madcap
FLARE™

Transition From RoboHelp Guide

Version 4.0

Copyrights

MadCap Software
7777 Fay Avenue
La Jolla, California 92037
858-320-0387
www.madcapsoftware.com

Contents

Introduction **1**
Terminology Differences **3**
Interface Differences **5**
Accordions 6
Content editors 7
Dialogs 10
Menu items 10
Toolbars 10
Toolbar buttons 11
Replicating the RoboHelp Interface **13**
File and Feature Location Differences **17**
Location of files 17
Locations for adding elements 19
Locations of editors and window panes 19
Location of content tags ("true code") 19
Feature Differences **21**
Importing Tips **25**
Index **31**

Introduction

The following sections are designed to help RoboHelp users transition to Flare.

- **Terminology differences** This section provides a list of common terms in RoboHelp and their counterparts (if different) in Flare. See "Terminology Differences" on page 3.
- **Interface differences** This section discusses some of the differences that you will see between the user interfaces in RoboHelp and Flare. See "Interface Differences" on page 5.
- **Replicating the RoboHelp interface** This section gives you steps for replicating the RoboHelp interface as close as possible by moving similar elements to the same locations as you are used to in RoboHelp. See "Replicating the RoboHelp Interface" on page 13.
- **File and feature location differences** This section explains how the location of files and features is often different in RoboHelp and Flare. See "File and Feature Location Differences" on page 17.
- **Feature differences** This section lists some of RoboHelp's features and gives a brief explanation of the same or similar feature in Flare where substantial differences exist. See "Feature Differences" on page 21.
- **Importing tips** This section provides bits of information that you might find useful when importing your RoboHelp project to Flare. It explains what happens to certain RoboHelp features in Flare when a project is imported. See "Importing Tips" on page 25.
- **Video tutorial - Flare Compared with RoboHelp** This 45-minute video tutorial provides you with a guided tour through tasks and features in RoboHelp and Flare, showing you the similarities and differences between the two applications. To open the video tutorial, see the online Help.

Note: In these sections, Flare is compared with RoboHelp HTML X5, not necessarily with every version of RoboHelp.

Terminology Differences

RoboHelp and Flare share many similar concepts, but the terminology used is sometimes different. Following are some examples.

RoboHelp Term	Flare Term
Conditional build tag	Condition tag
Conditional build tag expression	There is no specific term for this. You simply associate your condition tags with a target.
Default topic	Startup topic
Drop-down hotspot	Drop-down head (or drop-down hotspot)
Edit Map IDs dialog (All Map IDs option)	Alias Editor (used to open an alias file and associate identifiers with topics for context-sensitive Help)
Glossary Designer	Glossary Editor
Horizontal line	Ruler
Index Designer	There is no index designer or editor in Flare. Instead, the index is generated when you build your target, based on index keywords that you insert into topics.
Map file	Header file
Map ID	Identifier
Map number	Identifier value
Page (TOC)	Item or entry (TOC)
Popup	Topic popup
Primary layout	Primary target
Project Manager	The Project Organizer and Content Explorer serve the same purpose.

RoboHelp Term	Flare Term
Ruler	Horizontal ruler or vertical ruler
See Also link control	Concept link control
Single source layout	Target In Flare, a "layout" can refer to the configuration of the user interface. You can move interface elements around in Flare and save the layout (**Window>Layouts>Save Window Layout As**). The next time you open that layout in Flare, the elements are just where you want them. A layout can also refer to a page layout, which is an element used to help create page configurations for print-based output.
Skin	Skin Flare does not have skins in the same sense as RoboHelp. Instead, it has some of the same features from RoboHelp skins and merges them with features that are similar to those in RoboHelp windows. These are known in Flare as "skins."
Subkeyword	Second-level index keyword
Text-only popup	Popup
TOC Composer	TOC Editor
Topic templates	Master pages are similar, but not identical.
True Code Editor	Internal Text Editor In addition, you can open Flare topics in other editors, such as Notepad.
Window	Skin (merges the concepts of "windows" and "skins" in RoboHelp)
WYSIWYG Editor	XML Editor

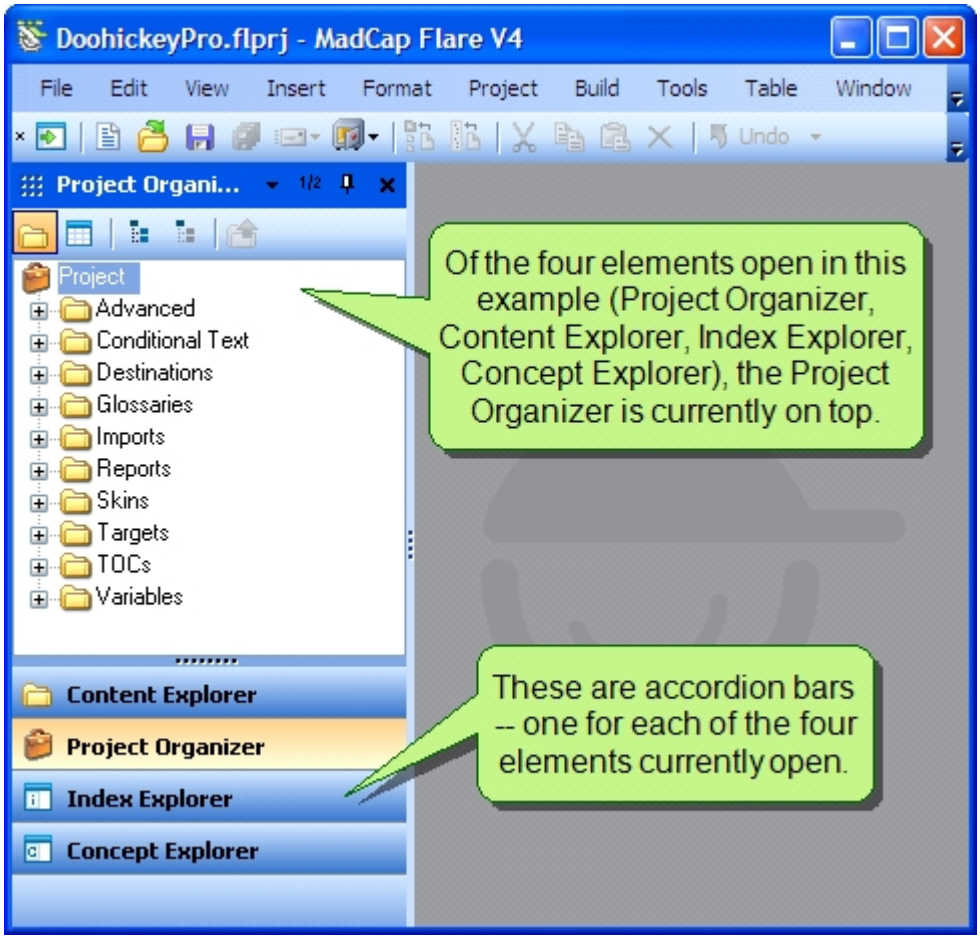
Interface Differences

In some ways, RoboHelp and Flare are alike. Both have some similar menu bars and toolbars. Both have panes on the left side of the interface (by default) where topic and project files are located. And both have a content editor where you enter topic content.

However, in many other ways, RoboHelp and Flare look quite different from each other. Following are some of these user interface differences.

Accordions

RoboHelp has some non-movable embedded panes on the left (Project Manager, TOC Composer, Index Designer, Glossary Designer, Tools) and on the right side of the program window (WYSIWYG, TrueCode Editor, Link View, Topics List). On the other hand, Flare displays elements in an accordion structure, located (by default) on the right, left, or bottom edge of the interface (depending on the elements that you open). Clicking a particular accordion bar brings that element to the front. In addition, you can close any of these elements, or move them, whenever you want.

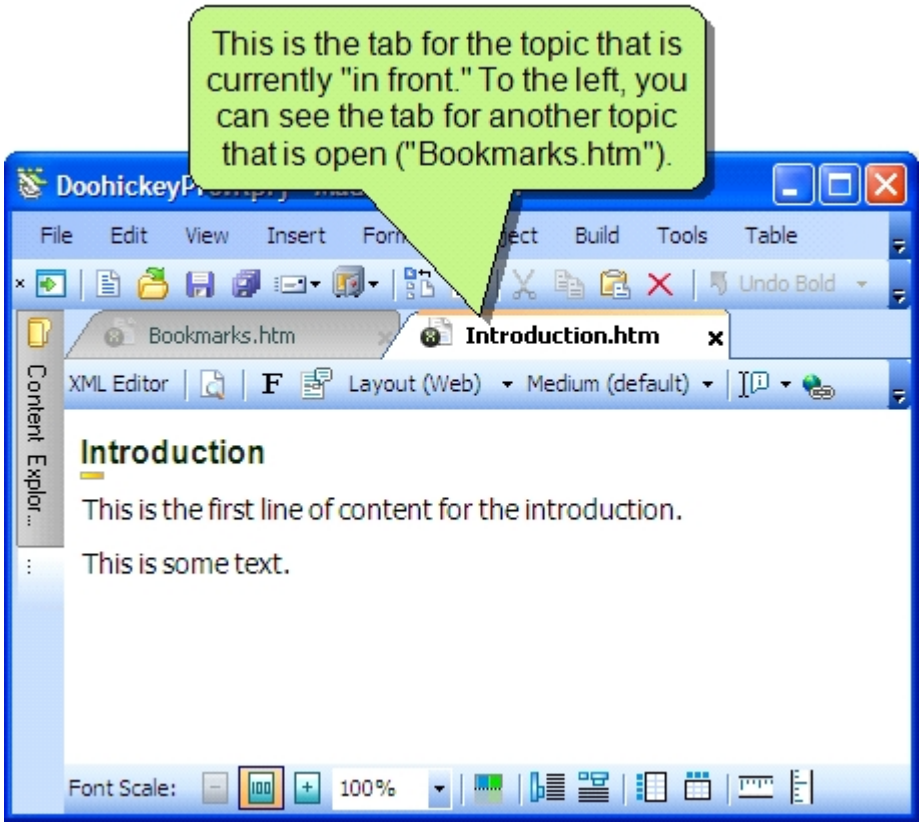


Content Editors

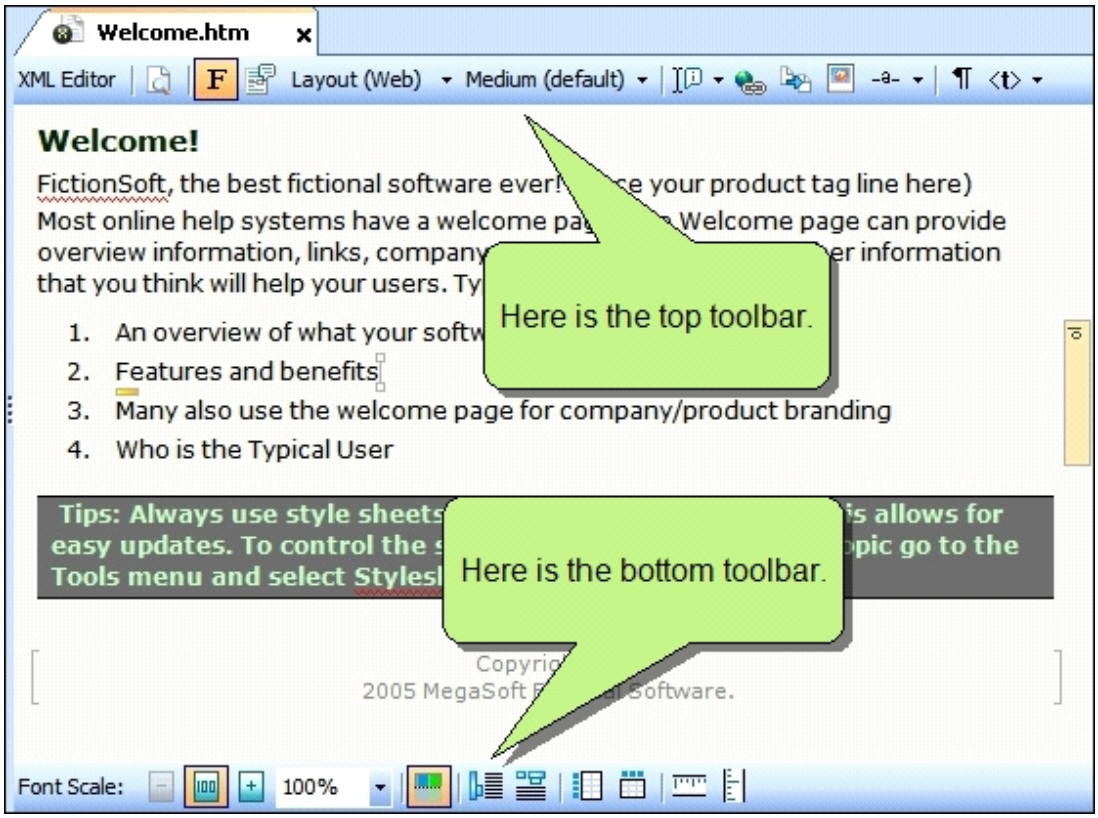
In RoboHelp, you use the WYSIWYG Editor to work on topics. In Flare, you use the XML Editor. Flare's XML Editor looks different from RoboHelp's WYSIWYG in a few ways.

First, the most obvious difference is that in RoboHelp you edit content in HTML. In Flare, you use XML (or XHTML).

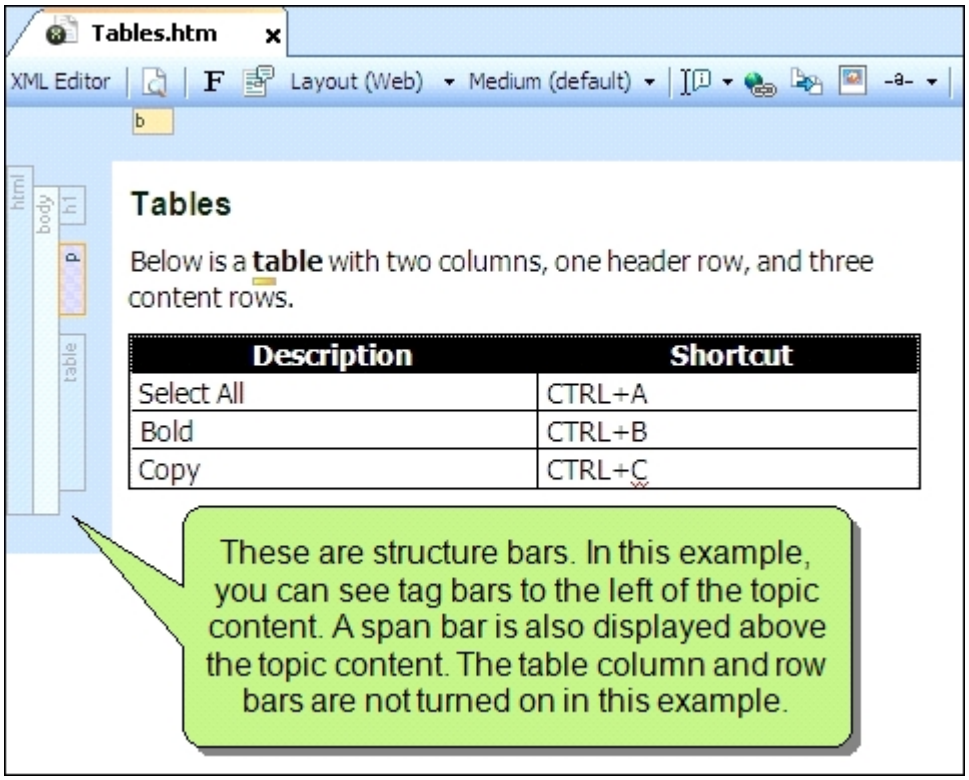
Second, you can have multiple topics open at the same time in the XML Editor. Each topic is displayed in its own instance of the XML Editor. The XML Editor has a tab at the top with the name of the topic, which can be used to bring a particular topic to the front when other topics or other editors are also open.



Second, the XML Editor has a two local toolbars—one at the top and the other at the bottom. These toolbars contain shortcuts to actions that are specific to the editor.



Third, the XML Editor has structure bars that can be turned on or off. When the structure bars are on, they appear either to the left or above the topic content (depending on the type of structure bar you display).



Dialogs

RoboHelp and Flare have many similar features that are implemented by using dialogs. However, you will notice that the dialogs look somewhat different in Flare than they do in RoboHelp, even if the features are similar. For example, in the dialog used to create a text hyperlink (Hyperlink dialog in RoboHelp, Insert Hyperlink dialog in Flare), RoboHelp uses a drop-down list, whereas Flare uses radio buttons.

Menu Items

RoboHelp and Flare share many of the same menu items, but they each have unique ones as well. The different menu items that might stand out most to you in Flare are the **Project** and **Build** menus. The Project menu lets you add to your project most of the different kinds of elements available in Flare (e.g., topic, TOC, snippet, style sheet, skin, target). The Build menu provides options for building, viewing, or publishing your primary target or the target currently open. It also lets you clean (delete) your output files and open your output folder.

Toolbars

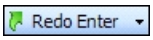
In RoboHelp, you can open seven toolbars, which are accessible from the View menu and displayed by default at the top of the interface. A few components in RoboHelp (e.g., TOC Composer, Index Designer) contain local toolbars.

In Flare, there are multiple global toolbars that you can open from the View menu. The Standard toolbar and Project toolbar in Flare correspond roughly with the Project toolbar in RoboHelp. The Text Format toolbar in Flare corresponds with the Formatting toolbar in RoboHelp. And the Review toolbar is unique to Flare. It lets you perform tasks related to sending topics for review to others or receiving contribution topics.

In addition to its global toolbars, Flare has many local toolbars, which are contained in almost every editor (e.g., XML Editor, TOC Editor, Skin Editor, Glossary Editor) and several other elements (e.g., Project Organizer, Content Explorer, Concept Explorer).

Toolbar Buttons

RoboHelp and Flare share many buttons with the same or similar functions; they just look different from each other. Check out the following examples of some of the more prominent differences in toolbar buttons between RoboHelp and Flare.

Button	RoboHelp	Flare
Save		
Build Primary Target		
View Primary Target		
Create New Topic		
Edit/Open Item		
Preview Topic		
Undo		
Redo		
Insert Picture		
New TOC Book		
New TOC Page/Item		

Replicating The RoboHelp Interface

It's not possible to make the Flare interface look *exactly* like the RoboHelp interface. Although the two applications have many concepts in common, they simply do not share all of the same features. And even if some features are similar, they often work somewhat differently in Flare than they do in RoboHelp. However, it is possible to move some elements around in Flare so that their locations are relatively close to their locations in RoboHelp. You may find that this helps your transition to Flare.

As you know, RoboHelp provides tabs for the Project Manager, TOC Composer, Index Designer, Glossary Designer, and Tools pane on the left. Meanwhile the WYSIWYG Editor, TrueCode Editor, Link View pane, and Topics pane are on the right. Let's compare these to their rough equivalents in Flare. Below these explanations are steps for replicating (as close as possible) the interface in RoboHelp.

- **Project Manager** The rough equivalents to RoboHelp's Project Manager are the Project Organizer and Content Explorer. By default, these elements are already located on the left side of the Flare interface (similar to the Project Manager in RoboHelp). So you can leave them in their default location.
- **TOC Composer** The rough equivalent to RoboHelp's TOC Composer is the TOC Editor. By default, the TOC Editor opens in the large middle section of the Flare interface (where you would be used to editing topics in RoboHelp). In Flare, you can have multiple TOCs open in multiple instances of the TOC Editor. However, you are likely to use one "master" TOC most often (if not exclusively). Therefore, you can move the TOC Editor holding your "master" TOC to the left side of the Flare interface, where you are used to working with the TOC Composer in RoboHelp. You can also hold down the **CTRL** key and double-click topic pages to open them in the XML Editor to the right.
- **Index Designer** There is not really a rough equivalent to RoboHelp's Index Designer in Flare. This is due to the fact that indexing is performed so differently in Flare than it is in RoboHelp. In Flare, there are two window panes that you are likely to use a great deal when indexing—the Index Entry window pane and the Index Explorer. By default, the Index Entry window pane opens on the right side of the Flare interface, and the Index Explorer opens on the left side (where you would expect to find the Index Designer in RoboHelp). Although the purpose of the Index Explorer is different from that of the Index Designer in RoboHelp, you might want to keep it open in its default location on the left. It is probably best to keep the Index Entry window pane in its default location on the right, because you are likely to want to be able to see both the Index Explorer and the Index Entry window pane at the same time when you perform indexing in Flare.

- **Glossary Designer** The rough equivalent to RoboHelp's Glossary Designer is the Glossary Editor. This is a situation similar to the TOC Composer/TOC Editor. In Flare, the default location is the large middle section of the interface, but you can move the editor to the left side where the Glossary Designer is found in RoboHelp.
- **Tools Pane** There is not an exact equivalent to RoboHelp's Tools pane. However, Flare has a Find and Replace window pane, which is similar to the Multiple Find and Replace tool in RoboHelp. By default, the Find and Replace window pane opens on the right side of the interface, but you can move it to the left side where the Tools pane is located in RoboHelp.
- **WYSIWYG Editor** The rough equivalent to RoboHelp's WYSIWYG Editor is the XML Editor. There is no need for you to move the XML Editor because it displays in the same general area as the WYSIWYG Editor in RoboHelp.
- **True Code Editor** The rough equivalent to RoboHelp's True Code Editor is the Internal Text Editor (or other editors, such as Notepad). If you want to view and modify the true code for a topic in Flare (including text, tags, and metadata), right-click on the topic in the Content Explorer and select **Open with>Internal Text Editor**. When you open a topic in the Internal Text Editor in Flare, it displays in the same general area as the True Code Editor in RoboHelp.

  If you want to open the code in another editor, such as Notepad, you can select it from the same context menu. You can also click the **Send To** button  in the Standard toolbar and select another editor from that menu drop-down.
- **Link View pane** There is not an exact equivalent to the Link View pane. However, you can see links to specific files in Flare by right-clicking the file in the Content Explorer and selecting **Show Dependencies**. The linked files are shown in the List Dependencies dialog.

  In addition, the built-in Analyzer (**View>Project Analysis**) lets you see incoming links to a topic. The external Analyzer lets you see both incoming and outgoing links.
- **Topics pane** The rough equivalent to RoboHelp's Topics pane is the File List window pane. By default, the File List window pane opens on the left side of the interface, but you can move it to the right side where the Topics pane is located in RoboHelp.

How to replicate certain elements of the RoboHelp interface

1. From the Project Organizer, open your "master" TOC.
2. Select **Window>Floating**.
3. Click in the title bar of the TOC Editor and drag it to the left side of the interface, dropping it on the bulls eye.

   The TOC Editor is now docked to the left edge of the program window with the Content Explorer and Project Organizer. (You can also hold down the **CTRL** key and double-click topic pages to open them in the XML Editor to the right.)
4. Select **View>Index Explorer**.

   The Index Explorer is now open on the left side of the interface with the other elements.
5. From the Project Organizer, open your glossary.
6. Select **Window>Floating**.
7. Click in the title bar of the Glossary Editor and drag it to the left side of the interface, dropping it on the bulls eye.

   The Glossary Editor is now docked to the left edge of the program window with the other elements.
8. Press **CTRL+F** on your keyboard.

   The Find and Replace window pane opens on the right side of the interface.
9. Click the "drag pane" button (the series of small dots)  in the upper-left corner of the Find and Replace window pane and drag it to the left side of the interface, dropping it on the bulls eye.

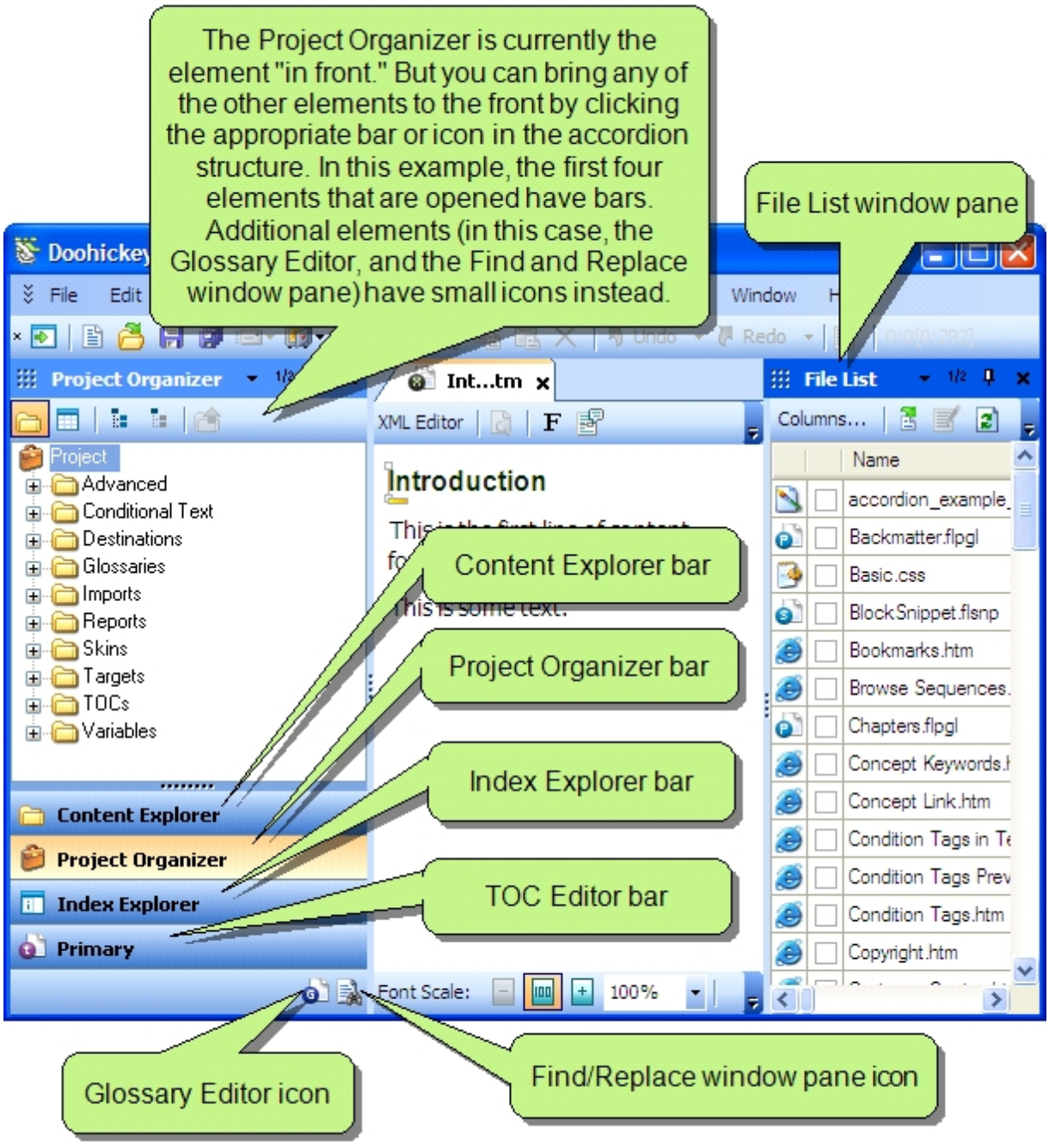
   The Find and Replace window pane is now docked to the left edge of the program window with the other elements.
10. Press **CTRL+SHIFT+J** on your keyboard.

    The File List window pane opens on the left side of the interface.
11. Click the "drag pane" button (the series of small dots)  in the upper-left corner of the File List window pane and drag it to the right side of the interface, dropping it on the blue arrow pointing to the right.

    The File List window pane is now docked to the right edge of the program window.

The Flare interface should now display the rough equivalents to the elements from RoboHelp.

In the following example, the elements open on the left and right sides seem to be squeezing the topic in the XML Editor (which is located in the middle). However, if we were to maximize the program window, we would see much more of the topic and would have additional space to work with.



File And Feature Location Differences

One of the biggest differences between RoboHelp and Flare is the location of files, features, and options. Certain tasks might be similar to those in RoboHelp, but where you access this or that is often different. Following are some of the primary location differences.

Location Of Files

RoboHelp and Flare store project and content files in different locations in the interface. There is also a difference in the way files are organized when you view them in Windows. The following table explains these differences.

	RoboHelp	Flare
Miscellaneous content	Miscellaneous content (e.g., images, multimedia, style sheets) are stored in their own subfolders in the Project Manager. Additional content files need to be placed in the Baggage folder. Some files are added for you, and other files require you to add them manually.	Miscellaneous content files are added automatically to the Resources subfolder in the Content Explorer. There is no Baggage folder, and you do not need to add any files manually. Miscellaneous content files include those for elements such as images, master pages, page layouts, snippets, and style sheets. However, if you import a RoboHelp project, Flare retains the file structure from your imported project. **EXAMPLE** If your style sheet was stored at the root level of your project, it is imported to the root level of the Content Explorer (as opposed to the Resources\Stylesheets subfolder, which is used to hold new style sheets that you create in Flare).

	RoboHelp	Flare
Miscellaneous project elements	Most project files are stored at the root level of the project folder in Windows. Skin files, however, are stored in a folder called !SkinSubFolder!	Miscellaneous project files are stored in the Project subfolder (within the main project folder) in Windows. This includes elements such as auto-index sets, browse sequences, conditional text, glossaries, index link sets, publishing destinations, reports, skins, targets, tables of contents, and variables.
Output files	Output files are stored in the !SSL! subfolder under the project folder in Windows.	The output files are stored in the Output subfolder under the project folder in Windows.
Project file (main)	The main Help project file is stored at the root level of the project folder in Windows.	The main Help project file (FLPRJ) is also stored at the root level of the project folder in Windows.
Topic files	Topic files are stored in the Project Manager under the HTML Files (Topics) folder or under the Topics folder (depending on your version of RoboHelp). In Windows, these files are located at the root level of the project folder.	Topic files are stored in the Content Explorer under the Content folder. In Windows, these files are located in the Content subfolder within the project folder.

Locations For Adding Elements

In RoboHelp, you can add many new elements from the **File>New** menu. Sometimes (but not always) you can right-click on a subfolder in the Project Manager to add a new element. And for other features, there is usually a unique method for adding a new element.

In Flare, most elements in the project can be added by using the **Project** menu. You can also right-click any subfolder in the Project Organizer to add a related element. And by right-clicking on any subfolder in the Content Explorer, you can add a new topic. In addition, context menus throughout the interface let you add various elements.

Locations Of Editors And Window Panes

In RoboHelp, some elements are "embedded" on the left or right side of the interface. On left side of the interface are the Project Manager, TOC Composer, Index Designer, Glossary Designer, and Tools. On the right side are the WYSIWYG Editor, TrueCode Editor, Link View pane, and Topics pane.

In Flare, many elements open in their own editors in the middle of the interface (e.g., topics in the XML Editor, TOCs in the TOC Editor, glossaries in the Glossary Editor). This includes some elements that you might be used to opening in dialog windows in RoboHelp (e.g., browse sequences, style sheets, condition tags). Other elements in Flare open in window panes on the left, right, or bottom edge of the interface. Elements that open on the left side include the Project Organizer, Content Explorer, Index Explorer, Concept Explorer, and more. Elements that open on the right side include the Attributes window pane, Style window pane, Index Entry window pane, Concept Entry window pane, Glossary Terms window pane, Spell Check window pane, and more. Elements that open at the bottom include the Frame Contents window pane, Edit Annotation window pane, Edit Footnote window pane, and more. These window panes are displayed in an accordion structure, and you can open, close, or move them any time you want.

Location Of Content Tags ("true Code")

In RoboHelp, you access the tags to topic content by opening the **True Code Editor**.

In Flare, you right-click on a file in the Content Explorer and open the topic with the **Internal Text Editor**.

Feature Differences

Many tasks in Flare are performed similarly to the way they are done in RoboHelp. Opening a topic, inserting a hyperlink, and inserting a picture are examples of similar tasks (although dialog windows might differ somewhat). In both applications, the overall workflow is somewhat similar. In both applications, you create a project; create topics and other content; develop your outputs (single-source layouts or targets); dress up your project and content using styles, formatting, and skins (windows); generate the output; and distribute it to your users.

However, there definitely are some differences between RoboHelp and Flare as far as features are concerned. The following table lists some features in RoboHelp and explanations of corresponding features in Flare where substantial differences exist.

Feature	RoboHelp	Flare
Drop-Down Text	In RoboHelp, you select the hotspot, select the drop-down option from the **DHTML** menu, and then enter your text into the popup editor.	In Flare, the process for creating drop-down text is quite different. You begin by selecting all of the content in the topic that you want to include in the drop-down effect (the hotspot as well as the drop-down body). Then you select the drop-down option from the **Insert** menu. In the dialog, you highlight the part of the content that you want to use as the hotspot. Unlike RoboHelp, there is no popup editor. Brackets indicate where a drop-down effect exists in the topic.
Editing Language	HTML	XML Similar to RoboHelp, you do not need to know the language to create content in the XML Editor. When you import HTML files, Flare converts them to XHTML for you.

Feature	RoboHelp	Flare
Indexing	In RoboHelp, you "bring the topics to the index keyword." After you create the index keyword in the Index Designer, you drag the related topics to the Index Designer.	In Flare, the process of creating an index is very different. There is no designer or editor for creating an index. The index is not actually built until you generate your output. Your job is to insert the index keywords into the appropriate topics ("bringing the index keywords to the topics"). You simply need to make sure that you insert or copy the same index keyword into every topic that you want it to be associated with in the output.
Output (building)	In RoboHelp, you can generate output (single-source layouts) from the application only.	In Flare, you can build output (targets) from both the application and the command line.
Output types	You have access to the following output types, depending on your version of RoboHelp: ■ Cross - platform, browser - independent (WebHelp, WebHelp Pro, FlashHelp) ■ CHM (Microsoft HTML Help) ■ Print (Printed Documentation)	You have access to the following output types: ■ Tight integration with application (DotNet Help) ■ Cross-platform, browser-independent (WebHelp, WebHelp Plus) ■ Web-based desktop Help (WebHelp AIR) ■ CHM (HTML Help) ■ Print (Adobe PDF, XHTML, Microsoft XPS, Microsoft Word, Adobe FrameMaker)
Restarting Numbering in Lists	In RoboHelp, you open the Bullets and Numbering dialog and restart the numbering for a list item.	In Flare, you restart numbering for a list item from **Format>List>List Actions** (or by using the shortcut button in the Text Format toolbar). After entering the number, you click the blue "Enter" arrow next to your entry.
See Also Links	In RoboHelp, you "bring the topics to the See Also keyword." After you create the See Also keyword in the Index Designer, you drag the related topics to the Index Designer.	The process of creating "See Also" links is a similar situation to that of creating an index. In Flare, you "bring the See Also keyword (or "concept") to the topics" by inserting them as needed. You can then insert the See Also control link into the topic, similar to the way you would in RoboHelp.

Feature	RoboHelp	Flare
Templates	Depending on your version of RoboHelp, you might use different kinds of templates. You might use a topic template to apply the same content and look to many topics. You might use a project template to create a new project with some settings and content already in place for you. Finally, you might use a window template for creating secondary output windows with a specific size, orientation, look, etc.	In Flare, templates are used when you create a new project, as well as when you add new content (e.g., topics, snippets, style sheets) or new project-related elements (e.g., tables of contents, targets, skins) to it. A template is simply an existing file or element of the same type that you are creating. By basing your addition on a template, you are giving yourself a head start because the template already contains some content, settings, or formatting for you. If you want to use something like topic templates in Flare, you can create a master page and associate it with a target. Like a topic template in RoboHelp, a master page in Flare lets you create a header and footer for multiple topics. It also lets you automatically include breadcrumbs and a mini-TOC at the top and bottom of topics (which help users know where the topic is placed in relation to other topics in the table of contents). If you import a RoboHelp project that has topic templates containing headers and footers, your headers and footers are saved as snippets in Flare (stored in the Resources\Snippets subfolder in the Content Explorer).

Importing Tips

Following is information that you might find useful when importing your RoboHelp project to Flare. Each item below discusses how Flare treats a particular RoboHelp feature when you import a project to Flare.

- **Browse sequences** After you import a RoboHelp project to Flare, a browse sequence is automatically created for you and given the same name as your RoboHelp project. If you did not have a browse sequence in RoboHelp, the new Flare browse sequence is empty. If you did have a browse sequence in RoboHelp, each browse sequence (or browse sequence heading) from the Browse Sequence Editor is converted to a "book" in the Flare browse sequence (similar to a table of contents). The associated topics are converted to items under the book and link to the appropriate topics.
- **Conditional build tags** In Flare, these are known as "condition tags" and they are held in a condition tag set. So after you import a RoboHelp project, you will find your RoboHelp conditional build tags contained in a condition tag set called "Primary," located in the Conditional Text folder of the Project Organizer. You can open the condition tag set and create additional condition tags within it. You can also change the name of the condition tag set to whatever you want and add more sets if you like.
- **Converting topics to XHTML** If you do not convert your topics to XML (XHTML) during the import process, you will be asked to do so when you open each topic in Flare. Also, your index keywords will not be converted if the option to convert your topics to XML ("Convert all topics at once") is not selected during the import process.
- **Glossary** After you import a RoboHelp project, you will find your RoboHelp glossary terms and definitions contained in a glossary called "Primary," located in the Glossaries folder of the Project Organizer. You can open the glossary and create additional terms and definitions within it. You can also change the name of the glossary to whatever you want and add more glossaries if you like.
- **Images** Flare retains your file structure. Therefore, after you import your project, you are likely to see your image files at the root level of the Content Explorer in Flare. However, the default location for new images that you create in Flare is the Resources\Images subfolder. You can keep your imported files at the root level, or you can drag them to the Resources\Images subfolder if you want to keep them grouped with new files that you create. When you drag the imported files to a subfolder, a dialog will ask whether you want to update the links to the file that you are moving. Select **Update Links**.

- **Inline formatting and styles** If you select the "Convert inline formatting to CSS styles" option during the import process, Flare will create new styles based on the local formatting that it finds in your topics.

  EXAMPLE

  If you have portions of content where you have highlighted words and manually applied bold formatting to that text, you might find that Flare has converted those areas of formatting to a style called "span_1."

- **Merged CHM files** Let's say you are working with HTML Help and have merged your RoboHelp project with a CHM file from another project. The linked CHM file is placed in a special subfolder in the Content Explorer (Resources\CHMSupport) after you import the RoboHelp project to Flare. In order for this to work properly, the linked CHM file must first be placed in the root folder of your RoboHelp project. This process helps to keep the size of your Flare CHM file down. The linked CHM file is simply copied to your Output folder, next to the Flare CHM file. You need to make sure to include this secondary CHM file in the final destination of your output files where users will access them. For more information, see the online Help.

  Note: The CHMSupport subfolder is created automatically when this happens. However, you can create this subfolder yourself, as long as you make sure that it is spelled exactly right—with no spaces—and is located under the Resources folder in the Content Explorer. The CHMSupport subfolder will be used for any linked files that you do not want to be compiled into the Flare CHM file.

- **Publishing destinations** If you created publishing destinations in RoboHelp for WebHelp or FlashHelp layouts, those destinations are converted to destinations in Flare. They are located in the Destinations folder in the Project Organizer.

- **Shortcut links** Let's say you are working with HTML Help and have included Shortcut links in your RoboHelp project. After you import the RoboHelp project to Flare, the linked Shortcut file (e.g., DOC, XLS) is placed in a special subfolder in the Content Explorer (Resources\CHMSupport). In order for this to work properly, the linked Shortcut file must be placed in the root folder of your RoboHelp project. This process helps to keep the size of your Flare CHM file down and ensures that the shortcut links work correctly. The linked shortcut file is simply copied to your Output folder, next to the Flare CHM file. You need to make sure to include this linked Shortcut file in the final destination of your output files where users will access them. For more information, see the online Help.

- **Single source layouts** Flare will automatically convert any HTML Help, WebHelp, or Printed Documentation single source layouts that you had in RoboHelp. They can be found in the Targets folder in the Project Organizer. You can add more targets using those same output types. You can also create additional targets using Flare's other output types.

- **Styles and style sheets** After you import your project, you are likely to see your style sheet files at the root level of the Content Explorer in Flare, because Flare retains your file structure. However, the default location for new style sheets that you create in Flare is the Resources\Stylesheets subfolder. You can keep your imported style sheets at the root level, or you can drag them to the Resources\Stylesheets subfolder if you want to keep them grouped with new style sheets that you create. When you drag the imported files to a subfolder, a dialog will ask whether you want to update the links to the file that you are moving. Select **Update Links**. When you open the style sheet, it is displayed in Flare's Stylesheet Editor. Your paragraph styles can be found as classes under the <p> tag, and your generic character styles can be found as classes under the tag.

  In addition to the styles you created in RoboHelp, you will notice many additional styles that Flare adds. These additional styles are adopted from W3C specifications (see http://www.w3c.org), or they are MadCap-specific styles used to support unique features in Flare.

  For more information, see the online Help or the Flare *Styles Guide*.

You can use the following steps to locate your imported styles.

1. Make sure the Content Explorer is open.
2. Navigate to your style sheet.

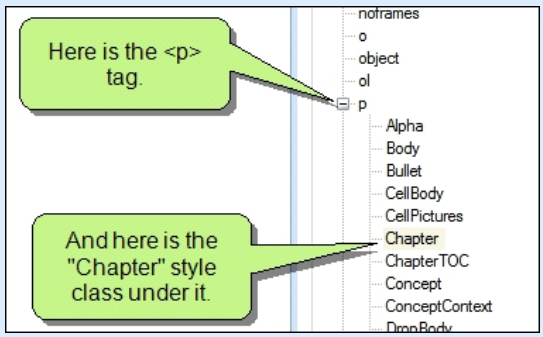
   Note: When you add new style sheets (not import them), the default location for those style sheets is within the Resources folder. To open a style sheet from this location, double-click the **Resources** folder. Then double-click the **Style-sheets** folder.

3. Double-click the style sheet. The Stylesheet Editor opens.
4. In the Stylesheet Editor, your imported paragraph and character styles can be found in the following locations.

   - **Paragraph styles** If you imported paragraph styles, they can be found under the <p> tag in the Stylesheet Editor.

     EXAMPLE

     If you imported a paragraph style called "Chapter," you can find that style under the <p> tag, and its new full name is "p.Chapter."

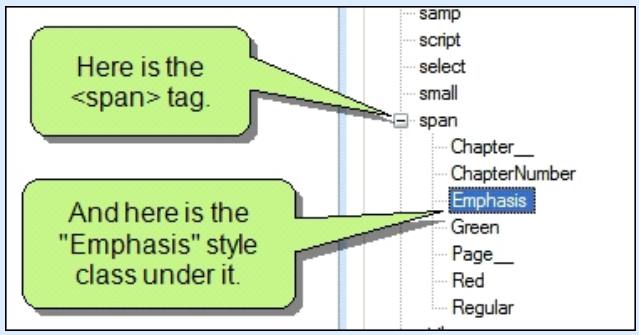
     

     Please note that this example is showing the Advanced view of the Stylesheet Editor. You can also work in the Simplified view of the editor, which shows the styles in a grid.

- **Character styles** If you imported generic character styles, your character styles can be found under the tag.

EXAMPLE

If you imported a generic character style called "Emphasis," you can find that style under the tag, and its new full name is "span.Emphasis."



- **Table of contents** After you import a RoboHelp project, you will find your RoboHelp table of contents (TOC) books and pages contained in a TOC called "Primary," located in the TOCs folder of the Project Organizer. You can open the TOC and create additional books and pages within it. You can also change the name of the TOC to whatever you want and add more TOCs if you like.
- **Template headers and footers** If you have headers or footers in a template in RoboHelp, Flare will save them as snippets in Flare. You can access these headers or footers by opening the Content Explorer and then opening the Resources\Snippets subfolder.
- **Windows** Your windows are automatically converted to "skins" and stored in the Skins folder of the Project Organizer.

Index

A

Accordions 6

B

Bars
- span 9
- structure 9
- table column 9
- table row 9
- tag 9

Browse sequences
- importing 25

Building
- output 22

Bulleted lists See *Lists*

Buttons
- Build Primary Target 11
- Create New Topic 11
- Insert Picture 11
- New TOC Book 11
- New TOC Item 11
- Open Item 11
- Preview Topic 11
- Redo 11
- Save 11
- toolbars 11
- Undo 11
- View Primary Target 11

C

Cascading style sheets 26

Characters
- styles 29

CHM See *HTML Help*

CHMSupport folder 26

Classes See *Styles*

Code See *Tags*

Compiling See *Building*

Condition tags
- importing 25

Content
- files 17
- RoboHelp versus Flare 17
- tags 19

Content Explorer 17

Converting
- topics to XHTML 25

CSS file 26

D

Default topic See *Startup topic*

Destinations
- importing 26

DOC file 26

DotNet Help 22

Drop-down text
- body 21
- heading 21
- hotspot 21
- RoboHelp versus Flare 21

F

Files
- content 17
- location 17
- output 18
- project 18
- RoboHelp versus Flare 18

FLPRJ file 18
Footers
  importing 29
Formatting
  importing 26

G

Generating See *Building*
Glossaries
  Glossary Editor 3
  importing 25
Graphics See *Pictures*

H

Headers
  importing 29
Heading
  drop-down text 21
Hotspot
  drop-down text 21
HTML Help 22
  merging 26

I

Images See *Pictures*
Importing
  glossaries 25
  headers and footers 29
  local formatting 26
  merged CHM files 26
  pictures 25
  publishing destinations 26
  Shortcut controls 26
  skins 29
  styles 27
  tables of contents 29
  targets 26
  tips 1, 25
Indexes
  RoboHelp versus Flare 22
Inline formatting See *Local formatting*
Interface See *Workspace*
Internal Text Editor 4

L

Links
  concept (See Also) 22
Lists
  restarting numbering 22
Local formatting 26
Local toolbars 8, 10

M

Master pages 23
Menus
  Build 10
  Project 10
  RoboHelp versus Flare 10
Merging
  HTML Help 26
Metadata 14
Microsoft HTML Help See *HTML Help*

N

Navigation links
  Shortcut controls 26
Numbered lists See *Lists*

O

Outline TOC See *Tables of contents*
Output
  building targets 22
  DotNet Help 22
  files 18
  FrameMaker 22
  HTML Help 22
  PDF 22
  printed output 22
  RoboHelp versus Flare 18

types 22
WebHelp 22
WebHelp AIR 22
WebHelp Plus 22
Word 22
XHTML 22
XPS 22

P

Paragraphs
styles 28
Pictures
importing 25
Printed output 22
Project toolbar 10
Projects
files 17-18
importing 25
Publishing
destinations 26

R

Redo
button 11
Resources folder 17, 25, 27, 29
Review toolbar 10
RoboHelp
importing 1, 25
RoboHelp versus Flare
building output 22
content 17
content editors 7
dialogs 10
drop-down text 21
editing language 21
feature differences 1, 21
indexing 22
interface differences 1, 5
location differences 1, 17
menu items 10
output files 18
output types 22
project elements 18
project files 18
replicating RoboHelp interface 1, 13
restarting numbered lists 22
See Also links 22
templates 23
terminology differences 1, 3
toolbars 10
topic files 18
video tutorial 1

S

Save
button 11
Screen captures See *Pictures*
Shortcut controls
importing 26
Shortcuts
toolbar buttons 11
Skins
importing 29
Snippets
footers 29
headers 29
Span bars 9
Standard toolbar 10
Startup topic 3
Structure bars 9
Styles
character 29
importing 27
paragraph 28
Symbols See *Characters*

T

Tables
columns 9
rows 9
Tables of contents
importing 29

Tabs
  editors 7
Tags
  location 19
  viewing 19
Targets
  building output 22
  importing 26
Templates 23
  RoboHelp versus Flare 23
Text Format toolbar 10
TOCs See *Tables of contents*
Toolbars
  buttons 11
  local 8, 10
  Project 10
  Review 10
  RoboHelp versus Flare 10
  Standard 10
  Text Format 10
Topics
  importing 25
  RoboHelp versus Flare 18
  XHTML 25
True code 14, 19
Tutorials
  Flare Compared with RoboHelp 1

U

Undo
  button 11
User interface See *Workspace*

V

Viewing
  tags (code) 19

W

WebHelp 22
WebHelp AIR 22
WebHelp Plus 22
Workspace
  accordions 6
  Build menu 10
  dialogs 10
  local toolbars 10
  menus 10
  Project menu 10
  Project toolbar 10
  Review toolbar 10
  Standard toolbar 10
  Text Format toolbar 10
  toolbars 10
  XML Editor 7

X

XHTML
  file 21, 25
XLS file 26