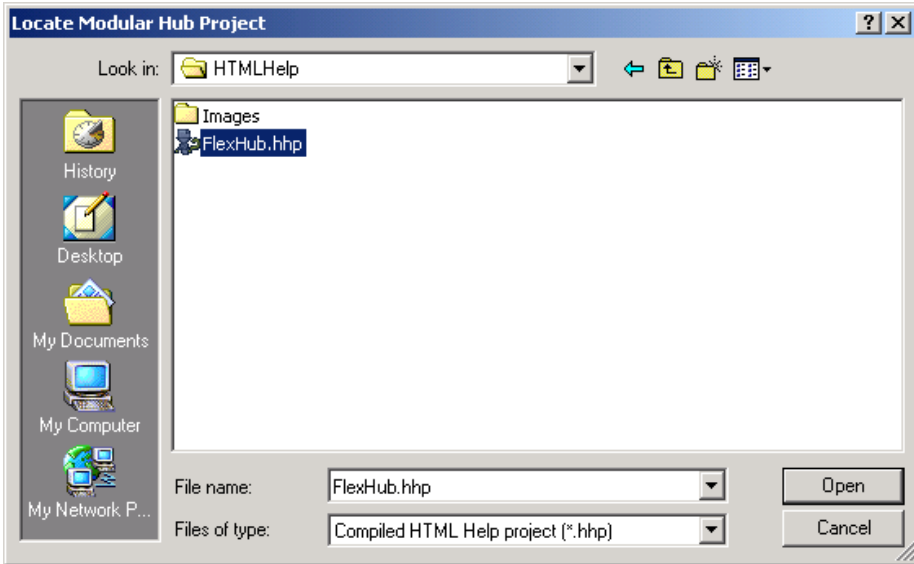
5. Provide the Modular TOC Utility with a .HUB file name and click **Save**.



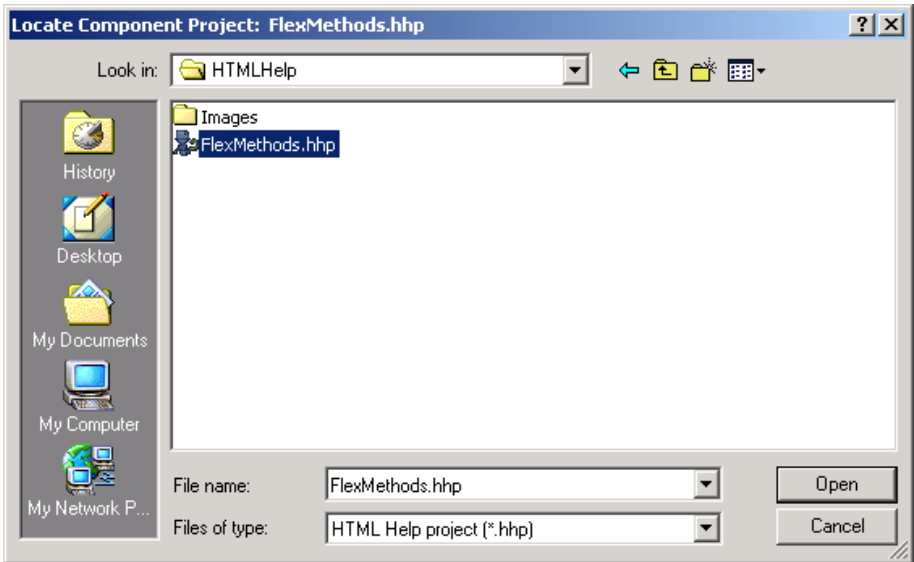
6. The Modular TOC Utility prompts you for the "hub" or parent .HHP file. Browse and select the file, then click **Open**. (This file is normally stored in the HTMLHelp sub-directory of your hub Help file.)



The Modular TOC Utility, lists all "child" or module .HHP files in your modular Help system.

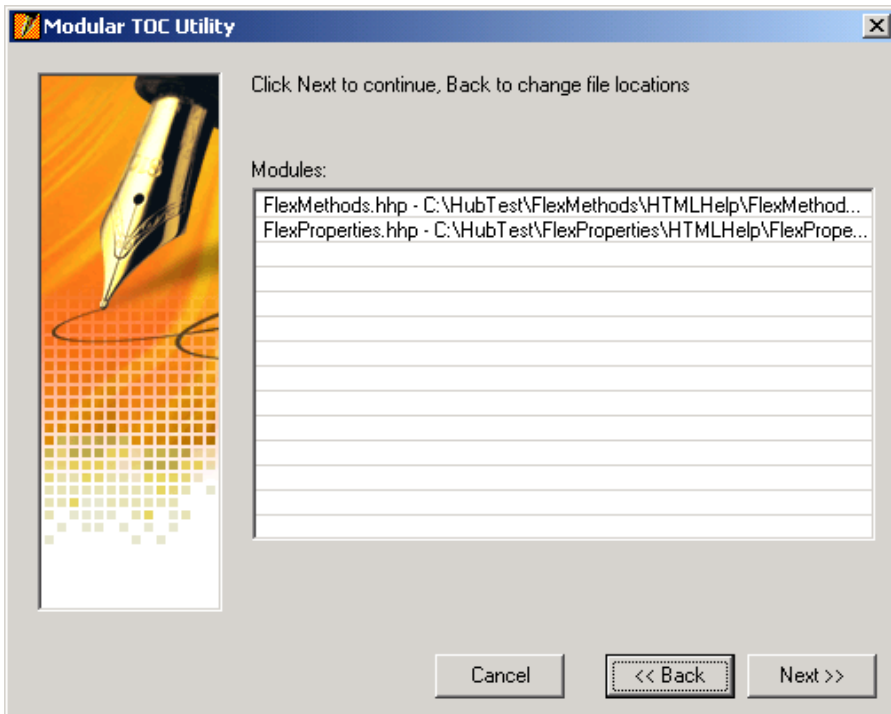


7. Click **Next>>**.
8. The Modular TOC Utility prompts you for the first "child" or module .HHP file. Browse and select the file, then click **Open**. (This file is normally stored in the HTMLHelp sub-directory of your module Help file.)



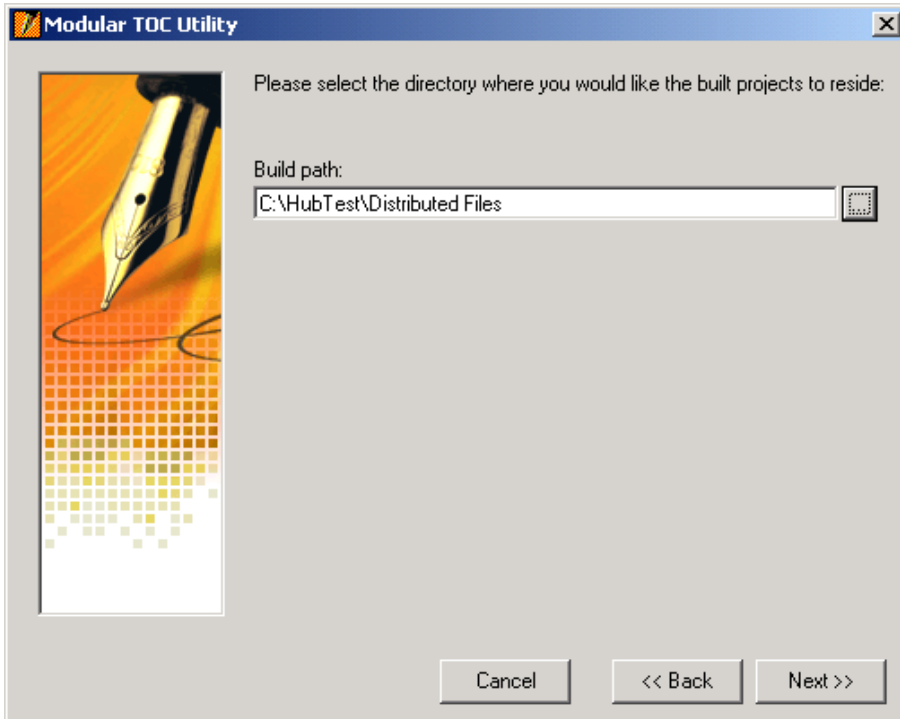
9. Repeat Step 8 until each of the module .HHP files has been located.

The Modular TOC Utility, prompts you to verify that the paths to each .HHP file are correct.



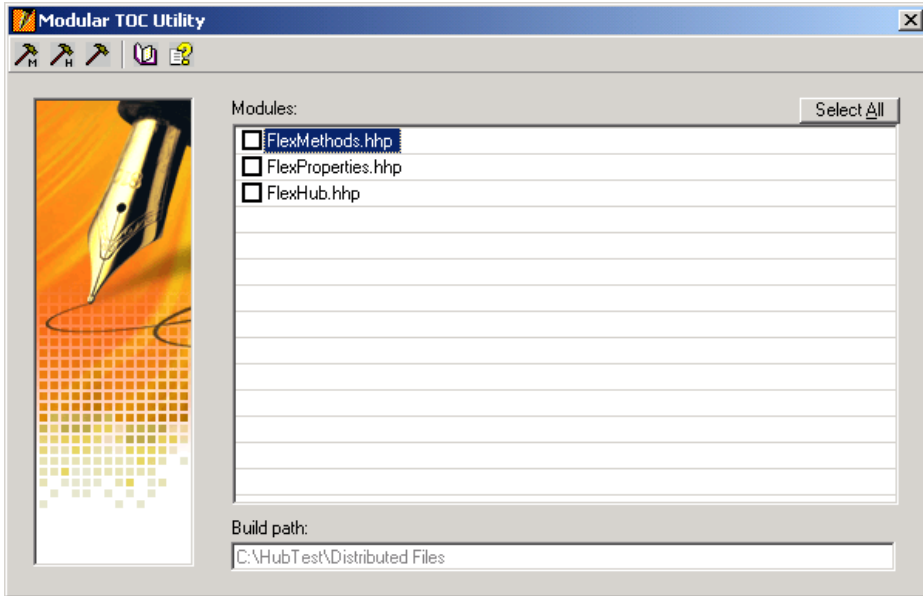
10. Click **Next>>**.

11. Browse To select a directory to store your new modular Help files.



12. Click **Next>>**.
13. Verify the .HUB file and build paths.
14. Click **Finish** to build the new modular Help files.

A number of black text boxes appear as the Modular TOC Utility builds the Help files. After the files are built, the Modular TOC Utility Editor opens.

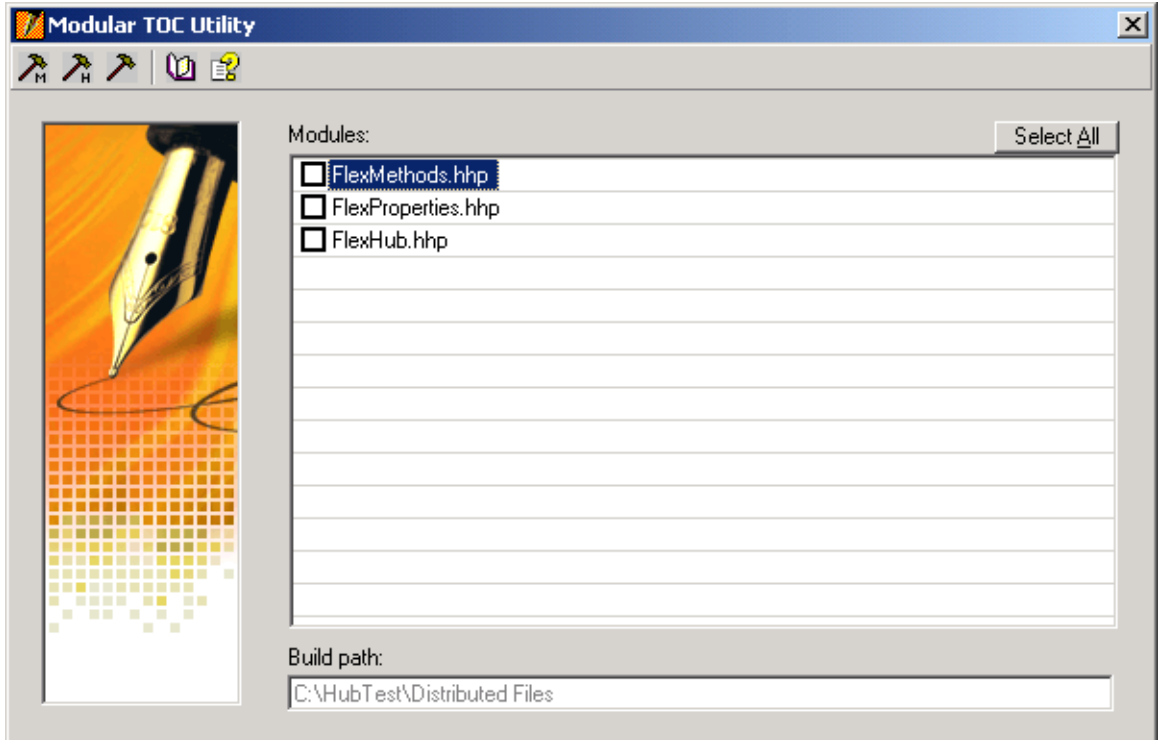


15. For now, close the Modular TOC Utility Editor and examine the Help files in your build directory. Note that all Help files, hub and module, include the entire table of contents.

For more information on using the Modular TOC Utility Editor, see Exploring the Modular TOC Utility Editor (page 409).

Exploring the Modular TOC Utility Editor

You can access the Modular TOC Editor immediately after building new modular Help files or by locating a previously built .HUB file. The editor allows you to compile or view any of the Help files in your modular Help system (by selecting the appropriate checkbox) or view the modular hub file.



Using the Context String Editor



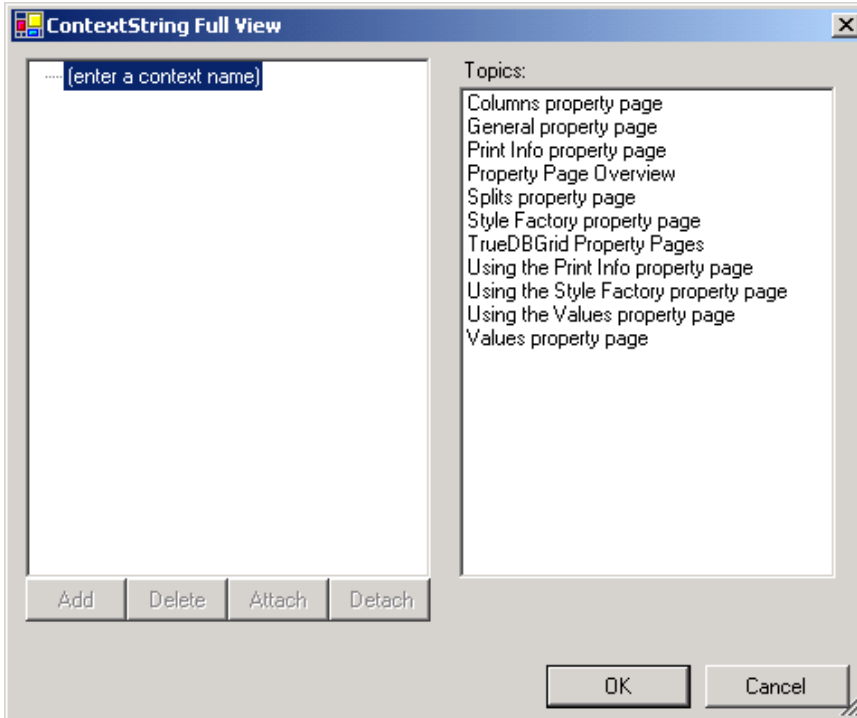
The purpose of the ComponentOne **Context String Editor** is to facilitate defining Microsoft Help 2.0 context strings and linking them to topics. For more information, see MS Help 2.0 Target (page 37).

In Microsoft Help 2.0, unique context strings are used to construct F1 Help. These hierarchical strings replace context ID numbers. An example of a string might be, **C1.C1Zip. C1ZipEntry.FileName**. This string is used to identify the topic that relates to specific .NET property.

Adding Context Strings

The **Context String Editor** allows you add context strings by defining each section of the hierarchical structure.

1. To open the Context String Editor, select **Topics** from the Icon Bar.
2. Select any Topic from the Right Pane.
3. From the Property Pane, select the cell next to the ContextString property and click the arrow.

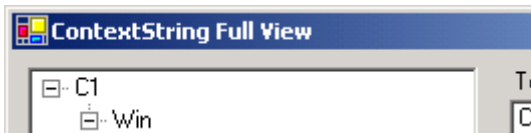


The Context String Editor opens. Note that the context strings are entered in the left pane while the topics in your Help project are displayed in the right pane.

4. In the **enter a context name field**, type in the first part of your context string and press **Enter**.

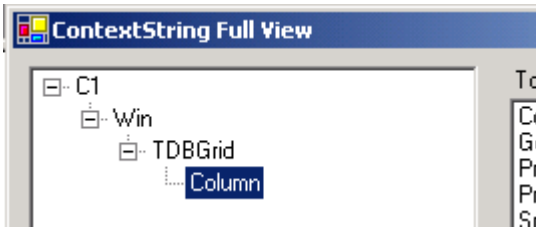


5. Click **Add**.
6. Enter the next section of your context string and press **Enter**.

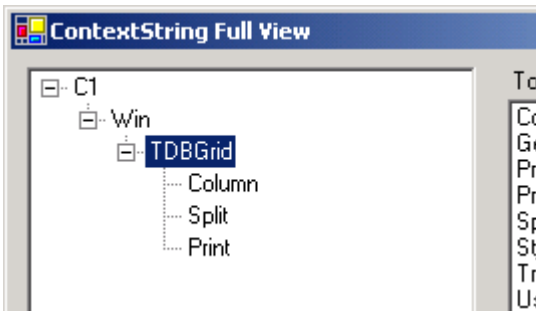


7. Click **Add**.
8. Enter the third section of your context string and press **Enter**.

9. Continue to do this until you have added the last section of the context string.



10. To add another context string, select the preceding section and click **Add**.
11. Enter the last section of the next context string and press **Enter**.
12. Continue this until you have entered all the context strings.

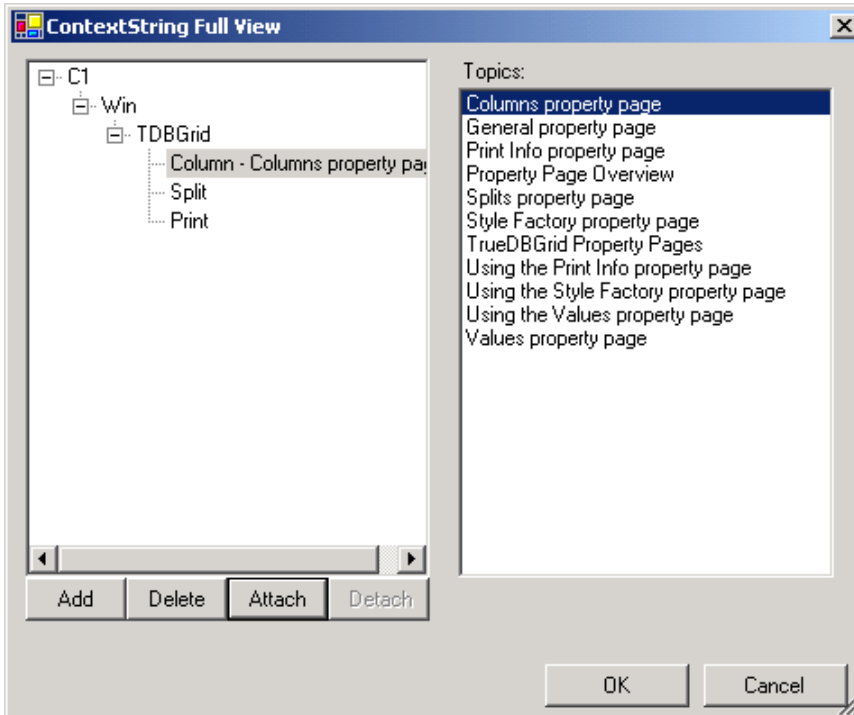


Linking Topics to Context Strings

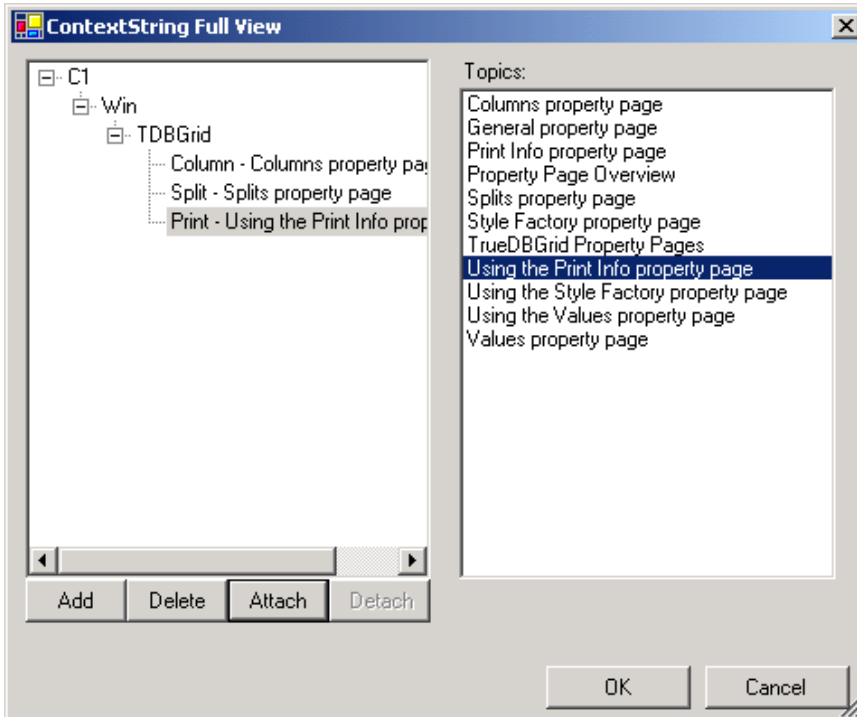
To link a topic to a context string:

Select one of your context strings.

1. Select the corresponding topic from the right pane.
2. Click **Attach**.



3. Continue to do this until you have finished attaching all topics to context strings.



4. Click **OK** when you are finished.

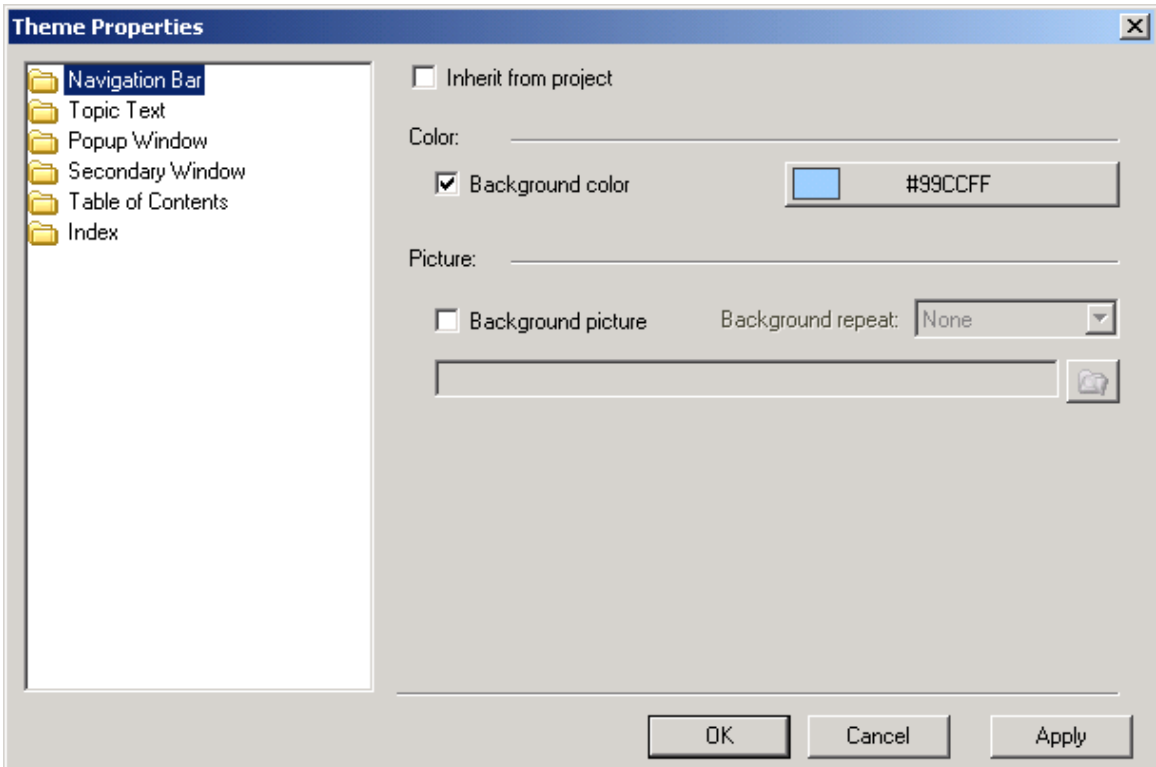
Using the Theme Designer



Theme Designer

Since the original implementation of Doc-To-Help themes, authors have needed a method by which they could modify themes to fit their needs. The ComponentOne **Theme Designer** provides you with the means to develop and save customized themes. The theme designer gives you control over navigation, background styles and image insertion for the any defined window, secondary window or pop-up. These themes can be used for HTML Help, HTML 4.0 and Help 2.0 targets.

The Theme Designer allows you to modify the Navigation Bar, Topic Text, Pop-up Windows and Secondary windows by simply clicking on the item in the left pane.

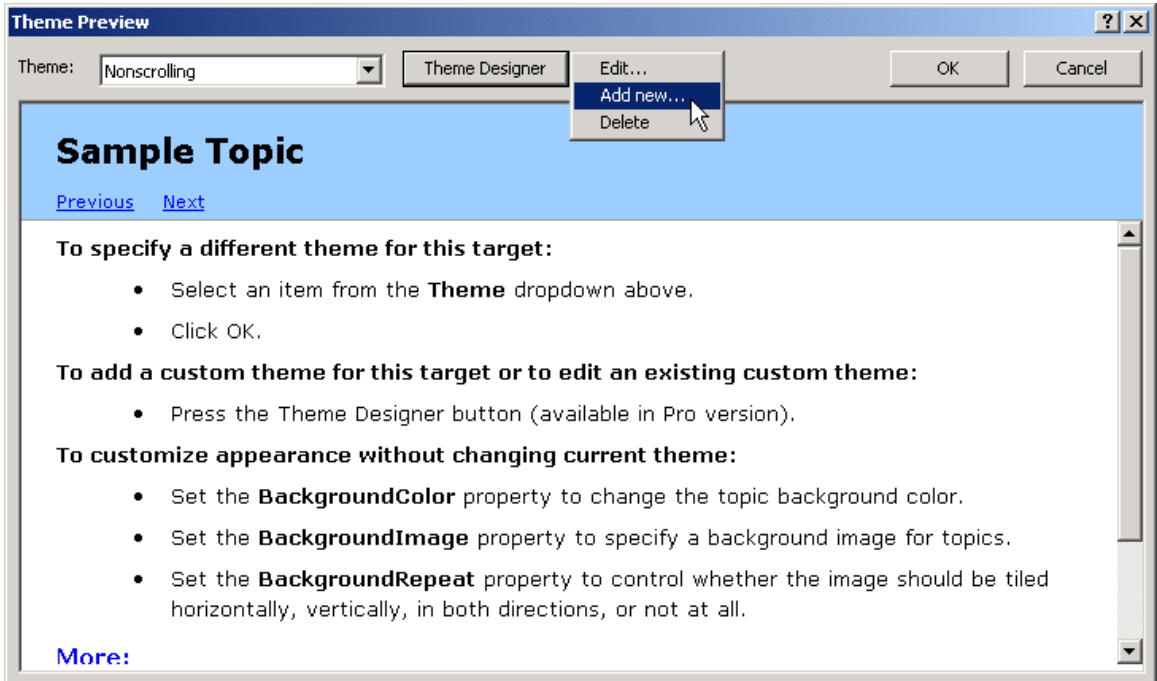


For more information on using themes, see [Using Themes to Modify the Help Window](#) (page 239).

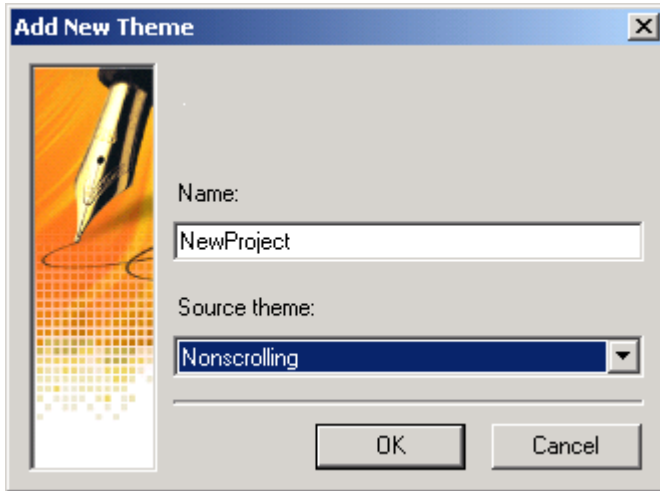
Creating a New Theme

Doc-To-Help comes with a number of pre-formatted themes for HTML Help and HTML. These pre-formatted themes cannot be modified. You can develop your own themes based upon the pre-formatted themes by following the steps below.

- Select the Help Target that you wish to modify.
- From the property pane, select the Theme property and click the button in the box to the right of the property name.
- Click the **Theme Designer** button and select **Add new** from the menu.



- The **Add New Theme** dialog opens.
- Enter the name of the new theme in the **Name** textbox.
- Select the pre-formatted theme you want to use as your source from the **Source Theme** dropdown.
- Click **OK**.

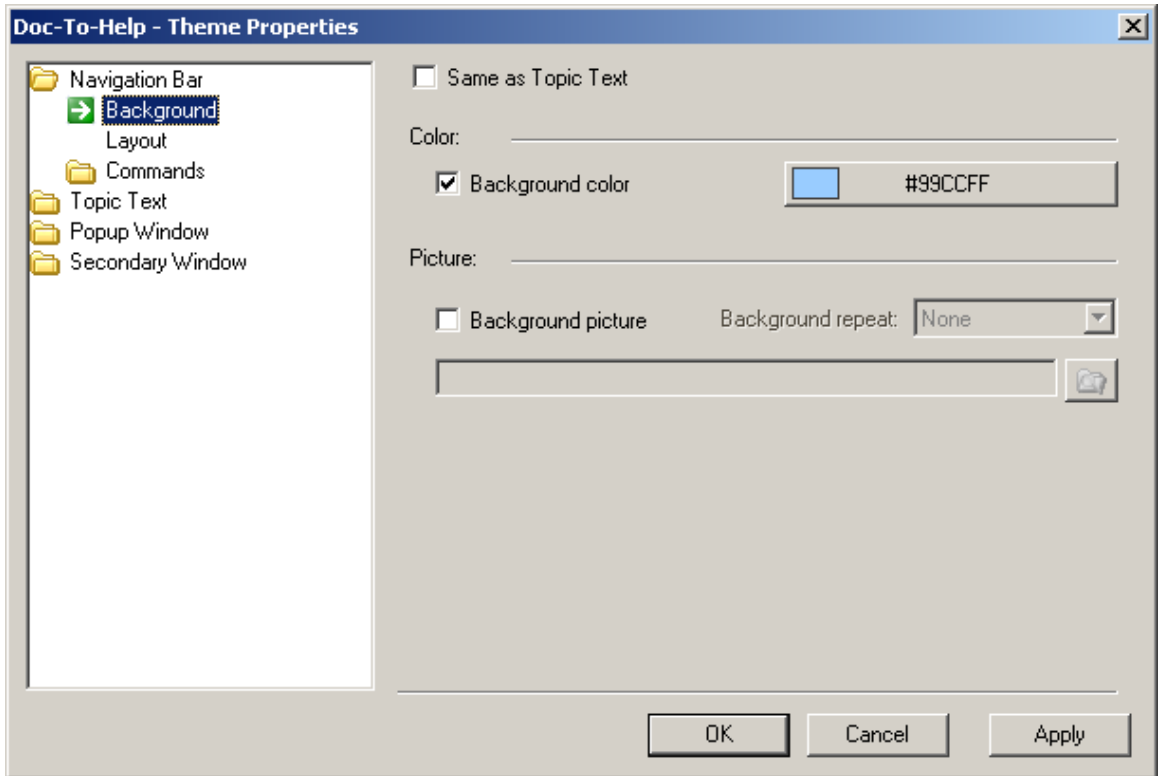



Doc-To-Help creates your new theme, and opens the theme editor.

Exploring the Navigation Bar

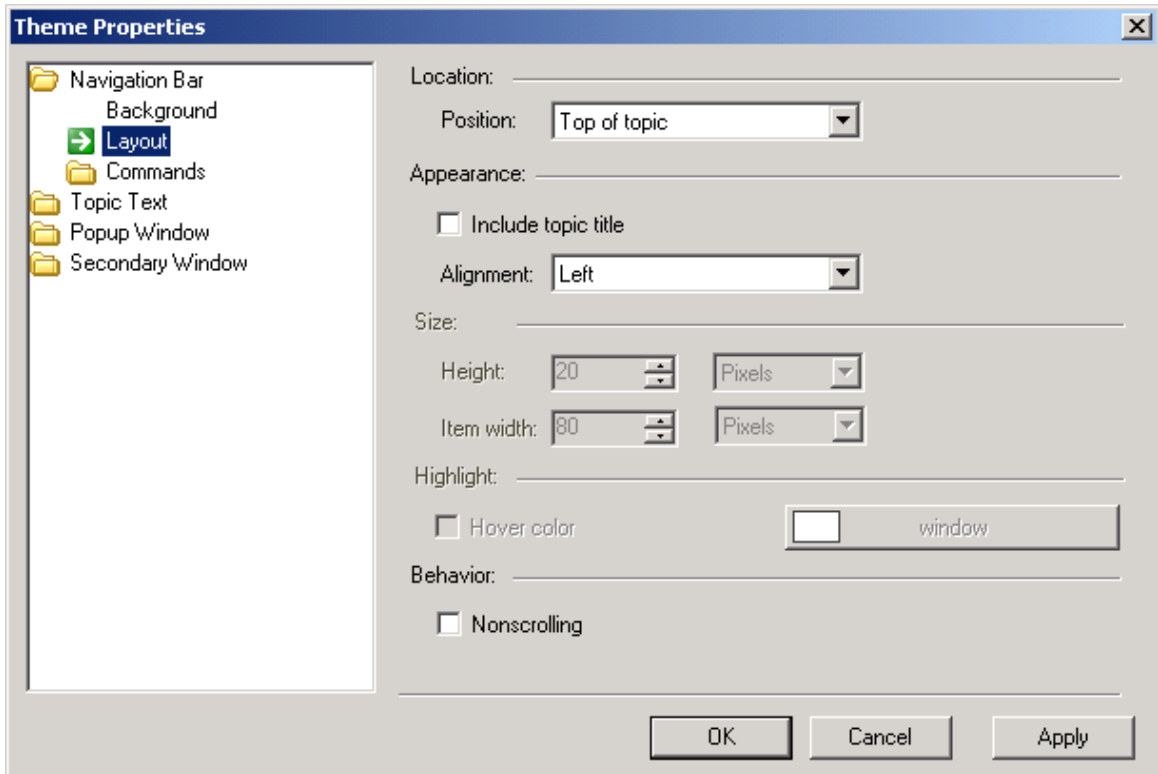
The Navigation bar area includes sections that allow you to define the background, layout and next and previous commands.

Background Controls



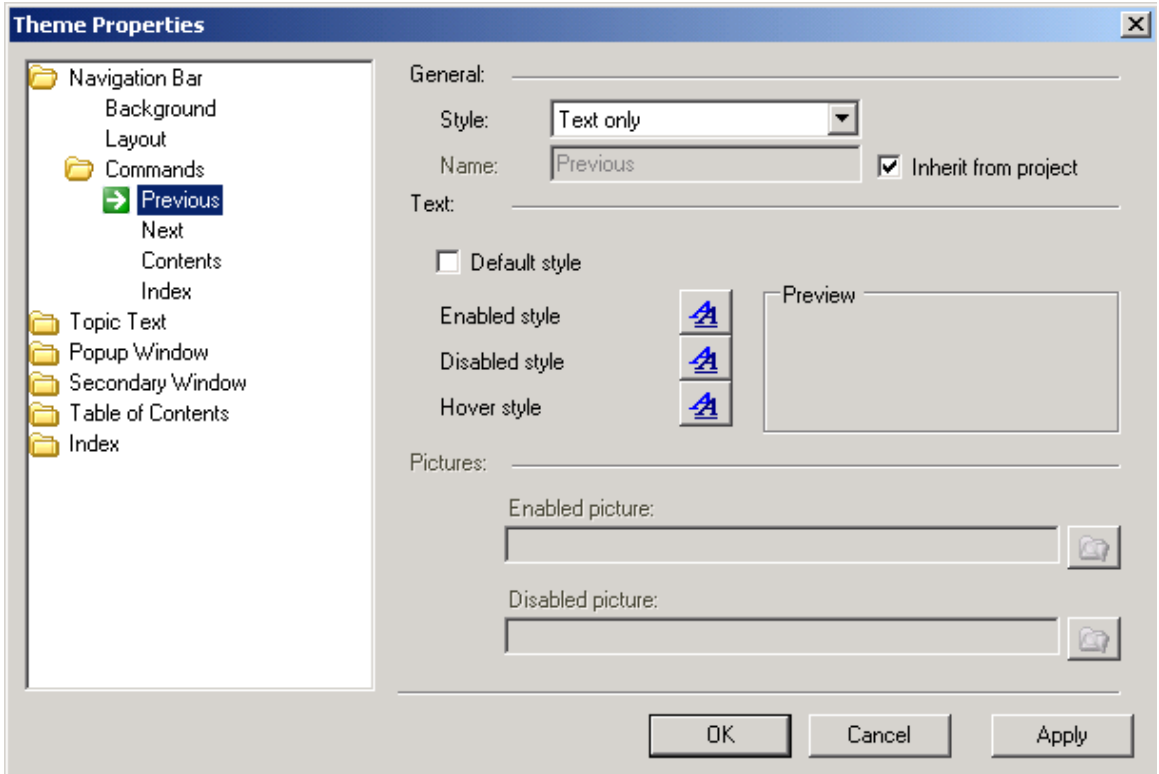
- Inherit from project** Sets the theme properties so they are inherited from the project settings.
- Background color** Defines the color in the navigation bar area. Click the button to the right to access color palettes.
- Background picture** Defines the background image in the navigation bar area. Click the browse button  to access the image directory. Note: ComponentOne suggests that any images you want to use should be stored in the following folder:
ComponentOne\DocToHelp7\Themes\Images\Backgrounds
- Background repeat** Defines where the image will appear in the background.

The Layout Controls



Position	Defines the location of the navigation area.
Include topic title	Determines if the topic title is included in the navigation area.
Alignment	Defines the position of the navigation elements.
Size	Defines the height and width of the navigation area. Note: The size attributes are only available in HTML 4.0 targets.
Hover color	Defines the hover color for the navigation links. Click the button to the right to access color palettes.
Nonscrolling	Determines if the navigational area is nonscrolling.

The Previous Commands



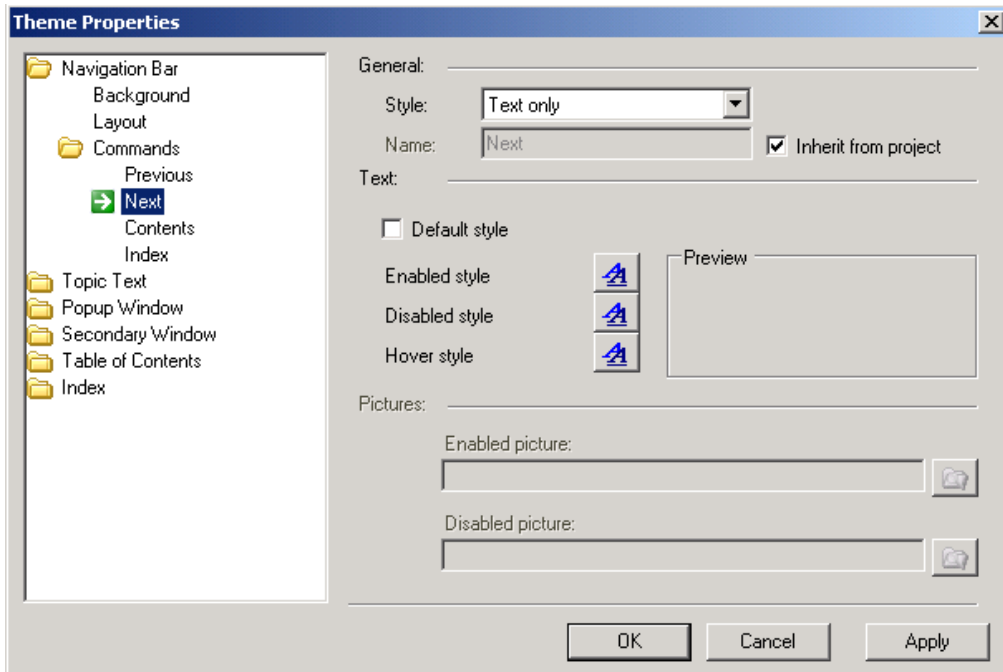
Style	Defines the Previous navigation as text, image or text and image.
Name	Defines text for the Previous link.
Inherit from project	Determines if the Previous link text is inherited from the project.
Default style	Determines if the Previous link style is inherited from the project.
Enabled style	Defines the enabled text style for the link.
Disabled style	Defines the disabled text style for the link.
Hover style	Defines the hover text style for the link.

Enabled picture

Defines the enabled image for the link. Note: ComponentOne suggests that any images you want to use should be stored in the following folder: ComponentOne\DocToHelp7\Themes\Images\Buttons and Icons

Disabled picture

Defines the disabled image for the link. Note: ComponentOne suggests that any images you want to use should be stored in the following folder: ComponentOne\DocToHelp7\Themes\Images\Buttons and Icons

The Next Commands**Style**

Defines the Next navigation as text, image or text and image.

Name

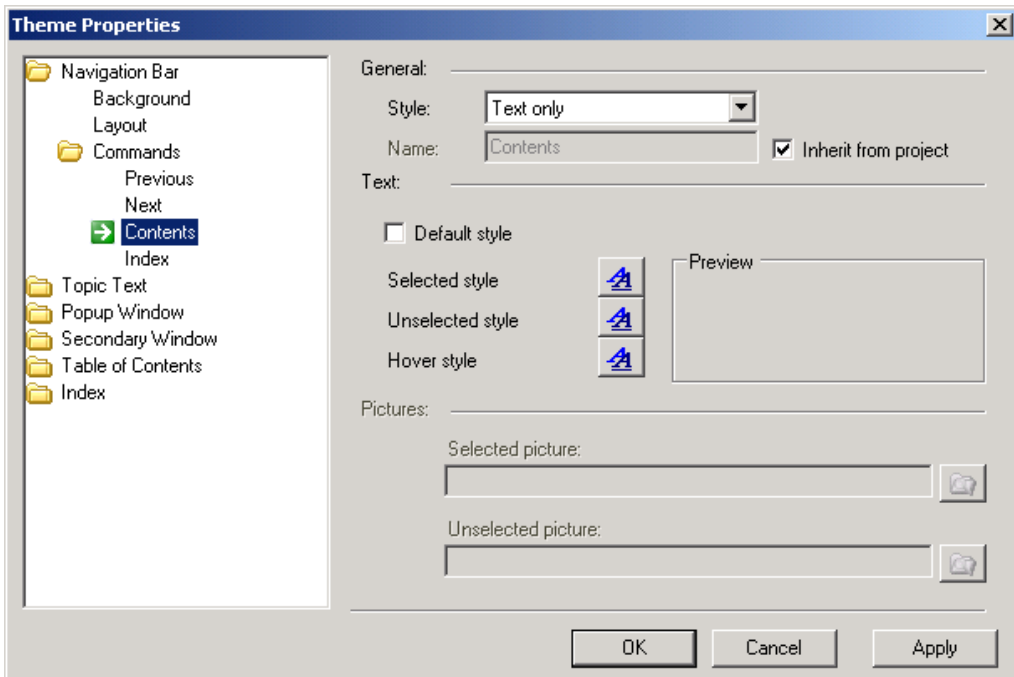
Defines text for the Next link.

Inherit from project

Determines if the Next link text is inherited from the project.

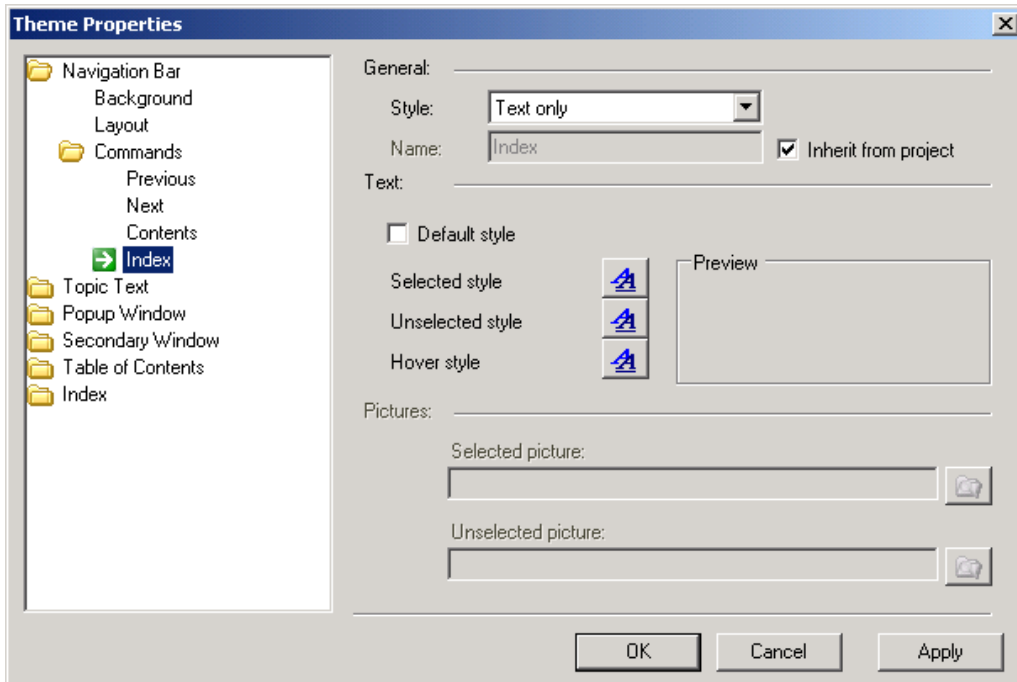
Default style	Determines if the Next link style is inherited from the project.
Enabled style	Defines the enabled text style for the link.
Disabled style	Defines the disabled text style for the link.
Hover style	Defines the hover text style for the link.
Enabled picture	Defines the enabled image for the link. Note: ComponentOne suggests that any images you want to use should be stored in the following folder: ComponentOne\DocToHelp7\Themes\Images\Buttons and Icons
Disabled picture	Defines the disabled image for the link. Note: ComponentOne suggests that any images you want to use should be stored in the following folder: ComponentOne\DocToHelp7\Themes\Images\Buttons and Icons

The Contents Commands (HTML 4.0 Only)



Style	Defines the Contents navigation as text, image or text and image.
Name	Defines text for the Contents link.
Inherit from project	Determines if the Contents link text is inherited from the project.
Default style	Determines if the Contents link style is inherited from the project.
Enabled style	Defines the enabled text style for the link.
Disabled style	Defines the disabled text style for the link.
Hover style	Defines the hover text style for the link.
Enabled picture	Defines the enabled image for the link. Note: ComponentOne suggests that any images you want to use should be stored in the following folder: ComponentOne\DocToHelp7\Themes\Images\Buttons and Icons
Disabled picture	Defines the disabled image for the link. Note: ComponentOne suggests that any images you want to use should be stored in the following folder: ComponentOne\DocToHelp7\Themes\Images\Buttons and Icons

The Index Commands (HTML 4.0 Only)



Style	Defines the Index navigation as text, image or text and image.
Name	Defines text for the Index link.
Inherit from project	Determines if the Index link text is inherited from the project.
Default style	Determines if the Index link style is inherited from the project.
Enabled style	Defines the enabled text style for the link.
Disabled style	Defines the disabled text style for the link.
Hover style	Defines the hover text style for the link.
Enabled picture	Defines the enabled image for the link. Note: ComponentOne suggests that any images you want to use should be stored in the following folder: ComponentOne\DocToHelp7\Themes\Images\Buttons and Icons

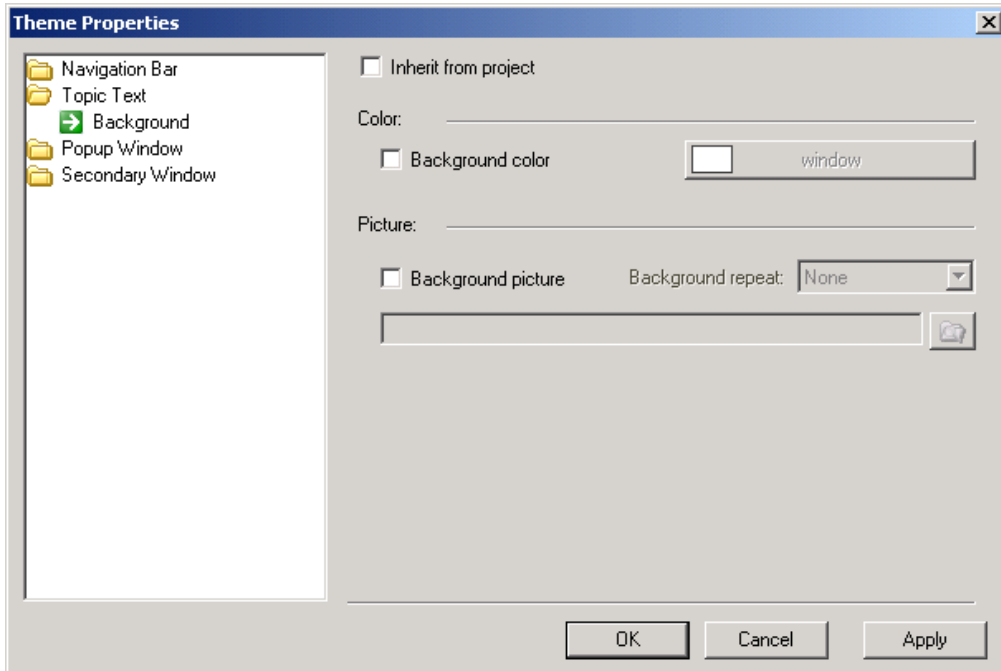
Disabled picture

Defines the disabled image for the link. Note: ComponentOne suggests that any images you want to use should be stored in the following folder:
 ComponentOne\DocToHelp7\Themes\Images\Buttons
 and Icons

Exploring the Topic Text Bar

The Topic Text bar area allows you to modify the formatting of the contents pane of your Help target.


Background Controls

**Inherit from project**

Sets the theme properties so they are inherited from the project settings.

Background color

Defines the color in the contents area. Click the button to the right to access color palettes.

- Background picture** Defines the background image in the contents area. Click the browse button  to access the image directory. Note: ComponentOne suggests that any images you want to use should be stored in the following folder: ComponentOne\DocToHelp7\Themes\Images\Backgrounds
- Background repeat** Defines where the image will appear in the background.


Exploring the Popup Window Bar

The Popup Window bar area allows you to modify the formatting of the popup windows of your Help target

Background Controls



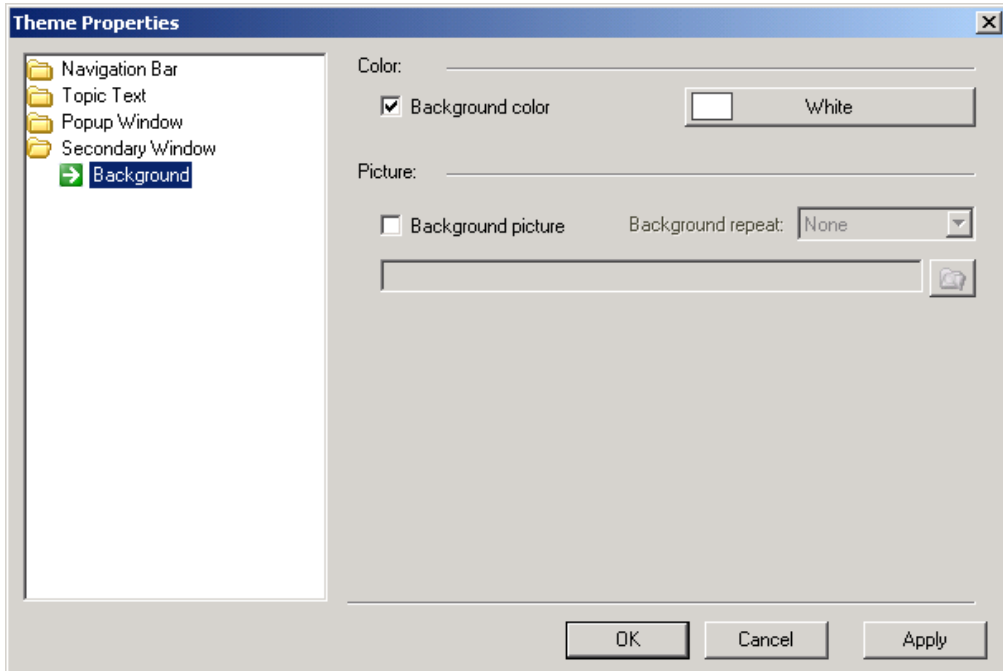
- Background color** Defines the color in the popup. Click the button to the right to access color palettes.

- Background picture** Defines the background image in the popup. Click the browse button  to access the image directory. Note: ComponentOne suggests that any images you want to use should be stored in the following folder:
ComponentOne\DocToHelp7\Themes\Images\Backgrounds
- Background repeat** Defines where the image will appear in the background.


Exploring the Secondary Window Bar

The Secondary Window bar area allows you to modify the formatting of the secondary windows of your Help target.

Background Controls



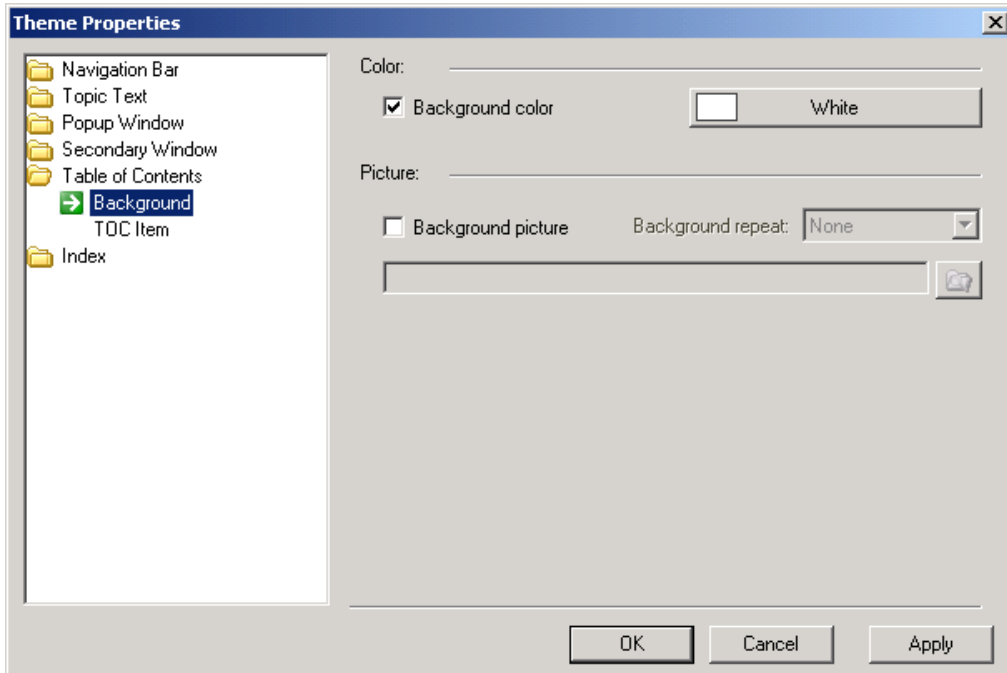
- Background color** Defines the color in the secondary windows. Click the button to the right to access color palettes.

- Background picture** Defines the background image in the secondary windows. Click the browse button  to access the image directory. Note: ComponentOne suggests that any images you want to use should be stored in the following folder:
ComponentOne\DocToHelp7\Themes\Images\Backgrounds
- Background repeat** Defines where the image will appear in the background.


Exploring the Table of Contents Bar (HTML 4.0 Only)

The Table of Contents bar area allows you to modify the background and text formatting of the table of contents area of your HTML 4.0 target.

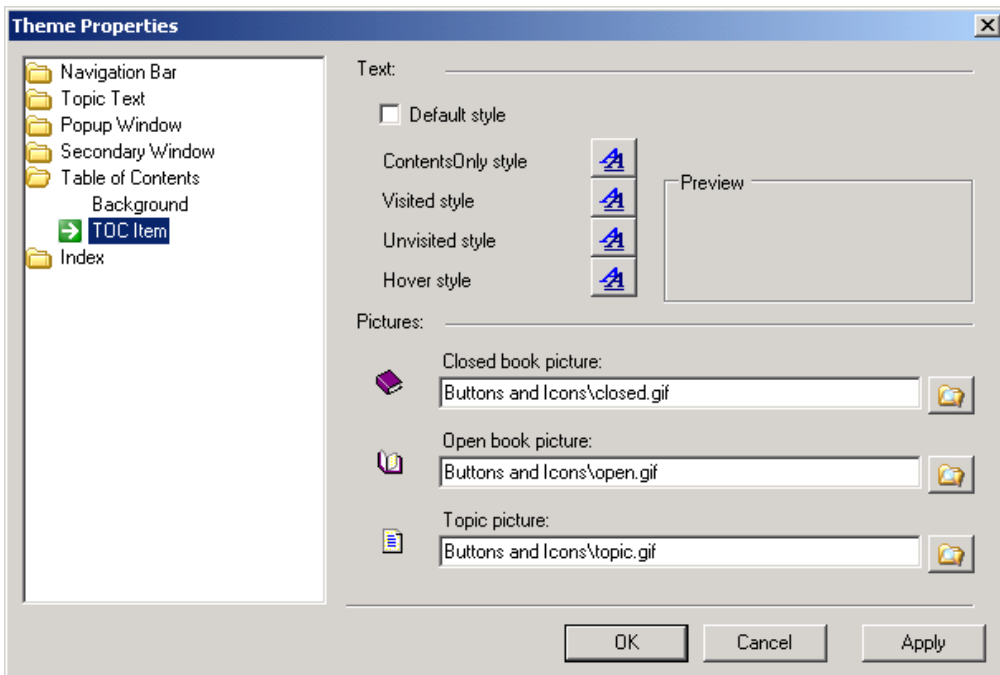
Background Controls (HTML 4.0 Only)



- Background color** Defines the color in the table of contents area. Click the button to the right to access color palettes.

- Background picture** Defines the background image in the table of contents area. Click the browse button  to access the image directory. Note: ComponentOne suggests that any images you want to use should be stored in the following folder:
ComponentOne\DocToHelp7\Themes\Images\Backgrounds
- Background repeat** Defines where the image will appear in the background.

The TOC Item Commands (HTML 4.0 Only)



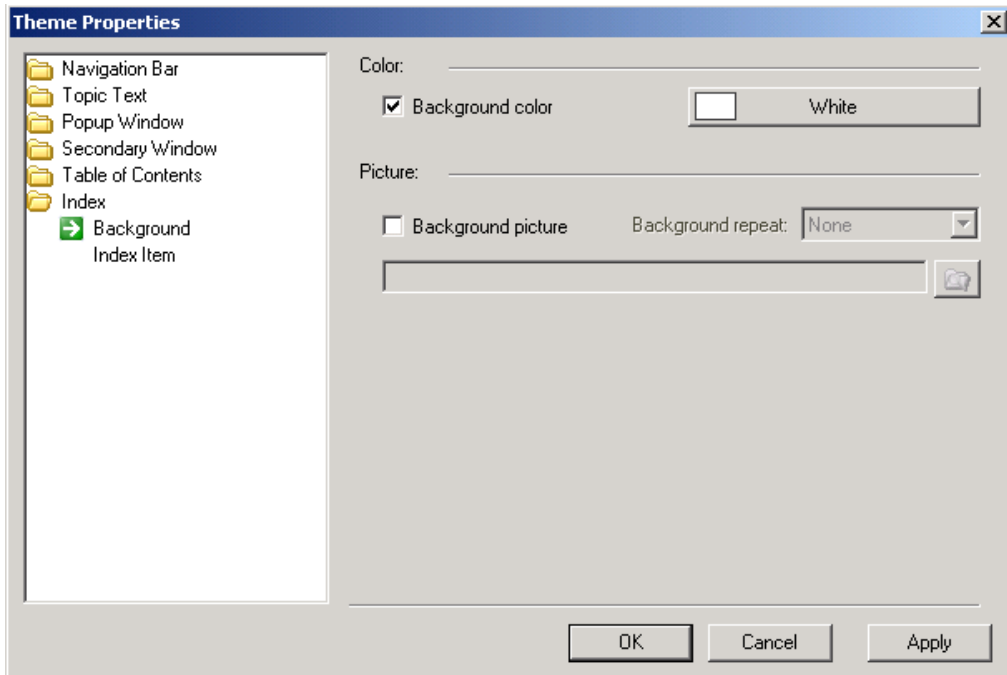
- Default style** Determines if the TOC style is inherited from the project.
- ContentsOnly style** Defines the text style for the TOC items defined as Contents Only.
- Visited style** Defines the text style for TOC items that have been visited.
- Unvisited style** Defines the text style for the TOC items that have not been visited.
- Hover style** Defines the hover text style for the TOC items.

Closed book picture	Defines the closed image for the TOC. Note: ComponentOne suggests that any images you want to use should be stored in the following folder: ComponentOne\DocToHelp7\Themes\Images\Buttons and Icons
Open book picture	Defines the opened image for the TOC. Note: ComponentOne suggests that any images you want to use should be stored in the following folder: ComponentOne\DocToHelp7\Themes\Images\Buttons and Icons
Topic picture	Defines the topic image for the TOC. Note: ComponentOne suggests that any images you want to use should be stored in the following folder: ComponentOne\DocToHelp7\Themes\Images\Buttons and Icons

Exploring the Index Bar (HTML 4.0 Only)

The Index bar area allows you to modify the background and text formatting of the index area of your HTML 4.0 target.

Background Controls (HTML 4.0 Only)




Background color

Defines the color in the index area. Click the button to the right to access color palettes.

Background picture

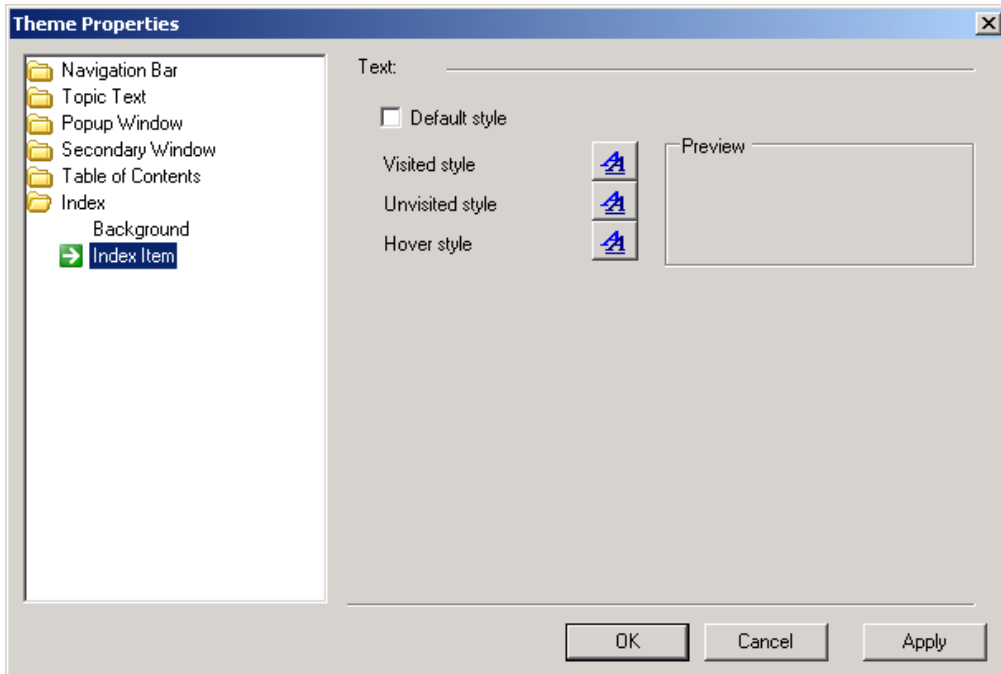
Defines the background image in the index area. Click the

browse button  to access the image directory. Note: ComponentOne suggests that any images you want to use should be stored in the following folder:
ComponentOne\DocToHelp7\Themes\Images\Backgrounds

Background repeat

Defines where the image will appear in the background.

The Index Item Commands (HTML 4.0 Only)



Default style	Determines if the Index area style is inherited from the project.
Visited style	Defines the text style for Index items that have been visited.
Unvisited style	Defines the text style for the Index items that have not been visited.
Hover style	Defines the hover text style for the Index items.

Including Custom HTML Content in Themes

In Doc-To-Help 7.5 you can include additional arbitrary HTML content in your themes. For example, to include a company logo or a link in a theme, prepare an HTML file (only the body, without <BODY> tags and anything outside them) and include it in the theme by selecting an appropriate Customize node and Position in Theme Designer. Specify the include file there.

What is an Include File?

An include file is either an image file or an HTML file. You can have multiple include files in a single theme. The result of the inclusion is the *.theme files that are used to build the help target. For an HTML include file, only its body, the content that is normally placed inside the <BODY> tags should be present in the file. The <BODY> tags themselves and everything outside (<HTML> tags, <HEAD> tags, etc) should be removed from the include file. Doc-To-Help does not verify this or change the included content. The inclusion is performed literally, the whole contents of the file is included.

Restrictions and Guidelines

Include HTML files and ancillary files (images, links) that are referenced in the include files must reside in the Customize sub-directory of the theme. When you include a file, it will be automatically copied to that directory. If you have ancillary files referenced in that file, you are responsible for putting them in the same directory with the include file. All paths to such ancillary files in HTML must be relative paths to the Customize directory, as in the following example of an HTML include file showing a logo image:

```

```

This and other restrictions on the content of HTML include file are the author's responsibility. Doc-To-Help does not change or verify the include files. Since everything outside <BODY> tags (and the <BODY> tags themselves) must be removed, the author of an include file should put everything that is needed inside the body, including styles and scripts.

Only use unique names of styles, scripts, etc. These names must not conflict with the names used by Doc-To-Help for other purposes. As a recommendation, it is better to avoid using names in include files altogether. Use only inline unnamed constructs.

Using the Topic Tools



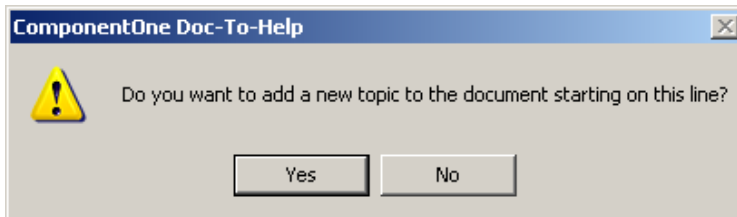
The Doc-To-Help toolbar includes three buttons to make adding and defining topics a simple process. The **Add Topic**, **Rename Topic** and **Delete Topic** buttons give you complete control over all the topics in your project. Simply select the location for the new topic, click the **Add Topic** button and type the topic text, to create a fully functional topic in your project. More importantly, all the features related to a topic (adding topic links, cross-references, etc.) can be accessed without first building your project. This eliminates the need for time consuming multiple builds.

Using the Add Topic Button

Adding a topic at any point in your source document with the **Add Topic** button is a simple procedure.

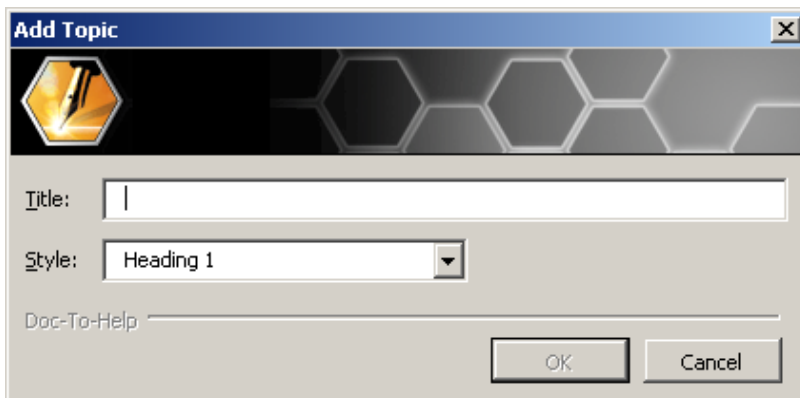
- Click the **Add Topic** button .

Doc-to-Help prompts you with a dialog box asking if you want to add a new topic on the line you have selected.

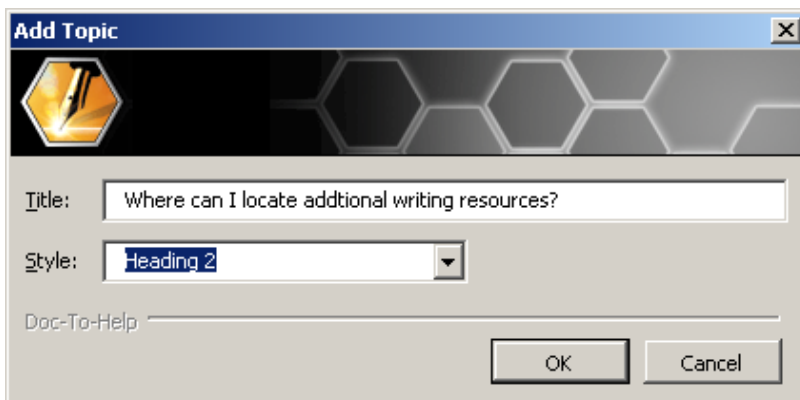


- Click **Yes**.

The **Add Topic** dialog box opens.

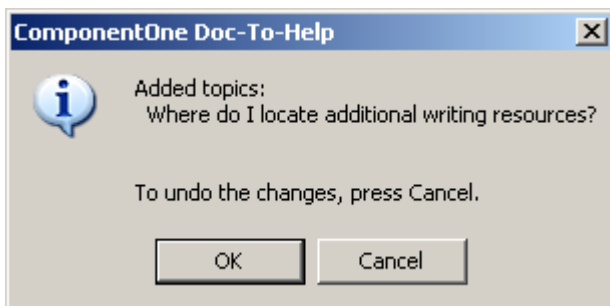


Enter the text for your topic and the heading style you want to use.

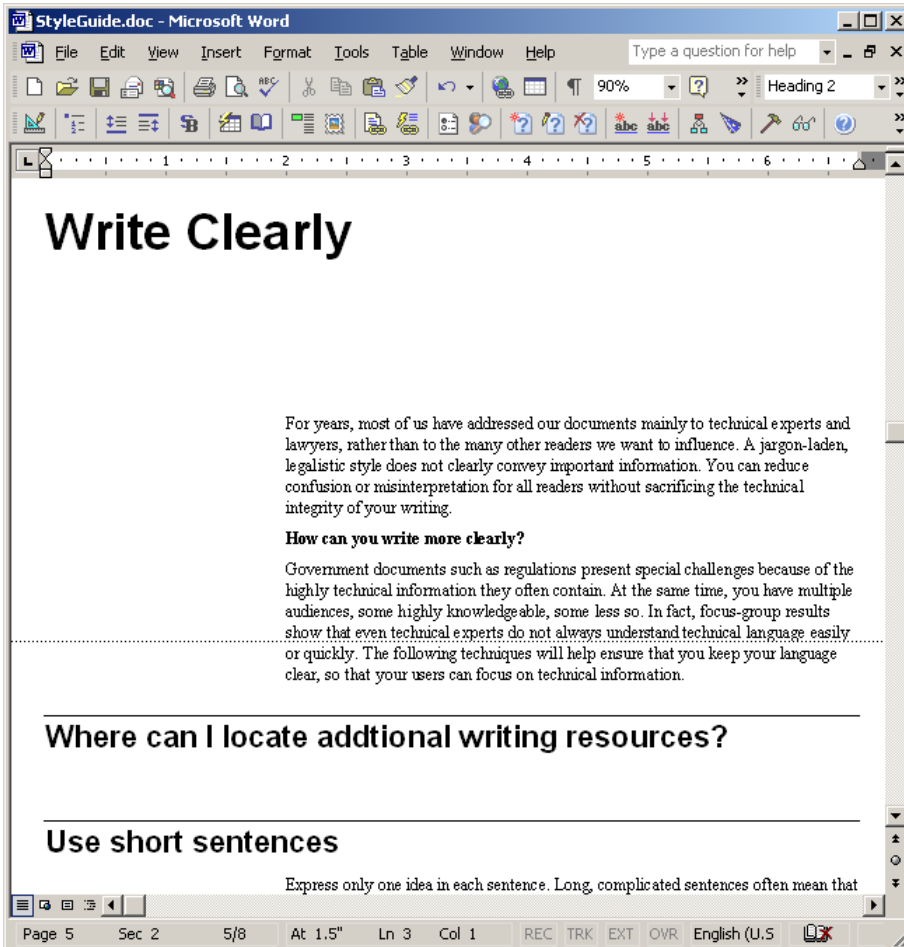


- Click **OK**.

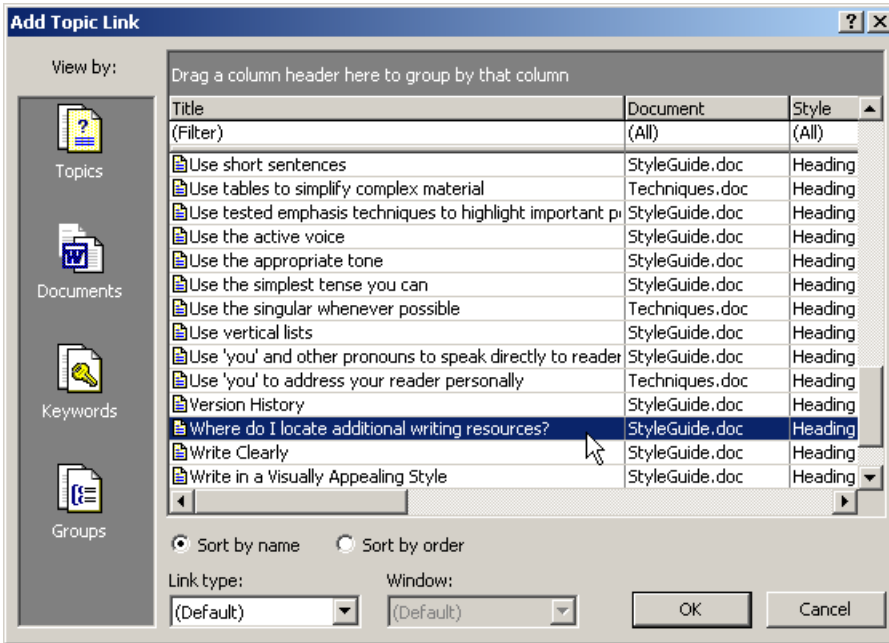
Doc-to-Help prompts you with a dialog box that allows you to see your new topic text and accept it as is or cancel the topic insert.



The new topic is added to your document in the selected location.




You can perform any topic function, for example **Add Topic Link**, without having to rebuild your project first.

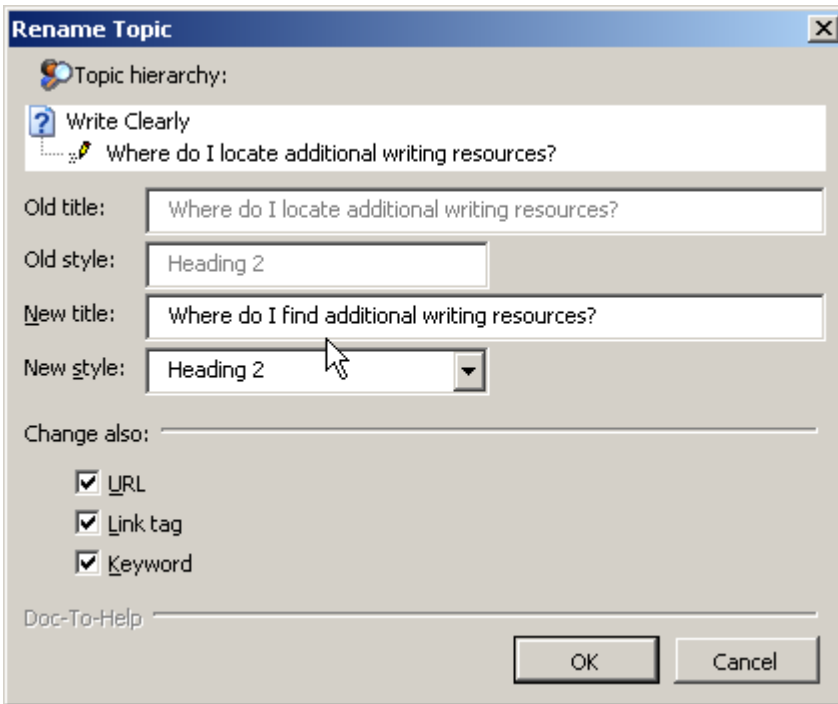


Using the Rename Topic Button

In cases where it is necessary to change the name of a topic, the **Rename Topic** button allows the author to change the topic name, as well as the automatically generated URL, Link Tag and Index Keyword related to that topic.

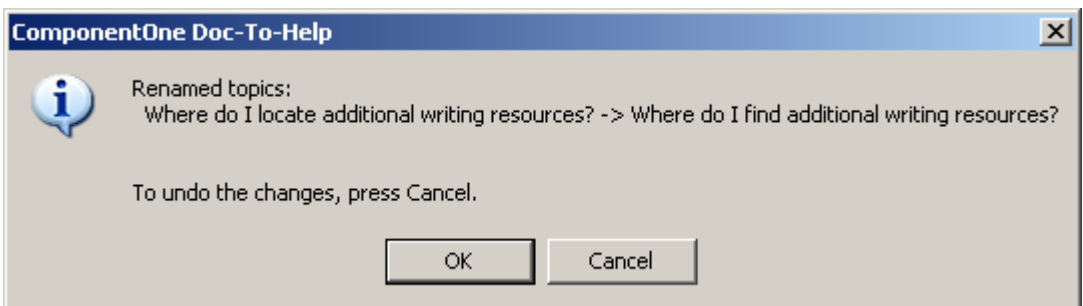
- Place your cursor anywhere within the topic to be modified.
- Click the **Rename Topic** button .

The **Rename Topic Dialog** opens.



- Modify the topic in the **New title** textbox. If you do not want Doc-To-Help change the **URL**, **Linktag** or **AutoKeyword** properties, uncheck the related checkboxes.
- Click **OK**.

Doc-to-Help prompts you with a dialog box that allows you to see the changes to your topic text and accept it as is or cancel the topic rename.



- After examining the change, click **OK**.

Note that the topic is changed as well as the related properties.

Using the Delete Topic Button

- To delete a topic, select the entire topic.

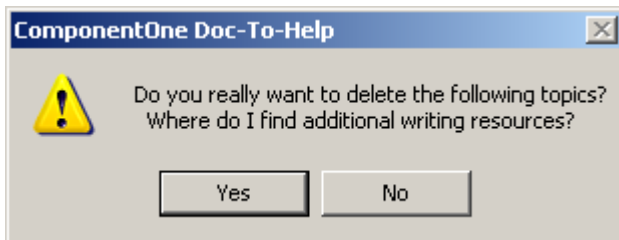
show that even technical experts do not always understand technical or quickly. The following techniques will help ensure that you keep clear, so that your users can focus on technical information.

Where do I find additional writing resources?

Use short sentences

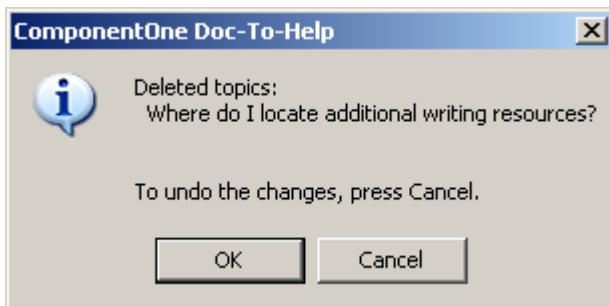
- Click the **Delete Topic** button .

Doc-to-Help prompts you with a dialog box that allows you to delete the topic or cancel the delete.



- Click **Yes**.

Doc-to-Help prompts you with a dialog box confirming the delete.



- Click **OK**.

Object Model Reference

AdjustForScreenSize Property

Determines whether the help window conforms to the resolution of the reader's display.

Type	Applies To
Boolean	Window (WinHelp)

Remarks

If true, the help window will not conform to the resolution of the reader's display.

Location

- Click the **Project Icon**.
- Make sure you have selected the WinHelp target from the Help Target Dropdown.
- Click on the **Windows** item in the left pane Tree View.
- The property is located under the Position group in the **Properties Pane**.

AdjustLeftIndent Property

Controls whether paragraph indentation is adjusted to account for wide margins when building online help.

Type	Applies To
Boolean	Document Project Settings

Remarks

By default, this property is true to accommodate the standard Doc-To-Help templates. Set this property to false if you are using custom templates and want to preserve the indentation used in your source documents.

Location (Document)

- Click the **Project Icon**.
- Select the **Documents** item from the left pane Tree View.

- The property is located under the Misc group in the **Properties Pane**.

Location (Project Settings)

- Click the **Project Icon**.
- Select the **Project Settings** item from the left pane Tree View.
- The property is located under the Document Defaults group in the **Properties Pane**.

AffectsAppearance Property

Determines whether the style defines the appearance of the text (font, color, etc.) in the target help file, or is only used to create a hotspot or keyword and does not affect target appearance.

Type	Applies To
Boolean	Character Style

Remarks

When set to False, the selected style is used to define hot spots (links) and keywords. It does not determine text formatting (font, color, etc.). The hot spot appearance is determined by the default rules of the target. (For example, links in HTML based targets are underlined blue.)

When set to True, the selected style defines the appearance of the text in the help target as well as the functionality of the links and keywords. In this case, formatting in the target help file will be exactly as specified in the document or the target template.

Character styles with an AffectsAppearance Property set to True can be used to specify the appearance of links generated by other means. For example, a style with AffectsAppearance Property = True and Type = None can be use to format topic links, dynamic links and margin notes if you need to override the default link appearance for them.

Location

- Click the **Project Icon**.
- Select the **Character Style** item from the left pane Tree View.
- The property is located under the Display group in the **Properties Pane**.

AlwaysOnTop Property

When true, the help window remains on top of all other windows in the desktop.

Type	Applies To
Boolean	Window (HTML Help) Window (WinHelp)

Location

- Click the **Project Icon**.
- Make sure you have selected the HTML Help or WinHelp targets from the **Help Target Dropdown**.
- Click on the **Windows** item in the left pane Tree View.
- The property is located under the Position group in the **Properties Pane**.

AnswerCount Property

Determines the maximum number of answers to be returned when a natural language search is performed.

Type	Applies To
Integer	Help Target (HTML) Help Target (HTML Help) Help Target (WinHelp)

Remarks

This property is only available in Doc-To-Help Pro.

Location

- Click the **Project Icon**.
- Select the **Help Targets** item from the left pane Tree View.
- The property is located under the Natural Search group in the **Properties Pane**.

AsciiName Property

Specifies the ASCII-only string used to identify the topic in situations where non-ASCII characters are not allowed.

Type	Applies To
String	Topic

Remarks

This property is important for Help authoring in languages that have non-ASCII alphabets, such as Cyrillic and Asian languages. In some instances, names generated by Doc-To-Help must be ASCII. Such instances include identifiers in *.h and *.bas map files that are used for context-sensitive help in C and Visual Basic programming languages, see **MapFileForC**, **MapFileForVB**.

Also, topic file URLs in HTML-based Help are also usually ASCII-only (controlled by the property **URLMode** with three possible values Full/AsciiOnly/TopicID). This is not a problem for languages with ASCII alphabets (English), and it is usually not a problem for languages that have only few non-ASCII letters (such as German). In these cases, the names, formed automatically from topic titles, omit the non-ASCII characters. In languages that are completely non-ASCII (such as Russian or Japanese) no part of the topic title can be used (ALL characters are omitted), so the names become a construct of underscores and numbers. By using the AsciiName property, you can make these names mnemonic.

You can change the default AsciiName property value for any topic to an ASCII string, and that string will be used as the topic identifier in a C or Visual Basic map file. It will also be used as the default value for the topic URL property, if URLMode=AsciiOnly and the topic URL property is empty.

Location

- Click the **Topic Icon**.
- The property is located under the Misc group in the **Properties Pane**.

AutoButtons Property

When true, displays subtopic buttons for paragraph styles with numeric outline levels greater than the parent level.

Type	Applies To
Boolean	Paragraph Style Topic Type

Remarks

To enable the subtopic buttons for a paragraph style, highlight the specific paragraph style in the upper right pane, then set the AutoButtons property to True.

Location

- Click the **Project Icon**.
- Select either the **Paragraph Styles** or **Topic Types** item from the left pane Tree View.
- The property is located under the Navigation group in the **Properties Pane**.

AutoContextID Property

When true, automatically creates Context ID's for all topics using this paragraph style or topic type.

Type	Applies To
Boolean	Paragraph Style Topic Type

Location

- Click the **Project Icon**.
- Select either the **Paragraph Styles** or **Topic Types** item from the left pane Tree View.
- The property is located under the Misc group in the **Properties Pane**.

AutoHideNavigationPane Property

Minimizes the HTML help navigation pane when HTML help is not the active window.

Type	Applies To
Boolean	Window (HTML Help)

Location

- Click the **Project Icon**.
- Make sure you have selected the HTML Help target from the **Help Target Dropdown**.
- Click on the **Windows** item in the left pane Tree View.
- The property is located under the Navigation group in the **Properties Pane**.

AutoIndex Property

When true, this property automatically creates index keywords from text formatted with that character style and associates them with the topic that contains the text.

Type	Applies To
Boolean	Character Style

Location

- Click the **Project Icon**.
- Select the **Character Styles** item from the left pane Tree View.
- The property is located under the Indexing group in the **Properties Pane**.

AutoKeyword Property

When true, this property automatically creates index keywords from topic titles formatted with that paragraph style or topic type and associates them with the appropriate topic.

Type	Applies To
Boolean	Paragraph Style Topic Type

Location

- Click the **Project Icon**.
- Select either the **Paragraph Styles** or **Topic Types** item from the left pane Tree View.
- The property is located under the Indexing group in the **Properties Pane**.

AutoLink Property

Creates unique link tags for topics formatted with the paragraph style or topic type by enabling the auto link tag feature.

Type	Applies To
Boolean	Paragraph Style Topic Type

Remarks

Link tags created in this manner are identical to the topic title except that spaces, hyphens and periods are converted into underscores.

Location

- Click the **Project Icon**.
- Select either the **Paragraph Styles** or **Topic Types** item from the left pane Tree View.
- The property is located under the Navigation group in the **Properties Pane**.

AutoNavigate Property

Determines which paragraph styles and topic types are included in the navigation sequence.

Type	Applies To
Boolean	Paragraph Style Topic Type

Location

- Click the **Project Icon**.
- Select either the **Paragraph Styles** or **Topic Types** item from the left pane Tree View.
- The property is located under the Navigation group in the **Properties Pane**.

AutoNext Property

Enables the next topic button for any active paragraph style.

Type	Applies To
Boolean	Paragraph Style Topic Type

Remarks

Displays a next topic button at the end of each associated topic that links to the next topic.

Location

- Click the **Project Icon**.

- Select either the **Paragraph Styles** or **Topic Types** item from the left pane Tree View.
- The property is located under the Navigation group in the **Properties Pane**.

AutoSizeHeight Property

When true, the height of secondary Help windows are automatically resized to fit the length of the current topic.

Type	Applies To
Boolean	Window (WinHelp)

Location

- Click the **Project Icon**.
- Make sure you have selected the WinHelp target from the **Help Target Dropdown**.
- Select the **Windows** item from the left pane Tree View.
- The Property is located under the Position group in the **Properties Pane**.

AutoSyncNavigationPane Property

Allows the heading or topic in the Navigation pane to automatically synchronize with whatever topic appears in the Topic pane.

Type	Applies To
Boolean	Window (HTML Help)

Location

- Click the **Project Icon**.
- Make sure you have selected the HTML Help target from the **Help Target Dropdown**.
- Select the **Windows** item from the left pane Tree View.
- The Property is located under the Navigation group in the **Properties Pane**.

BackgroundColor Property

Displays the current background color setting for the Help pane. To change the color, click the adjacent change button to open a standard color selection dialog.

Type	Applies To
RGBColor	Help Target (Help 2.0) Help Target (HTML) Help Target (HTML Help) Help Target (JavaHelp)

Location

- Click the **Project Icon**.
- Select the **Help Targets** item from the left pane Tree View.
- The property is located under the Appearance group in the **Properties Pane**.

BackgroundImage Property

Displays the image file to be used as a background in the Help pane. To insert a background image, click the adjacent change button to open the selection dialog.

Type	Applies To
String	Help Target (Help 2.0) Help Target (HTML) Help Target (HTML Help) Help Target (JavaHelp)

Location

- Click the **Project Icon**.
- Select the **Help Targets** item from the left pane Tree View.
- The property is located under the Appearance group in the **Properties Pane**.

BackgroundRepeat Property

Defines how the background image is arranged in the Help pane.

Type	Applies To
Enum	Help Target (Help 2.0) Help Target (HTML) Help Target (HTML Help) Help Target (JavaHelp)

Value	Description
None	Single image, no repeat.
Tiled	Images repeat in a tiled pattern.

Value	Description
Horizontal	Images repeated horizontally.
Vertical	Images repeated vertically.

Location

- Click the **Project Icon**.
- Select the **Help Targets** item from the left pane Tree View.
- The property is located under the Appearance group in the **Properties Pane**.

BaseName Property

Base file name to be used for generating help target files.

Type	Applies To
String	Help Target

Remarks

By default, this name will be the same as the file name for the help project itself.

Location

- Click the **Project Icon**.
- Select the **Help Targets** item from the left pane Tree View.
- The property is located under the Misc group in the **Properties Pane**.

BinaryIndex Property

When true, generates a binary index instead of a site map index.

Type	Applies To
Boolean	Help Target (HTML Help)

Remarks

In modular HTMLHelp projects, BinaryIndex can have any value in the module projects, but must be set to True in the hub project.

Location

- Click the **Project Icon**.

- Make sure you have selected the HTML Help target from the **Help Target Dropdown**.
- Select the **Help Targets** item from the left pane Tree View.
- The property is located under the Build group in the **Properties Pane**.

BinaryTOC Property

When true, generates a binary table of contents instead of a site map table of contents.

Type	Applies To
Boolean	Help Target (HTML Help)

Remarks

In modular HTMLHelp projects, BinaryTOC can have any value in the module projects, but must be set to False in the hub project.

Set this property to true if your main window uses the built-in Next and Previous buttons.

Location

- Click the **Project Icon**.
- Make sure you have selected the HTML Help target from the **Help Target Dropdown**.
- Select the **Help Targets** item from the left pane Tree View.
- The property is located under the Build group in the **Properties Pane**.

ButtonLabel Property

For Help targets, specifies the text that precedes the subtopic buttons.

For topics, overrides the Help target ButtonLabel text on a topic by topic basis.

Type	Applies To
String	Help Target (Help 2.0) Help Target (HTML) Help Target (HTML Help) Help Target (JavaHelp) Help Target (WinHelp) Topic

Remarks

Set this property to an empty string to suppress the default **More:** label.

Location (Help Targets)

- Click the **Project Icon**.
- Select the **Help Targets** item from the left pane Tree View.
- The property is located under the Display group in the **Properties Pane**.

Location (Topics)

- Click the **Topics Icon**.
- Select the property you wish to modify from the upper right Pane.
- The property is located under the Display group in the **Properties Pane**.

ButtonLabelStyle Property

Lists the name of the style used to format the label that precedes the subtopic buttons.

Type	Applies To
Style	Help Target (Help 2.0) Help Target (HTML) Help Target (HTML Help) Help Target (JavaHelp) Help Target (WinHelp)

Location

- Click the **Project Icon**.
- Select the **Help Targets** item from the left pane Tree View.
- The property is located under the Display group in the **Properties Pane**.

ButtonSeparator Property

When enabled, a dividing line separates the topic text from the subtopic buttons.

Type	Applies To
Boolean	Help Target (Help 2.0) Help Target (HTML) Help Target (HTML Help) Help Target (JavaHelp) Help Target (WinHelp)

Location

- Click the **Project Icon**.
- Select the **Help Targets** item from the left pane Tree View.
- The property is located under the Display group in the **Properties Pane**.

ButtonStyle Property

Lists the name of the style used to format automatically generated subtopic buttons.

Type	Applies To
Style	Help Target (Help 2.0) Help Target (HTML) Help Target (HTML Help) Help Target (JavaHelp) Help Target (WinHelp)

Location

- Click the **Project Icon**.
- Select the **Help Targets** item from the left pane Tree View.
- The property is located under the Display group in the **Properties Pane**.

ByLine Property

Specifies the byline text used on the title page of printed manuals.

Type	Applies To
String	Help Target (Manual)

Location

- Click the Project Icon.
- Make sure you have selected the Manual help target from the **Help Target Dropdown**.
- Select the **Help Targets** item from the left pane Tree View.

- The property is located under the Title Page group in the **Properties Pane**.

CancelKey Property

When true, this property disables the AutoKeyword property on a topic by topic basis.

Type	Applies To
Boolean	Topic

Location

- Click the **Topic Icon**.
- Make sure you have selected **All Topics**, **Topics With**, or **Topics Without** from the left pane Tree View.
- The property is located under the Behavior group in the **Properties Pane**.

CancelMap Property

When true, this property disables the AutoContextID property on a topic-by-topic basis.

Type	Applies To
Boolean	Topic

Location

- Click the **Topic Icon**.
- Make sure you have selected **All Topics**, **Topics With**, or **Topics Without** from the left pane Tree View.
- The property is located under the Behavior group in the **Properties Pane**.

CancelTag Property

When true, this property disables the **AutoLink** property on a topic by topic basis.

Type	Applies To
Boolean	Topic

Location

- Click the **Topic Icon**.
- Make sure you have selected **All Topics**, **Topics With**, or **Topics Without** from the left pane Tree View.

- The property is located under the Behavior group in the **Properties Pane**.

Caption Property

Represents the string to be displayed in the caption bar of the help window.

Type	Applies To
String	Window (HTML Help) Window (WinHelp) Help Target (JavaHelp)

Remarks

Because of limitations imposed by the Windows help compiler, Winhelp captions are limited to 50 characters.

Location

- Click the **Project Icon**.
- Make sure you have selected a HTML Help or WinHelp target from the **Help Target Dropdown**.
- Select the **Windows** item in the left pane Tree View.
- The property is located under the Misc group in the **Properties Pane**.

or

- Make sure you have selected the JavaHelp help target from the **Help Target Dropdown**.
- Select the **Help Targets** item from the left pane Tree View.
- The property is located under the Appearance group in the **Properties Pane**.

Charset Property

Specifies one of the character set aliases listed in HTML Character Sets and Named Entities.

Type	Applies To
String	Help Target (Help 2.0) Help Target (HTML) Help Target (HTML Help) Help Target (JavaHelp)

Location

- Click the **Project Icon**.
- Make sure you have selected the Help 2.0, HTML, HTML Help or JavaHelp target from the **Help Target Dropdown**.
- Select the **Help Targets** item from the left pane Tree View.
- The property is located under the Localization group in the **Properties Pane**.

Citation Property

Appends a citation to the end of any information (except in a context-sensitive pop-up window) that is copied from the Help file.

Type	Applies To
String	Help Target (WinHelp)

Location

- Click the **Project Icon**.
- Make sure you have selected the WinHelp target from the **Help Target Dropdown**.
- Select the **Help Targets** item from the left pane Tree View.
- The property is located under the Misc group in the **Properties Pane**.

Code Property

When activated, this property produces an editable textbox containing the script code.

Type	Applies To
String	Script

Location

- Click the **Project Icon**.
- Select the **Scripts** item from the left pane Tree View.
- The property is located under the Misc group in the **Properties Pane**.

CodePage Property

Defines the windows codepage number to be used by the help file.

Type	Applies To
Integer	Help Target

Remarks

This property is most commonly used when producing help files in foreign languages, providing special character sets for translation on the screen or in print.

Location

- Click the **Project Icon**.
- Select the **Help Targets** item from the left pane Tree View.
- The property is located under the Localization group in the **Properties Pane**.

ColorReduction Property

Determines the color threshold for images that are saved with source documents.

Type	Applies To
Enum	Help Target (HTML) Help Target (HTML Help) Help Target (WinHelp)

Value	Description
None	Images are not converted.
16 Colors	Images with more than 16 colors are reduced to 16.
256 Colors	Images with more than 256 colors are reduced to 256.

Remarks

Has no effect on linked images that are not saved with the document.

Location

- Click the **Project Icon**.
- Make sure you have selected the correct Help target from the **Help Target Dropdown**.
- Select the **Help Targets** item from the left pane Tree View.
- The property is located under the Display group in the **Properties Pane**.

Comments Property

When activated, this property provides an editable textbox to be used for comments by the help author. These comments are not accessible by the end user.

Type	Applies To
String	Topic

Location

- Click the **Topics Icon**.
- The property is located under the Misc group in the **Properties Pane**.

Compression Property

Determines the level of file compression used by the WinHelp compiler.

Type	Applies To
Enum	Help Target (WinHelp)

Value	Description
None	No compression.
Choose best	The WinHelp compiler determines the best algorithm to use.
Phrase	For help files under 100K.
Hall	For help files that will be compressed again by another utility.
Zeck	Minimal compression for quick builds.
Zeck + Phrase	Minimal compression for help files under 100K.
Zeck + Hall	Maximum compression.

Location

- Click the **Project Icon**.
- Make sure you have selected the WinHelp Help target from the **Help Target Dropdown**.
- Select the **Help Targets** item from the left pane Tree View.
- The property is located under the Misc group in the **Properties Pane**.

Concatenate Property

When true, subdocuments are concatenated into a single master document in printed manual builds.

Type	Applies To
String	Help Target (Manual)

Remarks

Set this property to false to generate a master document that uses Word RD fields to reference subdocuments.

Location

- Click the **Project Icon**.
- Make sure you have selected the Manual Help target from the **Help Target Dropdown**.
- Select the **Help Targets** item from the left pane Tree View.
- The property is located under the Build group in the **Properties Pane**.

ConfidenceLevel Property

Sets the minimum score percentage that controls which answers will be displayed when a natural language search is performed.

Type	Applies To
Integer	Help Target (HTML) Help Target (HTML Help) Help Target (WinHelp)

Remarks

Matching answers that have a lower score than the value of the ConfidenceLevel property will be discarded.

This property is only available in Doc-To-Help Pro.

Location

- Click the **Project Icon**.
- Select the **Help Targets** item from the left pane Tree View.
- The property is located under the Natural Search group in the **Properties Pane**.

ContentsOnly Property

When true, omits topics defined with this paragraph style or topic type from the help target, but uses them as book titles in the help contents.

Type	Applies To
Boolean	Paragraph Style Topic Type

Location

- Click the **Project Icon**.
- Select either the **Paragraph Styles** or **Topic Types** item from the left pane Tree View.
- The property is located under the Display group in the **Properties Pane**.

ContentsTab Property

When enabled, the Contents tab is added to the help window.

Type	Applies To
Boolean	Help Target (JavaHelp)

Location

- Click the **Project Icon**.
- Make sure you have selected the JavaHelp help target from the **Help Target Dropdown**.
- Select the **Help Targets** item from the left pane Tree View.
- The property is located under the Tabs group in the **Properties Pane**.

ContentsTitle Property

Allows for modification of a topic title in the help window contents area without changing the actual topic title in the source document.

Type	Applies To
String	Topic

Location

- Click the **Topics Icon**.

- The property is located under the Display group in the **Properties Pane**.

ContextString Property

When using context-sensitive and dynamic help with Microsoft Help 2.0, specifies a context string for the topic.

Type	Applies To
String	Topic

Remarks

The ContextString property applies to topics in the MS Help 2.0 target. It is used to build a context-sensitive index that is used in Microsoft Visual Studio .NET for context-sensitive help (F1) and dynamic help.

Each topic can have one or more context strings, or none. The ContextString property contains all context strings of the given topic separated by semicolon. Topic context strings form a hierarchical tree structure. A context string consists of dot-separated context names for each hierarchy level.

For example, a topic "Property MyProperty" can have:

ContextString = "MyCompany.MyProduct.MyProperty"

Topic "Properties MyProperty1 and MyProperty2" can have:

ContextString = "MyCompany.MyProduct.MyProperty1;MyCompany.MyProduct.MyProperty2"

This property is only available in Doc-To-Help Pro.

Location

- Click the **Topics Icon**.
- The property is located under the Display group in the **Properties Pane**.

CopyFolder Property

Defines the folder name where ancillary build files will reside.

Type	Applies To
String	Help Target

Location

- Click the **Project Icon**.
- Select the **Help Targets** item from the left pane Tree View.
- The property is located under the Misc group in the **Properties Pane**.

Copyright Property

Places a custom copyright notice in the Version dialog box of WinHelp.

Type	Applies To
String	Help Target (WinHelp)

Location

- Click the **Project Icon**.
- Make sure you have selected the WinHelp target from the **Help Target Dropdown**.
- Select the **Help Targets** item from the left pane Tree View.
- The property is located under the Misc group in the **Properties Pane**.

DefaultFile Property

File name to be used as part of the base URL for the HTML target.

Type	Applies To
String	Help Target (HTML)

Remarks

Typically, the filename will be *default.htm* or *index.html*.

Location

- Click the **Project Icon**.
- Make sure you have selected the HTML target from the **Help Target Dropdown**.
- Select the **Help Targets** item from the left pane Tree View.
- The property is located under the HTML group in the **Properties Pane**.

DefaultTemplate Property

Specifies the document template that will be attached to new documents created in the Project Editor.

Type	Applies To
Template	Project

Location

- Click the **Project Icon**.
- Select the **Project Settings** item from the left pane Tree View.
- The property is located under the Misc group in the **Properties Pane**.

DefaultWindow Property

Specifies the default window definition for the compiled help file.

Type	Applies To
Window	Window (HTML Help)

Location

- Click the **Project Icon**.
- Make sure you have selected the HTML target from the **Help Target Dropdown**.
- Select the **Help Targets** item from the left pane Tree View.
- The property is located under the Display group in the **Properties Pane**.

DisplayTitle Property

Allows for modification of a topic title with respect to help file searches without changing the actual topic title in the source document.

Type	Applies To
String	Topic

Remarks

Use this property to add qualifying text to like-named topics. For example, a search for “intro” may yield several topics named Introduction, but by modifying the DisplayTitle, you can get results such as Introduction (Help Authoring), Introduction (HTML), Introduction (WinHelp) without adding the text in parentheses to the source documents.

This is the equivalent of adding a \$ footnote in WinHelp or a <Title> in HTML.

Location

- Click the **Topics Icon**.
- The property is located under the Display group in the **Properties Pane**.

DynamicTOC Property

Generates a Dynamic HTML version of the contents page with an expandable outline.

Type	Applies To
Boolean	Help Target (HTML)

Location

- Click the **Project Icon**.
- Make sure you have selected the HTML target from the **Help Target Dropdown**.
- Select the **Help Targets** item from the left pane Tree View.
- The property is located under the HTML group in the **Properties Pane**.

ExplicitAccess Property

When true, defines a specific paragraph style or topic type as accessible only through a hyperlink. Paragraph styles and topic types defined as explicit are not accessible from the help contents, the index list or the text search.

Type	Applies To
Boolean	Paragraph Style Topic Type

Location

- Click the **Project Icon**.
- Select either the **Paragraph Styles** or **Topic Types** item from the left pane Tree View.
- The property is located under the Navigation group in the **Properties Pane**.

FavoritesTab Property

When enabled, the Favorites tab is added to the help window.

Type	Applies To
Boolean	Window (HTML Help) Help Target (Java Help)

Location

- Click the **Project Icon**.

- Make sure you have selected the HTML Help target from the **Help Target Dropdown**.
- Select the **Windows** item from the left pane Tree View.
- The property is located under the Tabs group in the **Properties Pane**.

or

- Make sure you have selected the JavaHelp help target from the **Help Target Dropdown**.
- Select the **Help Targets** item from the left pane Tree View.
- The property is located under the Tabs group in the **Properties Pane**.

FixLists Property

Controls how numbered lists are corrected in the printed manual output when the Concatenate property is set to true.

Type	Applies To
Boolean	Help Target (Manual)

Value	Description
Never	Never corrects the numbered lists.
Always	Always corrects the numbered lists.
Word 2000 Only (Default)	Corrects lists with Word 2000, but not with Word 2002 (XP).

Remarks

Because of the sophisticated manner in which Word 2000 tries to make list numbering intuitive, there are known issues with preserving numbering when performing copy/paste and other text manipulations. These issues are largely solved in Word 2002 (XP).

The default setting (Word 2000 Only), allows Doc-To-Help to automatically fix, re-number and re-format lists where it detects problems in the printed manual output.

On very rare occasions, attempts to detect and fix list problems can result in problems during Doc-To-Help builds. If you encounter such problems while the Doc-To-Help progress indicator reads "Restoring lists formatting...", you may want to skip the fix.

Set the FixLists property to Never when you want to skip the detect and fix functions.

Set the FixLists property to Always when you want to perform the detect and fix functions on Word 2002 (XP).

Location

- Click the Project Icon.

- Make sure you have selected the Manual help target from the **Help Target Dropdown**.
- Select the **Help Targets** item from the left pane Tree View.
- The property is located under the Build group in the **Properties Pane**.

Folder Property

Defines the folder name where the help files will reside.

Type	Applies To
String	Help Target

Location

- Click the **Project Icon**.
- Select the **Help Targets** item from the left pane Tree View.
- The property is located under the Misc group in the **Properties Pane**.

Frameset Property

Generates a frameset version of the HTML help project with the left frame displaying the contents and the right frame displaying the help topics.

Type	Applies To
Boolean	Help Target (HTML)

Location

- Click the **Project Icon**.
- Make sure you have selected the HTML target from the **Help Target Dropdown**.
- Select the **Help Targets** item from the left pane Tree View.
- The property is located under the HTML group in the **Properties Pane**.

GenerateCSS Property

When set to false, the existing cascading stylesheet will not be overwritten when building the Help target.

Type	Applies To
Boolean	Help Target (Help 2.0) Help Target (HTML) Help Target (HTML Help) Help Target (JavaHelp)

Location

- Click the **Project Icon**.
- Select the **Help Targets** item from the left pane Tree View.
- Select the Help target you wish to modify.
- The property is located under the Build group in the **Properties Pane**.

GenerateProjectFile Property

When set to false, the existing .hhp or .hpl files will not be overwritten when building the Help target.

Type	Applies To
Boolean	Help Target (HTML Help) Help Target (WinHelp)

Location

- Click the **Project Icon**.
- Select the **Help Targets** item from the left pane Tree View.
- Select the Help target you wish to modify.
- The property is located under the Build group in the **Properties Pane**.

Global Property

When True, \$global_ is prepended to the window name in generated files and <OBJECT> tags.

Type	Applies To
Boolean	Window (HML Help)

Remarks

Use this property in modular help projects to ensure that like-named secondary window references in child help files will resolve to the same physical window.

Location

- Click the **Project Icon**.
- Make sure you have selected the HTML target from the **Help Target Dropdown**.
- Select the **Windows** item from the left pane Tree View.
- The property is located under the Misc group in the **Properties Pane**.

Glossary Property

Specifies the document that contains glossary terms for the project.

Type	Applies To
Document	Project

Location

- Click the **Project Icon**.
- Select the **Project Settings** item from the left pane Tree View.
- The property is located under the Misc group in the **Properties Pane**.

Height Property

Determines the height of the help window.

Type	Applies To
Integer	Window (HTML Help) Window (WinHelp) Window (JavaHelp)

Location

- Click the **Project Icon**.
- Select the **Windows** item from the left pane Tree View.
- The property is located under the Position group in the **Properties Pane**.

Hidden Property

Any text formatted with a paragraph or character style with Hidden property set to True is omitted from the help target.

Type	Applies To
Boolean	Character Style Paragraph Style

Remarks

If a topic heading is formatted with this style, the whole topic is omitted from the help target.

Location

- Click the **Project Icon**.
- Select **Character Style** or **Paragraph Styles** item from the left pane Tree View.
- The property is located under the Display group in the **Properties Pane**.

HideNavigationPane Property

When true, the navigation pane for the help file is hidden by default.

Type	Applies To
Boolean	Window (HTML Help)

Location

- Click the **Project Icon**.
- Make sure you have selected the HTML Help target from the **Help Target Dropdown**.
- Select the **Windows** item from the left pane Tree View.
- The property is located under the Navigation group in the **Properties Pane**.

HubContents Property

Displays the name of the contents (.cnt) file to associate with the Help file.

Type	Applies To
String	Help Target (WinHelp)

Remarks

For modular WinHelp systems, use this property to associate a component Help file with its hub.

Location

- Click the **Project Icon**.
- Make sure you have selected the WinHelp target from the **Help Target Dropdown**.
- The property is located under the Misc group in the **Properties Pane**.

ID Property

Returns a unique numeric identifier assigned to the topic during compilation (read-only).

Type	Applies To
Long	Topic

Location

- Click the **Topics Icon**.
- The property is located under the Misc group in the **Properties Pane**.

IgnoreFontNames Property

When set to True, font names specified in the source document are ignored. Default font names are used in the help target.

Type	Applies To
Boolean	Help Target (JavaHelp)

Remarks

Default: True.

This property is used in conjunction with IgnoreFontSizes.

Location

- Click the **Project Icon**.
- Make sure you have selected the JavaHelp help target from the **Help Target Dropdown**.
- Select the **Help Targets** item from the left pane Tree View.
- The property is located under the Fonts group in the **Properties Pane**.

IgnoreFontSizes Property

When set to True, font sizes specified in the source document are ignored. Default font sizes are used in the help target.

Type	Applies To
Boolean	Help Target (JavaHelp)

Remarks

Default: True.

This property is used in conjunction with IgnoreFontNames. If this property is set to False, the ScaleFontSizes property takes effect.

Location

- Click the **Project Icon**.
- Make sure you have selected the JavaHelp help target from the **Help Target Dropdown**.
- Select the **Help Targets** item from the left pane Tree View.
- The property is located under the Fonts group in the **Properties Pane**.

IndexTab Property

When enabled, the Index tab is added to the help window.

Type	Applies To
Boolean	Help Target (JavaHelp)

Jump1Caption Property

Enter the caption for the Jump1 button here.

Type	Applies To
String	Window (HTML Help)

Location

- Click the **Project Icon**.

- Make sure you have selected the HTML Help target from the **Help Target Dropdown**.
- Select the **Windows** item from the left pane Tree View.
- The property is located under the Buttons group in the **Properties Pane**.

Jump1URL Property

Enter the URL of the HTML file for the Jump1 button here.

Type	Applies To
String	Window (HTML Help)

Location

- Click the **Project Icon**.
- Make sure you have selected the HTML Help target from the **Help Target Dropdown**.
- Select the **Windows** item from the left pane Tree View.
- The property is located under the Buttons group in the **Properties Pane**.

Jump2Caption Property

Enter the caption for the Jump2 button here.

Type	Applies To
String	Window (HTML Help)

Location

- Click the **Project Icon**.
- Make sure you have selected the HTML Help target from the **Help Target Dropdown**.
- Select the **Windows** item from the left pane Tree View.
- The property is located under the Buttons group in the **Properties Pane**.

Jump2URL Property

Enter the URL of the HTML file for the Jump2 button here.

Type	Applies To
String	Window (HTML Help)

Location

- Click the **Project Icon**.
- Make sure you have selected the HTML Help target from the **Help Target Dropdown**.
- Select the **Windows** item from the left pane Tree View.
- The property is located under the Buttons group in the **Properties Pane**.

KeepOutlineNumbers Property

When enabled, outline numbers are included as part of topic titles and are present in help targets.

Type	Applies To
Boolean	Document Project Settings

Remarks

This property can be set as default in the Project Settings properties pane. Modifying this property for an individual document will override the global settings for that document only.

This property only affects the RTF files generated from source documents; it does not affect the source documents themselves.

Location (Documents)

- Click the **Project Icon**
- Select the **Documents** item from the left pane Tree View
- The property is located under the Misc group in the **Properties Pane**

Location (Project Settings)

- Click the **Project Icon**
- Select the **Project Settings** item from the left pane Tree View
- The property is located under the Document Defaults group in the **Properties Pane**

KeepPageBreaks Property

When true, this property retains the page break characters in the source documents during compilation of a Printed Manual Help target. Set this property to false to discard page break characters.

Type	Applies To
Boolean	Project Settings

Remarks

This property can be set as default in the Project Settings properties pane. Modifying this property for an individual document will override the global settings for that document only.

Location (Project Settings)

- Click the **Project Icon**
- Select the **Project Settings** item from the left pane Tree View
- The property is located under the Document Defaults group in the **Properties Pane**

Left Property

Determines the location of the left edge of the help window.

Type	Applies To
Integer	Window (HTML Help) Window (WinHelp) Window (JavaHelp)

Remarks

In conjunction with the Top property, determines the position of the help window on the screen.

Location

- Click the **Project Icon**.
- Select the **Windows** item from the left pane Tree View.
- The property is located under the Position group in the **Properties Pane**.

Locale Property

This property is used to specify the language in the resulting target files. Each target has its own way of specifying language. For example, HTML Help has a Language option in the [OPTIONS] section of the project file (.hhp).

Type	Applies To
Integer	Help Target (Help 2.0) Help Target (HTML) Help Target (HTML Help) Help Target (JavaHelp) Help Target (WinHelp)

Remarks

Some common values for this property are as follows:

English (U.S.)	1033
English (U.K.)	2057
French	1036
German	1031
Italian	1040
Spanish	1034

Location

- Click the **Project Icon**.
- Make sure you have selected the WinHelp target from the **Help Target Dropdown**.
- The property is located under the Localization group in the **Properties Pane**.

MapFileForC Property

Creates C or C++ header files for context sensitive help.

Type	Applies To
Boolean	Help Target (WinHelp)

Location

- Click the **Project Icon**.
- Make sure you have selected the WinHelp target from the **Help Target Dropdown**.

- Select the **Help Targets** item from the left pane Tree View.
- The property is located under the Misc group in the **Properties Pane**.

MapFileForVB Property

Creates Visual Basic header files for context sensitive help.

Type	Applies To
Boolean	Help Target (HTML Help) Help Target (WinHelp)

Location

- Click the **Project Icon**.
- Make sure you have selected the HTML Help or the WinHelp target from the **Help Target Dropdown**.
- Select the **Help Targets** item from the left pane Tree View.
- The property is located under the Misc group in the **Properties Pane**.

MapNumberOffset Property

The value of this property is added to the automatically generated map numbers to prevent numbering conflicts in modular Help systems.

Type	Applies To
Integer	Project Settings

Location

- Click the **Project Icon**.
- Select the **Project Settings** item from the left pane Tree View.
- The property is located under the Misc group in the **Properties Pane**.

MapNumbers Property

Determines whether a unique Context ID (map number) is generated for each topic.

Type	Applies To
Enum	Help Target (HTML Help) Help Target (WinHelp)

Value	Description
Automatic	Map numbers are generated automatically for each topic.
Automatic+Custom	Map numbers are generated automatically for each topic, but can be customized as required.
Custom	No map numbers are generated automatically.

Remarks

For the Context IDs to be generated, the **AutoContextID** property must be set to **True** for each relevant Topic Type or Paragraph Style.

Location

- Click the **Project Icon**.
- Make sure you have selected the HTML Help or WinHelp targets from the **Help Target Dropdown**.
- Select the **Help Targets** item in the left pane Tree View.
- The property is located under the Misc group in the **Properties Pane**.

Maximized Property

If true, the help window is automatically maximized when displayed.

Type	Applies To
Boolean	Window (HTML Help) Window (WinHelp)

Location

- Click the **Project Icon**.
- Make sure you have selected the HTML Help or WinHelp targets from the **Help Target Dropdown**.
- Select the **Windows** item from the left pane Tree View.
- The property is located under the Position group in the **Properties Pane**.

MergeContents Property

For a placeholder topic in a modular hub project, this property specifies the platform-dependent contents file to be loaded dynamically.

Type	Applies To
String	Topic

Remarks

This property only applies to modular hub projects for the WinHelp and HTML Help platforms.

Location

- Click the **Topics Icon**.
- The property is located under the Modular group in the **Properties Pane**.

MergeFile Property

For a placeholder topic in a modular hub project, this property specifies the platform-dependent help file to be loaded dynamically.

Type	Applies To
String	Topic

Remarks

This property only applies to modular hub projects for the WinHelp and HTML Help platforms.

Location

- Click the **Topics Icon**.
- The property is located under the Modular group in the **Properties Pane**.

MergeTitle Property

For a placeholder topic in a modular hub project, this property specifies the text used to disambiguate like-named topics in keyword search lists.

Type	Applies To
String	Topic

Remarks

This property only applies to modular hub projects for the WinHelp platform. If not specified, the value of the MergeFile property is used.

Location

- Click the **Topics Icon**.
- The property is located under the Modular group in the **Properties Pane**.

MidTopic Property

Defines a paragraph style or a topic type as an “in topic” jump location similar to an HTML anchor. By defining a paragraph style or topic type as MidTopic and defining a character style as a jump, end users will have jump functionality within a lengthy topic.

Type	Applies To
Boolean	Paragraph Style Topic Type

Location

- Click the **Project Icon**.
- Select either the **Paragraph Styles** or **Topic Types** item from the left pane Tree View.
- The property is located under the Navigation group in the **Properties Pane**.

ModularHub Property

If true, the generated help file can dynamically load the contents of other help files, if present.

Type	Applies To
Boolean	Project

Remarks

To specify a component help file in a modular hub project, create a placeholder topic, then set the MergeFile and MergeContents properties of that topic to the component filenames. When testing your project, you will need to copy the component files into the output folder of each modular help target.

This property only applies to the WinHelp and HTML Help platforms.

Location

- Click the **Project Icon**.
- Select the **Project Settings** item from the left pane Tree View.
- The property is located under the Misc group in the **Properties Pane**.

MultiLink Property

When false, only the first occurrence of a "jump" character style (in a given topic) generates a link. All other formatted occurrences are skipped.

Type	Applies To
Boolean	Character Style

Location

- Click the **Project Icon**.
- Select the **Character Style** item from the left pane Tree View.
- The property is located under the Display group in the **Properties Pane**.

Name Property

Returns a string containing the name of a specified object (read-only).

Type	Applies To
String	Attribute Character Style Document Group Help Target Keyword Paragraph Style Script Template Topic Type Window

Location

- Click the **Project Icon**.
- Select any of the items in the left pane Tree View.
- The property is located under the Misc group in the **Properties Pane**.

Namespace Property

Specifies the Help 2.0 namespace that is used to register the Help file after a successful build.

Type	Applies To
String	Help Target (Help 2.0)

Remarks

If the Namespace property is empty, the BaseName property value is used. If BaseName is empty, the file name of the project itself is used.

NOTE: The Namespace and ParentNamespace properties only affect Help file registration on the author's machine. Registration on the user machine is handled by standard Help 2.0 means, with Windows Installer.

Location

- Click the Project Icon.
- Make sure you have selected the Help 2.0 help target from the **Help Target Dropdown**.
- Select the **Help Targets** item from the left pane Tree View.
- The property is located under the Strings group in the **Properties Pane**.

NavigationPaneWidth Property

Defines the width of the navigation pane of the help file in pixels.

Type	Applies To
Integer	Window (HTML Help)

Location

- Click the **Project Icon**.
- Make sure you have selected the HTML Help target from the **Help Target Dropdown**.
- Select the **Windows** item from the left pane Tree View.
- The property is located under the Navigation group in the **Properties Pane**.

Nonscrolling Property

When the Nonscrolling property of a paragraph style or topic type is set to true, any topic headings formatted with that paragraph style or topic type will appear in the non-scrolling region of the help window.

Type	Applies To
Boolean	Paragraph Style Topic Type

Remarks

This option only applies to WinHelp builds.

Location

- Click the **Project Icon**.
- Select either the **Paragraph Styles** or **Topic Types** item from the left pane Tree View.
- The property is located under the Display group in the **Properties Pane**.

NonscrollingColor Property

Displays the current color setting for the non-scrolling help window. To change the color, click the adjacent change button to open a standard color selection dialog.

Type	Applies To
Long	Window (WinHelp)

Location

- Click the **Project Icon**.
- Make sure you have selected the WinHelp target from the **Help Target Dropdown**.
- Select the **Windows** item from the left pane Tree View.
- The property is located under the Appearance group in the **Properties Pane**.

OneBrowseSequence Property

When true, this property provides a continuous browse sequence that spans multiple source documents.

Type	Applies To
Boolean	Window (WinHelp)

Location

- Click the **Project Icon**.

- Make sure you have selected the WinHelp target from the **Help Target Dropdown**.
- Select the **HelpTargets** item from the left pane Tree View.
- The property is located under the Navigation group in the **Properties Pane**.

OnlineOnly Property

When true, omits paragraphs formatted with this style from the printed manual target only.

Type	Applies To
Boolean	Paragraph Style

Remarks

This property has no effect in online Help targets.

Location

- Click the **Project Icon**.
- Select the **Paragraph Styles** item from the left pane Tree View.
- The property is located under the Display group in the **Properties Pane**.

OnPage Property

Specifies the default text used when completing a cross-reference.

Type	Applies To
String	Project

Location

- Click the **Project Icon**.
- Select the **Project Settings** item from the left pane Tree View.
- The property is located under the Localization group in the **Properties Pane**.

Order Property

Determines the order of the documents in the help contents.

Type	Applies To
Integer	Document

Location

- Click the **Project Icon**.
- Select the **Documents** item from the left pane Tree View.
- The property is located under the Misc group in the **Properties Pane**.

PageRef Property

When producing a manual, this property places the page number of the referenced topic next to the text formatted with this style.

Type	Applies To
Boolean	Character Style

Remarks

This option is only available for Jump and Popup links.

Location

- Click the **Project Icon**.
- Select either the **Character Style** item from the left pane Tree View.
- The property is located under the Display group in the **Properties Pane**.

Parent Property

Shows the ID number of the related parent topic.

Type	Applies To
Long	Topic

Location

- Click the **Topics Icon**.
- The property is located under the Misc group in the **Properties Pane**.

ParentNamespace Property

Specifies the namespace for the built help collection to plug into. Usually, that will be the namespace of the Visual Studio .NET Combined Collection. Set this property if you want the built Help to be automatically registered as a plug-in for Visual Studio help. For example, for Visual Studio .NET 2003, set ParentNamesace to "MS.VSCC.2003".

Type	Applies To
String	Help Target (Help 2.0)

Remarks

NOTE: The Namespace and ParentNamespace properties only affect Help file registration on the author's machine. Registration on the user machine is handled by standard Help 2.0 means, with Windows Installer.

Location

- Click the Project Icon.
- Make sure you have selected the Help 2.0 help target from the **Help Target Dropdown**.
- Select the **Help Targets** item from the left pane Tree View.
- The property is located under the Strings group in the **Properties Pane**.

Popup Property

When the popup property of a paragraph style or topic type is set to true, any topic formatted with that paragraph style or topic type will appear in a popup window on top of the help window.

Type	Applies To
Boolean	Paragraph Style Topic Type

Location

- Click the **Project Icon**.
- Select either the **Paragraph Styles** or **Topic Types** item from the left pane Tree View.
- The property is located under the Display group in the **Properties Pane**.

PrefixIndexListFile Property

Allows modification of the default prefix for topic list files corresponding to the main index.

Type	Applies To
String	Help Target (HTML) Help Target (JavaHelp)

Location

- Click the **Project Icon**.
- Make sure you have selected the HTML target from the **Help Target Dropdown**.
- Select the **Help Targets** item from the left pane Tree View.
- The property is located under the Strings group in the **Properties Pane**.

PrefixTopicListFile Property

Allows modification of the default prefix for topic list files corresponding to keyword hot spots.

Type	Applies To
String	Help Target (HTML) Help Target (JavaHelp)

Location

- Click the **Project Icon**.
- Make sure you have selected the HTML target from the **Help Target Dropdown**.
- Select the **Help Targets** item from the left pane Tree View.
- The property is located under the Strings group in the **Properties Pane**.

RelatedJumps Property

Controls the order of automatic and/or custom jumps on a topic by topic basis.

Type	Applies To
Enum	Topic

Value	Description
None	No Jumps are displayed
Automatic	Only Automatic jumps are displayed
Automatic + Custom	Automatic then Custom jumps are displayed
Custom + Automatic	Custom then Automatic jumps are displayed
Custom	Only Custom jumps are displayed

Location

- Click the **Topic Icon**.
- The property is located under the Navigation group in the **Properties Pane**.

Replacement Property

In certain cases, returns a string that overrides the generated default text.

Type	Applies To
String	Character Style Paragraph Style Topic Type

Remarks

For paragraph styles and topic types with AutoNext set to true, this string is used instead of the topic title next to the generated button.

For character styles with PageRef set to true, this string specifies the format of page number references in printed manual targets. This string is ignored unless it contains a pound sign, which is replaced with a PAGEREF field.

Location

- Click the **Project Icon**.
- Select either the **Character Style**, **Paragraph Styles** or **Topic Types** item from the left pane Tree View.
- The property is located under the Display group in the **Properties Pane**.

SaveUserPosition Property

When true, stores the size and position of the help window in the registry when the user modifies it.

Type	Applies To
Boolean	Window (HTML Help)

Location

- Click the **Project Icon**.
- Make sure you have selected the HTML Help target from the **Help Target Dropdown**.
- Select the **Windows** item from the left pane Tree View.
- The property is located under the Position group in the **Properties Pane**.

ScaleFontSizes Property

If IgnoreFontSizes = False, this property is used to scale all font sizes in the help target.

Type	Applies To
Real	Help Target (JavaHelp)

Remarks

This property is used in conjunction with IgnoreFontSizes property. If IgnoreFontSizes = True, this property is ignored.

JavaHelp has a known problem that makes all fonts appear smaller (approximately 1.3 times smaller) than they should be. By setting IgnoreFontSizes property to True, you will have consistent default JavaHelp font sizes. Alternatively, you can set the ScaleFontSizes property to a real number other than default 1.0 that will scale all fonts to adjust their sizes. For example, if JavaHelp displays fonts 1.3 times smaller that they should be, you can set ScaleFontSizes to 1.3 to fix that.

Location

- Click the **Project Icon**.
- Make sure you have selected the JavaHelp help target from the **Help Target Dropdown**.
- Select the **Help Targets** item from the left pane Tree View.
- The property is located under the Fonts group in the **Properties Pane**.

ScaleGraphics Property

When set to false, graphics embedded within source documents will not be scaled, but will be displayed at their original size in online Help targets.

Type	Applies To
Boolean	Help Target (Help 2.0) Help Target (HTML) Help Target (HTML Help) Help Target (JavaHelp) Help Target (WinHelp)

Location

- Click the **Project Icon**.
- Make sure you have select the correct help target from the **Help Target Dropdown**.
- The property is located under the Display group in the **Properties Pane**.

Script Property

Displays the script to be run during compilation whenever text formatted with this style is encountered.

Type	Applies To
Script	Character Style Paragraph Style Topic Type

Remarks

Will show None if no script is associated with the style.

Location

- Click the **Project Icon**.
- Select either the **Character Style**, **Paragraph Styles** or **Topic Types** item from the left pane Tree View.
- The property is located under the Misc group in the **Properties Pane**.

SearchCaption Property

Specifies the text added to the tab caption (HTML Help) or button caption (WinHelp) when ComponentOne Natural Search is enabled.

Type	Applies To
String	Help Target (HTML Help) Help Target (WinHelp)

Remarks

This property is only available in Doc-To-Help Pro.

Location

- Click the **Project Icon**.
- Select the **Help Targets** item from the left pane Tree View.
- The property is located under the Natural Search group in the **Properties Pane**.

SearchEnabled Property

Controls whether ComponentOne Natural Search is enabled for a Help target, document, or topic.

Type	Applies To
Boolean	Document Help Target (HTML) Help Target (HTML Help) Help Target (WinHelp) Topic

HTML Help or WinHelp

Set this property to true to generate Natural Search indexes for a Help target and add a custom tab (HTML Help) or button (WinHelp) to the Help file.

HTML

Set this property to true to generate Natural Search indexes for a Help target. Note that the Help target itself will not have a Natural Search interface. This option is provided solely for importing HTML content and indexes into **ComponentOne Response**.

Document

Set this property to false to exclude an individual document from Natural Search processing.

Topic

Set this property to false to exclude an individual topic from Natural Search processing.

Remarks

This property is only available in Doc-To-Help Pro.

Location (Help Targets)

- Click the **Project Icon**.
- Select the **Help Targets** item from the left pane Tree View.
- The property is located under the Natural Search group in the **Properties Pane**.

Location (Documents)

- Click the **Project Icon**.
- Select the **Documents** item from the left pane Tree View.
- Select the appropriate document from the right pane.
- The property is located under the Natural Search group in the **Properties Pane**.

Location (Topics)

- Click the **Topics Icon**.
- Select the appropriate topic from the right pane.
- The property is located under the Natural Search group in the **Properties Pane**.

SearchTab Property

When enabled, the Search tab is added to the help window.

Type	Applies To
Boolean	Help Target (Java Help)

Location

- Click the **Project Icon**.

- Make sure you have selected the HTML Help target from the **Help Target Dropdown**.
- Select the **Windows** item from the left pane Tree View.
- The property is located under the Tabs group in the **Properties Pane**.

SearchTabAdvanced Property

Adds additional functionality to the search tab.

Type	Applies To
Boolean	Window (HTML Help)

Remarks

Only available when the SearchTab property is true.

Location

- Click the **Project Icon**.
- Make sure you have selected the HTML Help target from the **Help Target Dropdown**.
- Select the **Windows** item from the left pane Tree View.
- The property is located under the Tabs group in the **Properties Pane**.

ShowBackButton Property

When selected, the Back button is included in the help window.

Type	Applies To
Boolean	Window (HTML Help) Window (WinHelp)

Location

- Click the **Project Icon**.
- Make sure you have selected the HTML Help or the WinHelp targets from the **Help Target Dropdown**.
- Select the **Windows** item from the left pane Tree View.
- The property is located under the Buttons group in the **Properties Pane**.

ShowBrowseButtons Property

When selected, the Browse buttons are included in the help window.

Type	Applies To
Boolean	Window (WinHelp)

Location

- Click the **Project Icon**.
- Make sure you have selected the WinHelp target from the **Help Target Dropdown**.
- Select the **Windows** item from the left pane Tree View.
- The property is located under the Buttons group in the **Properties Pane**.

ShowContentsButton Property

When selected, the Contents button is included in the help window.

Type	Applies To
Boolean	Window (WinHelp)

Location

- Click the **Project Icon**.
- Make sure you have selected the WinHelp target from the **Help Target Dropdown**.
- Select the **Windows** item from the left pane Tree View.
- The property is located under the Buttons group in the **Properties Pane**.

ShowFindButton Property

When selected, the Find button is included in the help window.

Type	Applies To
Boolean	Window (WinHelp)

Location

- Click the **Project Icon**.

- Make sure you have selected the WinHelp target from the **Help Target Dropdown**.
- Select the **Windows** item from the left pane Tree View.
- The property is located under the Buttons group in the **Properties Pane**.

ShowForwardButton Property

When selected, the Forward button is included in the help window.

Type	Applies To
Boolean	Window (HTML Help)

Location

- Click the **Project Icon**.
- Make sure you have selected the HTML Help target from the **Help Target Dropdown**.
- Select the **Windows** item from the left pane Tree View.
- The property is located under the Buttons group in the **Properties Pane**.

ShowHelpTopicsButton Property

When selected, the Help Topics button is included in the help window.

Type	Applies To
Boolean	Window (WinHelp)

Location

- Click the **Project Icon**.
- Make sure you have selected the WinHelp target from the **Help Target Dropdown**.
- Select the **Windows** item from the left pane Tree View.
- The property is located under the Buttons group in the **Properties Pane**.

ShowHideShowButton Property

When selected, the Hide and Show buttons are included in the help window.

Type	Applies To
Boolean	Window (HTML Help)

Location

- Click the **Project Icon**.
- Make sure you have selected the HTML Help target from the **Help Target Dropdown**.
- Select the **Windows** item from the left pane Tree View.
- The property is located under the Buttons group in the **Properties Pane**.

ShowHomeButton Property

When selected, the Home button is included in the help window.

Type	Applies To
Boolean	Window (HTML Help)

Location

- Click the **Project Icon**.
- Make sure you have selected the HTML Help target from the **Help Target Dropdown**.
- Select the **Windows** item from the left pane Tree View.
- The property is located under the Buttons group in the **Properties Pane**.

ShowIndexButton Property

When selected, the Index button is included in the help window.

Type	Applies To
Boolean	Window (WinHelp)

Location

- Click the **Project Icon**.
- Make sure you have selected the WinHelp target from the **Help Target Dropdown**.
- Select the **Windows** item from the left pane Tree View.

- The property is located under the Buttons group in the **Properties Pane**.

ShowJump1Button Property

When selected, the Jump1 button is included in the help window.

Type	Applies To
Boolean	Window (HTML Help)

Remarks

Choose the caption and URL for this button with the Jump1Caption and Jump1URL properties.

Location

- Click the **Project Icon**.
- Make sure you have selected the HTML Help target from the **Help Target Dropdown**.
- Select the **Windows** item from the left pane Tree View.
- The property is located under the Buttons group in the **Properties Pane**.

ShowJump2Button Property

When selected, the Jump2 button is included in the help window.

Type	Applies To
Boolean	Window (HTML Help)

Remarks

Choose the caption and URL for this button with the Jump2Caption and Jump2URL properties.

Location

- Click the **Project Icon**.
- Make sure you have selected the HTML Help target from the **Help Target Dropdown**.
- Select the **Windows** item from the left pane Tree View.
- The property is located under the Buttons group in the **Properties Pane**.

ShowLocateButton Property

When selected, the Locate button is included in the help window.

Type	Applies To
Boolean	Window (HTML Help)

Location

- Click the **Project Icon**.
- Make sure you have selected the HTML Help target from the **Help Target Dropdown**.
- Select the **Windows** item from the left pane Tree View.
- The property is located under the Buttons group in the **Properties Pane**.

ShowNextButton Property

When selected, the Next button is included in the help window.

Type	Applies To
Boolean	Window (HTML Help)

Remarks

The **BinaryTOC** property must be set to True for the navigation buttons to function.

Location

- Click the **Project Icon**.
- Make sure you have selected the HTML Help target from the **Help Target Dropdown**.
- Select the **Windows** item from the left pane Tree View.
- The property is located under the Buttons group in the **Properties Pane**.

ShowOptionsButton Property

When selected, the Options button is included in the help window.

Type	Applies To
Boolean	Window (HTML Help) Window (WinHelp)

Location

- Click the **Project Icon**.
- Make sure you have selected the HTML Help or the WinHelp target from the **Help Target Dropdown**.
- Select the **Windows** item from the left pane Tree View.
- The property is located under the Buttons group in the **Properties Pane**.

ShowPreviousButton Property

When selected, the Previous button is included in the help window.

Type	Applies To
Boolean	Window (HTML Help)

Remarks

The **BinaryTOC** property must be set to True for the navigation buttons to function.

Location

- Click the **Project Icon**.
- Make sure you have selected the HTML Help target from the **Help Target Dropdown**.
- Select the **Windows** item from the left pane Tree View.
- The property is located under the Buttons group in the **Properties Pane**.

ShowPrintButton Property

When selected, the Print button is included in the help window.

Type	Applies To
Boolean	Window (HTML Help) Window (WinHelp)

Location

- Click the **Project Icon**.
- Make sure you have selected the HTML Help or the WinHelp target from the **Help Target Dropdown**.
- Select the **Windows** item from the left pane Tree View.
- The property is located under the Buttons group in the **Properties Pane**.

ShowRefreshButton Property

When selected, the Refresh button is included in the help window.

Type	Applies To
Boolean	Window (HTML Help)

Location

- Click the **Project Icon**.
- Make sure you have selected the HTML Help target from the **Help Target Dropdown**.
- Select the **Windows** item from the left pane Tree View.
- The property is located under the Buttons group in the **Properties Pane**.

ShowStopButton Property

When selected, the Stop button is included in the help window.

Type	Applies To
Boolean	Window (HTML Help)

Location

- Click the **Project Icon**.
- Make sure you have selected the HTML Help target from the **Help Target Dropdown**.
- Select the **Windows** item from the left pane Tree View.
- The property is located under the Buttons group in the **Properties Pane**.

SkipGlossary Property

When true, omits the Glossary topic from the generated Help contents.

Type	Applies To
Boolean	Help Target (HTML Help) Help Target (WinHelp)

Remarks

For each component Help file in a modular system, set this property to true to eliminate duplicate Glossary topics when viewing the hub.

- Click the **Project Icon**.
- Make sure you have selected the HTML Help or the WinHelp targets from the **Help Target Dropdown**.
- The property is located under the Misc group in the **Properties Pane**.

SplitFile Property

Controls whether a document is split into separate topic files for HTML-based platforms.

Type	Applies To
Boolean	Document (Help 2.0) Document (HTML) Document (HTML Help) Project Settings

Help 2.0 or HTML

When true, generates a separate HTML file for each topic. When false, generates a single HTML file containing a named anchor for each topic.

This property can be set as default in the Project Settings properties pane. Modifying this property for an individual document will override the global settings for that document only.

HTML Help

When the SplitFile and the related SplitFileWithHHCtrl properties are true, a single HTML file is generated using intrinsic HTML Help commands to give the appearance of separate HTML files for each topic. When the SplitFile property is true and the SplitFileWithHHCtrl property is false, a separate HTML file is generated for each topic in the document.

This property can be set as default in the Project Settings properties pane. Modifying this property for an individual document will override the global settings for that document only.

Location (Documents)

- Click the **Project Icon**.
- Make sure you have selected the Help 2.0, the HTML, or the HTML Help target from the **Help Target Dropdown**.
- Select the **Documents** item from the left pane Tree View.
- The property is located under the Misc group in the **Properties Pane**.

Location (Project Settings)

- Click the **Project Icon**.
- Select the **Project Settings** item from the left pane Tree View.
- The property is located under the Document Defaults group in the **Properties Pane**.

SplitFileWithHHCtrl Property

When true, a single HTML file is generated using intrinsic HTML Help commands to give the appearance of separate HTML files for each topic in the document. When false, individual HTML files are generated for each topic in the document.

Type	Applies To
Boolean	Document (HTML Help) Project Settings

Remarks

This property can be set as default in the Project Settings properties pane. Modifying this property for an individual document will override the global settings for that document only.

The SplitFile property must be true for this property to be functional.

Location (Documents)

- Click the **Project Icon**.
- Make sure you have selected the HTML Help target from the **Help Target Dropdown**.
- Select the **Documents** item from the left pane Tree View.

- The property is located under the Misc group in the **Properties Pane**.

Location (Project Settings)

- Click the **Project Icon**.
- Select the **Project Settings** item from the left pane Tree View.
- The property is located under the Document Defaults group in the **Properties Pane**.

StemPercent Property

Determines the number of characters (as a percentage of the total) that are added to the root word to create a new word group when building natural language search indexes.

Type	Applies To
Integer	Help Target (HTML) Help Target (HTML Help) Help Target (WinHelp)

Remarks

The default value for this property is 33. That is, 1 character is added to the root word for every 3 additional characters in the word. Some international languages, where lengthy suffixes are common, may benefit from a higher percentage.

This property is only available in Doc-To-Help Pro.

Location

- Click the **Project Icon**.
- Select the **Help Targets** item from the left pane Tree View.
- The property is located under the Natural Search group in the **Properties Pane**.

StemSize Property

Determines the maximum number of characters used as a root word when building natural language search indexes.

Type	Applies To
Integer	Help Target (HTML) Help Target (HTML Help) Help Target (WinHelp)

Remarks

The default value of this property is 5. Highly technical documentation, where longer search terms are common, may benefit from a longer stem size.

This property is only available in Doc-To-Help Pro.

Location

- Click the **Project Icon**.
- Select the **Help Targets** item from the left pane Tree View.
- The property is located under the Natural Search group in the **Properties Pane**.

StringContents Property

Allows modification of the **Contents** string value.

Type	Applies To
String	Help Target (HTML) Help Target (JavaHelp)

Location

- Click the **Project Icon**.
- Make sure you have selected the HTML target from the **Help Target Dropdown**.
- Select the **Help Targets** item from the left pane Tree View.
- The property is located under the Strings group in the **Properties Pane**.

StringFavorites Property

Allows modification to the Favorites string value.

Type	Applies To
String	Help Target (JavaHelp)

StringFoundMany Property

Allows modification of the **%d topics found:** string value.

Type	Applies To
String	Help Target (HTML) Help Target (JavaHelp)

Location

- Click the **Project Icon**.
- Make sure you have selected the HTML target from the **Help Target Dropdown**.
- Select the **Help Targets** item from the left pane Tree View.
- The property is located under the Strings group in the **Properties Pane**.

StringFoundOne Property

Allows modification of the **1 topic found:** string value.

Type	Applies To
String	Help Target (HTML) Help Target (JavaHelp)

Location

- Click the **Project Icon**.
- Make sure you have selected the HTML target from the **Help Target Dropdown**.
- Select the **Help Targets** item from the left pane Tree View.
- The property is located under the Strings group in the **Properties Pane**.

StringFoundZero Property

Allows modification of the **No topics found.** string value.

Type	Applies To
String	Help Target (HTML) Help Target (JavaHelp)

Location

- Click the **Project Icon**.
- Make sure you have selected the HTML target from the **Help Target Dropdown**.
- Select the **Help Targets** item from the left pane Tree View.
- The property is located under the Strings group in the **Properties Pane**.

StringIndex Property

Allows modification of the **Index** string value.

Type	Applies To
String	Help Target (HTML) Help Target (JavaHelp)

Location

- Click the **Project Icon**.
- Make sure you have selected the HTML target from the **Help Target Dropdown**.
- Select the **Help Targets** item from the left pane Tree View.
- The property is located under the Strings group in the **Properties Pane**.

StringNext Property

Allows modification of the **Next** string value.

Type	Applies To
String	Help Target (HTML) Help Target (HTML Help) Help Target (JavaHelp)

Location

- Click the **Project Icon**.
- Make sure you have selected the HTML target from the **Help Target Dropdown**.
- Select the **Help Targets** item from the left pane Tree View.
- The property is located under the Strings group in the **Properties Pane**.

StringNoMatch Property

Specifies the text used in the message box when ComponentOne Natural Search does not find a match for a question.

Type	Applies To
String	Help Target (HTML) Help Target (HTML Help) Help Target (WinHelp)

Remarks

This property is only available in Doc-To-Help Pro.

Location

- Click the **Project Icon**.
- Select the **Help Targets** item from the left pane Tree View.
- The property is located under the Natural Search group in the **Properties Pane**.

StringPrevious Property

Allows modification of the **Previous** string value.

Type	Applies To
String	Help Target (HTML) Help Target (HTML Help) Help Target (JavaHelp)

Location

- Click the **Project Icon**.
- Make sure you have selected the HTML target from the **Help Target Dropdown**.
- Select the **Help Targets** item from the left pane Tree View.
- The property is located under the Strings group in the **Properties Pane**.

StringReturnToIndex Property

Allows modification of the **Return to Index** string value.

Type	Applies To
String	Help Target (HTML) Help Target (JavaHelp)

Location

- Click the **Project Icon**.
- Make sure you have selected the HTML target from the **Help Target Dropdown**.
- Select the **Help Targets** item from the left pane Tree View.
- The property is located under the Strings group in the **Properties Pane**.

StringSearch Property

Allows modification of the **Search** string value.

Type	Applies To
String	Help Target (HTML) Help Target (JavaHelp)

SuperTitle Property

Specifies the text that appears above the title string on the title page of printed manuals.

Type	Applies To
String	Help Target (Manual)

Location

- Click the **Project Icon**.
- Make sure you have selected the Manual target from the **Help Target Dropdown**.
- Select the **Help Targets** item from the left pane Tree View.
- The property is located under the Title Page group in the **Properties Pane**.

Template Property

Specifies the name of the document template applied to intermediate files during compilation.

Type	Applies To
Template	Help Target

Location

- Click the **Project Icon**.
- Select the **Help Targets** item from the left pane Tree View.
- The property is located under the Misc group in the **Properties Pane**.

TextPopups Property

If true, generates a plain text only version of the help file for context sensitive help topics.

Type	Applies To
Boolean	Document (HTML Help) Project Settings

Remarks

This property can be set as default in the Project Settings properties pane. Modifying this property for an individual document will override the global settings for that document only.

Location (Documents)

- Click the **Project Icon**.
- Make sure you have selected the HTML Help target from the **Help Target Dropdown**.
- Select the **Documents** item from the left pane Tree View.
- The property is located under the Misc group in the **Properties Pane**.

Location (Project Settings)

- Click the **Project Icon**.
- Select the **Project Settings** item from the left pane Tree View.
- The property is located under the Document Defaults group in the **Properties Pane**.

Theme Property

Determines the appearance of the Help pane and framesets for HTML-based targets.

Type	Applies To
Folder	Help Target (Help 2.0) Help Target (HTML) Help Target (HTML Help) Help Target (JavaHelp)

Remarks

You can use the `BackgroundColor`, `BackgroundImage`, and `BackgroundRepeat` properties in conjunction with the selected theme to customize the appearance of all topics.

Certain themes, such as `Nonscrolling`, will only work with Internet Explorer 5.0 or greater.

Location

- Click the **Project Icon**.
- Make sure you have selected the HTML or HTML Help target from the **Help Target Dropdown**.
- The property is located under the Appearance group in the **Properties Pane**.

Title Property

Represents the title string of an object.

Type	Applies To
String	Help Target (Manual) Help Target (WinHelp) Topic

Remarks

For printed manuals, this property sets the text that appears below the SuperTitle string on the title page.

For topics, this property is read-only.

For the WinHelp target, this property represents the string to be displayed in the caption bar of the help window.

Location (Help Target)

- Click the **Project Icon**.
- Make sure you have selected the Manual or WinHelp target from the **Help Target Dropdown**.
- Select the **HelpTargets** item from the left pane Tree View.
- The property is located under the Title Page group in the **Properties Pane**.

Location (Topic)

- Click the **Topics Icon**.
- The property is located under the Misc group in the **Properties Pane**.

Top Property

Determines the location of the top edge of the help window.

Type	Applies To
Integer	Window (HTML Help) Window (WinHelp) Window (JavaHelp)

Remarks

In conjunction with the `Left` property, determines the position of the help window on the screen.

Location

- Click the **Project Icon**.
- Select the **Windows** item from the left pane Tree View.
- The property is located under the Position group in the **Properties Pane**.

TopicColor Property

Displays the current color setting for the topic help window. To change the color, click the adjacent change button to open a standard color selection dialog.

Type	Applies To
Long	Window (WinHelp)

Location

- Click the **Project Icon**.
- Make sure you have selected the WinHelp target from the **Help Target Dropdown**.
- Select the **Windows** item from the left pane Tree View.
- The property is located under the Appearance group in the **Properties Pane**.

TopicsFound Property

Determines whether ALink/KLink topic lists are displayed in a dialog box (the default) or a popup menu.

Type	Applies To
Enum	Help Target (HTML Help)

Remarks

This property controls the appearance of links created with the Add Dynamic Link command or character styles. It does not override <OBJECT> tags for the HTML Help ActiveX control.

Location

- Click the **Project Icon**.
- Make sure you have selected the Html Help target from the **Help Target Dropdown**.
- Select the **Help Targets** item from the left pane Tree View.
- The property is located under the Display group in the **Properties Pane**.

TopicType Property

Specifies a named set of display, navigation, and indexing characteristics to be associated with an object (such as what window the help topic appears in, how the help topic is accessed, and whether it gets a map number).

Type	Applies To
TopicType	Paragraph Style Topic

Location

- Click the **Project Icon**.
- Select either the **Paragraph Styles** or **Topic Types** item from the left pane Tree View.
- The property is located under the Misc group in the **Properties Pane**.

TriPaneWindow Property

When true, enables the standard tripane format for HTML Help.

Type	Applies To
Boolean	Window (HTML Help)

Location

- Click the **Project Icon**.

- Make sure you have selected the HTML Help target from the **Help Target Dropdown**.
- Select the **Windows** item from the left pane Tree View.
- The property is located under the Appearance group in the **Properties Pane**.

Type Property

Determines the help authoring behavior associated with a style at compile time.

Type	Applies To
Enum	Character Style Paragraph Style

For Character Styles:

Value	Description
None	No hot spot.
Jump	Topic jump hot spot.
Popup	Topic popup hot spot.
Group	Group link hot spot.
Keyword	Keyword link hot spot.

For Paragraph Styles:

Value	Description
Body Text	Topic body.
Heading Text	Topic with no outline level.
Level 1	Topic at outline level 1.
Level 2	Topic at outline level 2.
Level 3	Topic at outline level 3.
Level 4	Topic at outline level 4.
Level 5	Topic at outline level 5.
Level 6	Topic at outline level 6.
Level 7	Topic at outline level 7.
Level 8	Topic at outline level 8.
Level 9	Topic at outline level 9.

Location

- Click the **Project Icon**.
- Select either the **Paragraph Styles** or **Character Styles** item from the left pane Tree View.
- The property is located under the Misc group in the **Properties Pane**.

Untitled Property

When true, the topic title is not displayed in the help window.

Type	Applies To
Boolean	Paragraph Style Topic Type

Location

- Click the **Project Icon**.
- Select either the **Paragraph Styles** or **Topic Type** item from the left pane Tree View.
- The property is located under the Display group in the **Properties Pane**.

URL Property

Specifies the name of the generated .htm file for a topic.

Type	Applies To
String	Topic

Remarks

By default, the name of the .htm file that is produced matches the topic text with spaces, hyphens and underscores omitted.

Location

- Click the **Topics Icon**.
- The property is located under the Miscellaneous group in the **Properties Pane**.

URLMode Property

Specifies the rules used for naming the .htm files generated for topics.

Type	Applies To
Enum	Project Settings

Value	Description
Full	File name contains all letters and digits from the topic title, even non-ASCII national alphabet letters.
ASCIIOnly	Non-ASCII characters are removed from the file name. File name complies with URL standard.
TopicId	File name is formed from the numeric topic ID.

Remarks

These rules apply when the topic URL is generated automatically, building the project after the topic was first added.

By changing the URLMode property value, you can optionally apply the new rules to all existing topics.

You can modify the URL property of a topic later, manually overriding the default rules.

Location

- Click the **Project Icon**.
- Select the **Project Settings** item from the left pane Tree View.
- The property is located under the Misc group in the **Properties Pane**.

Value Property

Value portion of an arbitrary named value pair assigned to topics or help targets.

Type	Applies To
String	Attribute

Location

- Click the **Project Icon**.
- Select the **Attributes** item from the left pane Tree View.

- The property is located under the Misc group in the **Properties Pane**.

Width Property

Determines the width of the help window.

Type	Applies To
Integer	Window (HTML Help) Window (WinHelp) Window (JavaHelp)

Location

- Click the **Project Icon**.
- Select the **Windows** item from the left pane Tree View.
- The property is located under the Position group in the **Properties Pane**.

Window Property

Shows the name of the window in which topics formatted in this style are displayed.

Type	Applies To
Window	Character Style Paragraph Style Topic Type

Location

- Click the **Project Icon**.
- Select either the **Paragraph Styles**, **Character Styles** or **Topic Type** item from the left pane Tree View.
- The property is located under the Display group in the **Properties Pane**.

WinHelpMacro Property

Specifies a macro to run when a Help file, topic, or secondary window is opened.

Type	Applies To
String	Help Target (WinHelp) Topic Window

Remarks

The WinHelpMacro property set for the WinHelp target will override the WinHelpMacro property set for the “main” window.

This property only applies to the WinHelp platform.

Location (WinHelp)

- Click the **Project Icon**.
- Make sure you have selected the WinHelp target from the **Help Target Dropdown**.
- Select the **Help Targets** item from the left pane Tree View.
- The property is located under the Misc group in the **Properties Pane**.

Location (Topics)

- Click the **Topics Icon**.
- The property is located under the Misc group in the **Properties Pane**.

Location (Windows)

- Click the **Project Icon**.
- Make sure you have selected the WinHelp target from the **Help Target Dropdown**.
- Select the **Windows** item from the left pane Tree View.
- The property is located under the Misc group in the **Properties Pane**.

Index

A

- Active Character Style 197
- ActiveX Control 271
- Add Topic Link Dialog Box 203
- Adding
 - Attributes 307
 - Character Style 104
 - Conditional Text 301
 - Context IDs 190
 - Context Strings 411
 - Document 67, 134
 - Glossary Entries 84, 267
 - Group 229
 - Help Targets 119
 - Jumps 74, 198
 - Keywords 221, 223, 227
 - Link Tags 178, 179
 - Links 74
 - Margin Notes 207
 - Paragraph Styles 103
 - Scripts 334
 - Source Documents 67
 - Styles 103, 104, 147, 150
 - Tables 72
 - Templates 138
 - Topic Types 158
 - Topics 183
 - Windows 43
- AdjustForScreenSize property 443
- AdjustLeftIndent property 443
- AffectsAppearance property 444
- ALinks 201
- AlwaysOnTop property 444
- AnswerCount property 445
- Anti-virus 16
- AsciiName property 445
- Attributes 42, 299
 - Adding 307
 - Using 305
- Authoring Templates 16
- AutoButtons property 446
- AutoContextID property 447
- AutoHideNavigationPane property 447
- AutoIndex property 448
- AutoKeyword property 448
- AutoLink property 448
- AutoNavigate property 449
- AutoNext property 449
- AutoSizeHeight property 450
- AutoSyncNavigationPane property 450

B

- Back Button 494
- Background 242
 - Color 242
 - Image 243
- BackgroundColor property 450
- BackgroundImage property 451
- BackgroundRepeat property 451
- BaseName property 452
- BinaryIndex property 452
- BinaryTOC property 453
- Books 183
 - Creating 183
 - Renaming 185
- Browse Buttons 495
- Build Icon 48
- ButtonLabel property 453
- ButtonLabelStyle property 454
- ButtonSeparator property 454
- ButtonStyle property 455
- ByLine property 455

C

- CancelKey property 456

- CancelMap property 456
 - CancelTag property 456
 - Caption property 457
 - Cascading Style Sheets 163
 - Character Styles 135, 149
 - Active 197
 - Applying 104
 - Creating 150
 - Default 150
 - Properties 149, 151
 - Charset property 457
 - Citation property 458
 - Code property 458
 - CodePage property 458
 - ColorReduction property 459
 - Comments property 460
 - Compacting 110
 - Compiling
 - Glossary Entries 267
 - Source Documents 141
 - Compression property 460
 - Concatenate property 461
 - Conditional Text 299, 300
 - Adding 85, 301
 - Applying 86, 87, 88
 - Editing 303
 - Font Color 311
 - Format 311, 313
 - Printed Manual 86
 - Removing 303
 - Specifying 299
 - View 309, 310
 - ConfidenceLevel property 461
 - Contents Button 495
 - Contents Icon 46
 - ContentsOnly property 462
 - ContentsTab property 462
 - ContentsTitle property 462
 - Context IDs 186, 189
 - Adding 190
 - Generating 188
 - Context Sensitive Help 186
 - Context Strings 411
 - Adding 411
 - Linking Topics 413
 - ContextString property 463
 - Converting 54
 - Doc-To-Help 2000 54, 111
 - True Help 114
 - CopyFolder property 304, 463
 - Copying
 - Data 264
 - Styles 64
 - Copyright property 464
 - Creating
 - ALinks 201
 - Character Styles 150
 - Context Sensitive Help 188
 - Cross References 204
 - Hot Spots 252
 - Jumps 198
 - Keywords 221, 337
 - KLinks 200
 - Modular Help 319, 324
 - Paragraph Styles 147
 - PopupTopics 199
 - Projects 115
 - Scripts 332
 - Styles in Word 102
 - Topic Groups 228, 338
 - Windows 237
 - Cross Reference 79, 204, 205
- ## D
- DefaultFile property 464
 - DefaultTemplate property 464
 - DefaultWindow property 465
 - Deleting
 - Help Topics 271, 442
 - Margin Notes 210, 211
 - Project Elements 265

DisplayTitle property 465

Distributing Files 14

Doc-To-Help 7.2 53

 Licensing 14

 Shortcut Keys 295

 Support 15

Documenter for .NET 341

 Controls 357

 Document 360

 Link Colors 385

 Narrative 351

 Project 345

 Special Characters 383

 Styles 371, 372, 373, 374, 375, 376, 377, 378,
 379, 380, 381

 Toolbar 358

 Tutorial 342

Documents 42, 133

 Adding 67, 134

 Compiling 141

 Opening 140

 Order of 137

 Source 141

Dynamic Link 215

Dynamic Links 215

 Adding 215

DynamicTOC property 466

E

Editing

 Character Styles 151

 Conditional Text Marks 303

 Glossary 266

 Help Contents 181

 Help Window 238

 Hyperlink Fields 258

 Macros 289

 Paragraph Styles 148

 Scripts 333

 Topic 177

 Topic Type 158

Existing Projects

 Converting 54, 111

ExplicitAccess property 466

F

FavoritesTab property 466

Find Button 495

FixLists property 467

Folder property 468

Forward Button 496

Frameset property 468

G

GenerateCSS property 468

GenerateProjectFile property 469

Global property 469

Global Template 18

Glossary

 Editing 266

 Opening 266

 Sorting 270

 Terms 84

Glossary property 470

Graphics 249

 Hot Spots 252

 Image Map 397

 Inserting 249

 Scaling 76, 249

Group Links 201, 229

Groups

 Assigning Topics 229

 Renaming 228

Guided Tour 53, 342

H

Height property 470

Help 2.0

 Context Strings 411

Help Authoring 21

Help Contents

 Adding Topics 183

 Editing 180, 181

 Organizing 28, 182

 Saving 185

 Specifying 170

Help Project

 Creating 319, 324

 Documents 67

 Starting 55

Help Systems

 Building 26

 Defined 21

 Managing 317

Help Target 44, 118

 Adding 119

 Building 118

 Changing 118

 Directory 120

 Viewing, current 118

Help Topics Button 496

Help Windows 233

Hidden property 470

HideNavigationPane property 471

Home Button 497

Hot Spots 197, 252, 255, 397

HTML Help 125

 ActiveX control 271

HTML Help Object Tags 271

HTML Passthrough Code 303, 304

HubContents property 471

Hyperlink 256

 Editing 258

 Inserting 257

I

ID property 472

IgnoreFontNames property 472

IgnoreFontSizes property 473

Image Map 397

Images 249

Importing

 Styles 151, 153

In Place Editing 140

Index 23, 28

 Building 219

 Constructing 28

 Keyword 221, 223, 224, 227

Index Icon 47

Index Keywords

 Adding 221, 223

 Creating 221, 337

 Secondary 223

Indexing 28, 92

IndexTab property 473

Installation 16

Internet Sites 255

J

JavaHelp 34, 130

Jump Button 473, 474

Jump1Caption property 473

Jump1URL property 474

Jump2Caption property 474

Jump2URL property 474

Jumps 174

K

KeepOutlineNumbers property 475

KeepPageBreaks property 476

Keyboard Shortcuts 295

Keyword

 Adding 221, 223, 227, 337

 Assigning Topics 224

 Renaming 224

KLinks 200

L

- Left property 476
- License 14
- Link Tags 178, 179
- Linking
 - Internet Sites 255
 - Margin Notes 82, 208
 - Topics, related 30
- Links 173, 197
 - Alinks 201
 - Dynamic 215
 - Klinks 200
 - Next Topic 176
 - Subtopics 172, 174
- Lists 90, 161
 - Applying 90
 - Creating 90
 - Nesting 162
 - Printed Manual 163
- Locale property 477
- Locate Button 499

M

- Macros 273
 - Deleting 289
 - Hotspot 274, 275, 276
 - Startup 278, 283
 - Topic 286
- Map Numbers 188, 189, 190, 447
- MapFileForC property 477
- MapFileForVB property 478
- MapNumberOffset property 478
- MapNumbers property 478
- Margin Notes 81, 207
 - Adding 207
 - Creating 82
 - Definitions, deleting 211
 - Linking 208
 - Links, clearing 212
 - Removing 209

- Maximized property 479
- MergeContents property 480
- MergeFile property 480
- MergeTitle property 480
- Microsoft Help 2.0
 - Context Strings 411, 413
- Microsoft Word 99, 101, 102
- MidTopic Jump 213
- MidTopic property 481
- Modular Help 317
 - Creating 319, 324
 - Managing 317, 478
 - Skip Glossary 502
 - TOC Utility 403
- Modular TOC Utility 403
- ModularHub property 481
- MultiLink property 482

N

- Name property 482
- Namespace property 483
- Natural Search 387, 388
 - Content 388
 - Distribution 394
 - No Match Message 391
 - Noise List 394
 - Properties 387
 - AnswerCount 445
 - ConfidenceLevel 461
 - SearchCaption 492
 - SearchEnabled 492
 - StemPercent 504
 - StemSize 504
 - StringNoMatch 507
 - Returns 392
 - Searches 394
 - Support 393
 - Tab Caption 390
- Navigation 39, 69
- Navigation Sequence 171

NavigationPaneWidth property 483
Next Button 499
NonScrolling property 483
NonscrollingColor property 484
Numbered List 90

O

OneBrowseSequence property 484
OnlineOnly property 485
OnPage property 485
Opening

- Files 140
- Glossary 266

Options Button 499
Order property 485
Organizer 151, 153

P

PageRef property 486
Paragraph Style 43

- Active 169
- Applying 103
- Creating 147
- Editing 148
- Properties 148

Parent property 486
ParentNamespace property 487
Picture 77, 252
Popup property 487
Popup topics 199
PrefixIndexListFile property 488
PrefixTopicListFile property 488
Previous Button 500
Print Button 500
Print Preview 261
Printed Manual 38, 127

- Creating 96
- Examining 96
- Lists 163

Printing

Current view 261
Previewing 261
Reports 262
Project Editor 39
Project Files

- Compacting 110
- Managing 264
- Organizing 133

Project Icon 41
Projects 107

- Removing files 141
- Settings 44
- Toolbar 41

Properties

- Character Style 149, 151, 371
- Document 134
- Help Target 120
- Help Window 238
- Paragraph Style 143, 148
- Project Settings 116
- Topic 167, 177
- Topic Type 155, 158
- Window 233, 236

R

Redistributable Files 14
Refresh Button 501
RelatedJumps property 488
Renaming

- Books 185
- Groups 228
- Keywords 224
- Topics 185

Replacement property 489
Reports 260, 263

- Generating 260
- Selecting 262

S

SaveUserPosition property 489

- ScaleFontSizes property 490
 - ScaleGraphics property 491
 - Script property 491
 - Scripts 43, 329, 332
 - Debugging 334
 - Defined 329
 - Editing 333
 - Module 333
 - Operations 335
 - Techniques 329
 - SearchCaption property 492
 - SearchEnabled property 492
 - SearchTab property 493
 - SearchTabAdvanced property 494
 - Secondary Keywords 223, 337
 - Shortcut Keys 295
 - ShowBackButton property 494
 - ShowBrowseButtons property 495
 - ShowContentsButton property 495
 - ShowFindButton property 495
 - ShowForwardButton property 496
 - ShowHelpTopicsButton property 496
 - ShowHideShowButton property 496
 - ShowHomeButton property 497
 - ShowIndexButton property 497
 - ShowJump1Button property 498
 - ShowJump2Button property 498
 - ShowLocateButton property 499
 - ShowNextButton property 499
 - ShowOptionsButton property 499
 - ShowPreviousButton property 500
 - ShowPrintButton property 500
 - ShowRefreshButton property 501
 - ShowStopButton property 501
 - SkipGlossary property 502
 - Sorting
 - Glossary 270
 - Topics 69, 192, 193
 - SplitFile property 502
 - SplitFileWithHHCtrl property 503
 - Standard Table 253
 - StemPercent property 504
 - StemSize property 504
 - Stop Button 501
 - StringContents property 505
 - StringFavorites property 505
 - StringFoundMany property 505
 - StringFoundOne property 506
 - StringFoundZero property 506
 - StringIndex property 507
 - StringNext property 507
 - StringNoMatch property 507
 - StringPrevious property 508
 - StringReturnToIndex property 508
 - StringSearch property 509
 - Styles 60, 102, 143
 - Assigning Scripts 334
 - Assigning Windows 237
 - Character 104, 149
 - Creating 102
 - Documenter 351, 372, 373, 374, 375, 376, 377, 378, 379, 380, 381
 - Importing 153
 - Link 372, 373, 374, 375, 376, 377, 378, 379, 380, 381
 - List 161
 - Paragraph 143, 169
 - Subtopics 172, 183
 - Labels 174, 175
 - SuperTitle property 509
 - Support 15
 - System Requirements 16
- ## T
- Tables 72
 - Insert 253
 - Technical Support 15
 - Template property 509
 - Templates 133
 - Adding 138
 - Authoring 16
 - TextPopups property 509

- Theme property 510
- Themes 239, 417
 - Designer 417
- Title property 511
- Toolbars 48
 - Components 48
 - Contents 46, 181
 - Index 47, 220
 - Project 41
 - Project Properties 42
 - Topic Properties 45
- Top property 511
- Topic Groups 228
- Topic Tools 437
 - Add Topic Button 437
 - Delete Topic Button 442
 - Rename Topic Button 440
- Topic Types 43, 154
 - Adding 158
 - Applying 159, 160
 - Default 157
- TopicColor property 512
- Topics 74, 173
 - Customizing 174
 - Defining 27, 165
 - Filtering 194
 - Found 202
 - Popup 199
 - Properties 177
 - Removing 230
 - Sorting 192
- Topics Icon 45
- TopicsFound property 512
- TopicType property 513
- TriPaneWindow property 513
- TrueHelp 114
- Tutorial 53
- Type property 514

U

- Untitled property 515
- URL property 515
- URLMode property 516

V

- Value property 516
- View
 - Conditional Text 309, 310
- View Conditional Text 309
- Views Icon 44

W

- Width property 517
- Window 237
- Window property 517
- Windows 43
 - Background 242
 - Customizing 233, 246
 - Editing 245
 - Themes 239
- WinHelp 4.0 128, 236
- WinHelpMacro property 517