MadCap Software

Targets Guide

Flare 10

Copyright 2014 MadCap Software. All rights reserved.

Information in this document is subject to change without notice. The software described in this document is furnished under a license agreement or nondisclosure agreement. The software may be used or copied only in accordance with the terms of those agreements. No part of this publication may be reproduced, stored in a retrieval system, or transmitted in any form or any means electronic or mechanical, including photocopying and recording for any purpose other than the purchaser's personal use without the written permission of MadCap Software.

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

THIS GUIDE WAS CREATED USING MADCAP FLARE.

CONTENTS

CHAPTER 1 Introduction 7

Initial Targets Provided 8
Batch Targets 9
Exporting Projects Using Targets 10
Steps for Using Targets 11

CHAPTER 2 Determining the Output Type 13

Online Output Versus Print-based Output 14
Flare's Online Output Types 15
Flare's Print-based Output Types 40
Other Output Types 43

CHAPTER 3 Adding Targets 49

CHAPTER 4 Importing Targets 51

CHAPTER 5 Renaming Targets 53

CHAPTER 6 Opening Targets 55

CHAPTER 7 Setting a Primary Target 57

CHAPTER 8 Editing Target Settings 59

Applying Empty Alt Text 62
Associating a Browse Sequence with a Target 64
Associating Condition Tags with Targets 65
Associating an Alias File with a Target 84
Adding the DOCTYPE Declaration to Generated Topics 85
Specifying EPUB Options 87
Enabling Feedback in Flare 93
Associating Glossaries with Targets 94

Specifying Buttons for HTML Help Output 96
Displaying Merged Navigation in HTML Help Child Outputs 98
Patching Stylesheets and Image Links for HTML Help 100
Setting Up an HTML5 Target 101
Embedding Images in Microsoft Word Output 102
Generating Web-Safe Images 103
Disabling Auto-Sync of Import Files 104
Excluding Index Entries From Search 105
Selecting a Language 106
Adding Mark of the Web 108
Associating Master Pages with Targets 110
Disabling the Use of Master Pages 111
Adding Custom Meta Tags 112
Specifying the Output File and Folder in a Target 115
Replacing Characters and Spaces with Underscores 116
Using Custom File Extensions 117
Excluding Content not Linked Directly or Indirectly from Targets 118
Converting Output File Names and Folders to Lowercase 121
Omitting the Content Folder from Output 122
Changing the Output Type for a Target 123
Associating Master Page Layouts with Targets 124
Specifying PDF Options 125
Specifying PDF Output Via Word or FrameMaker 136
Collapsing Top Margins 138
Improving the Processing Performance of the Target 140
Specifying How Online Features are Converted in Printed Output 143
Associating Publishing Destinations with Targets 145
Integrating Flare and Pulse 146
Setting Redacted Text on Targets 174
Associating Relationship Tables with Targets 175
Associating a Search Filter Set with a Target 176
Enabling Partial-word Search 177
Generating Sitemaps for SEO 179
Associating Skins with Targets 180
Setting All Skins to be Generated 181
Getting the Latest Version 182
Specifying the Startup Topic in a Target 184
Applying Master Stylesheets 185

Associating a Medium with a Target 187
Associating a Master Table of Contents with a Target 188
Using the TOC Depth for Heading Levels in Print Output 189
Creating Headings for Unlinked Books in a Generated TOC 192
Removing Images from Headings in a Generated TOC 195
Overriding Variable Definitions in Targets 196
Specifying WebHelp AIR Options 198
Setting Up a WebHelp Plus Target 199
Enabling Mirror Margins in Word Output 201
Specifying XPS Output Via Microsoft Word 202
Ignoring Warnings When Building Output 204

CHAPTER 9 Building Output 213

Ways to Build Output 214
Build Progress Dialogs 215
Where Output Files are Stored 216
Building the Primary Target 217
Building a Single Target 219
Building Output Using a Batch Target 220

CHAPTER 10 Publishing Output to Destinations 229

CHAPTER 11 Viewing Output 233

APPENDIX A PDF Guides 243

APPENDIX B Output Type Comparison Tables 245

General 246
Accessibility 248
Context-Sensitive Help 249
Feedback Statistics and Reporting Features 250
Generated Content 251
Glossaries 253
Horizontal Rules 254
Images 255
Indexes 256
Language Support 257
Master Pages and Page Layouts 259
Merging Output 260

Miscellaneous Features 261
Multimedia 263
Navigation Links 264
Pulse (Socially Enabled Output) 265
Search 266
Skins 273
Tables of Contents and Mini-TOCs 278
Target Settings 280
Variables 283

CHAPTER 1

Introduction

There are several types of online output (DotNet Help, Eclipse Help, Microsoft HTML Help, HTML5, WebHelp, WebHelp AIR, WebHelp Mobile, WebHelp Plus) and several types of print-based output (Adobe FrameMaker, Adobe PDF, Microsoft Word, Microsoft XPS, XHTML) that you can produce in Flare. In addition, you can create EPUB output, which is used for ebooks. There is also an output type that lets you generate DITA code from your project (DITA code can be used to create either online or print-based output). Each output type has its own set of advantages.

It is easy to confuse output types with targets, but they are two different concepts. A target is one instance of an output type. When you build your final output, you are essentially building one or more of the targets in your project.

This chapter discusses the following:

Initial Targets Provided 8
Batch Targets 9
Exporting Projects Using Targets 10
Steps for Using Targets 11

Initial Targets Provided

When you create a new project using one of Flare's factory templates, one or more new targets are added to your project.

However, just because you are provided with one or more targets initially, this does not mean that you are limited to just that target in your project. In Flare you can add targets using any of the available formats, and you can make as many copies of an existing target that you want. For example, your project might end up containing three targets that are all based on the output type (in addition to other targets).

A target file has an .fltar extension and is stored in the Project Organizer under the Targets folder.

Batch Targets

In addition to regular targets, you can use batch targets to generate and/or publish one or multiple targets in a batch from the user interface, perhaps scheduled to run at a specific time. See "Building Output Using a Batch Target" on page 220.

Exporting Projects Using Targets

You can export project files and content based on targets that you create. One reason you might want to use this feature is to quickly and easily archive projects, especially if you have an extremely large Flare project and need to archive only parts of it. Another use for this feature is translation. If you only need a portion of a master project to be translated, you don't want to send the translator all of the files, but rather a smaller version of the project containing only the files requiring translation. For more information see the Flare online Help.

Steps for Using Targets

Following are the basic steps involved with targets:

1. **Determine Output Type** The first task in developing output for your project is to determine which type of output is most appropriate for your needs. You might even need to produce multiple outputs and require more than one output type. See "Determining the Output Type" on page 13.
2. **Add Target** You can add as many new targets as you want to a project. You can also copy existing targets. See "Adding Targets" on page 49.
3. **Rename Target** Let's say that you have compared the different output types available and decided that Microsoft HTML Help is the output type that best meets your needs. However, you want to produce two online Help systems from your project. The solution is to add a new Microsoft HTML Help target or make a copy of another Microsoft HTML Help target so that you now have two targets based on the Microsoft HTML Help output type. You can then rename the targets to help identify them. See "Renaming Targets" on page 53.
4. **Open Target** You can open a target at any time to edit it. See "Opening Targets" on page 55.
5. **Set Primary Target** You can have as many targets in a project as you want, and any of your targets can be built (generated) whenever you like. However, chances are that you will have one target that you work with more than the others. Therefore, you can set it as your primary target. See "Setting a Primary Target" on page 57.
6. **Edit Target Settings** Using the Target Editor, you can edit properties for any of your targets. See "Editing Target Settings" on page 59.

CHAPTER 2

Determining the Output Type

One of the first tasks in developing output for your project is to determine which type of output is most appropriate for your needs. You might even need to produce multiple outputs and require more than one output type.

This chapter discusses the following:

Online Output Versus Print-based Output 14
Flare's Online Output Types 15
Flare's Print-based Output Types 40
Other Output Types 43

Online Output Versus Print-based Output

There is a fine line between what is called "online output" and what is called "print-based" output. The truth is that topics in virtually any of Flare's online output types can be sent to a printer, and therefore considered print-based. Similarly, any of the print-based output types can be viewed electronically, and therefore considered online. The real distinction between online and print-based outputs has to do with their primary purpose. Online outputs are usually intended to be viewed on a screen, rather than on a printed page. The idea is to show only small pieces of content at a time and allow users to jump around to other topics or elements of the output. On the other hand, print-based output follows a more traditional format that you would find in an actual book or manual—with the pieces of the output following one after the other on pages until the end of the book (e.g., title page, table of contents, preface, chapters, index, appendixes—with page numbers, as well as header or footer content, shown along the way). Then there is EPUB output, which is intended to be viewed on a screen, but follows a structure closer to print-based outputs.

Flare's Online Output Types

Following are overviews of each of Flare's online output types.

DotNet Help

DotNet Help is a Help output format developed by MadCap Software for Windows desktop applications. It was designed to include the best attributes of Microsoft HTML Help and WebHelp, while filling the holes left behind by those formats. DotNet Help is designed specifically to support Visual Studio developers. It includes a freely redistributable viewer (MadCap Help Viewer), as well as components for the Visual Studio developer. These components can be dropped into your Flare project to facilitate context-sensitive Help, embedded Help, and features such as automated search string communication between the application and the DotNet Help documentation.

The output consists of a collection of files that you will distribute to users with the freely downloadable MadCap Help Viewer. The main entry file has an .mchelp extension.

The online Help attached to the Flare application was created with DotNet Help.

KEY FEATURES

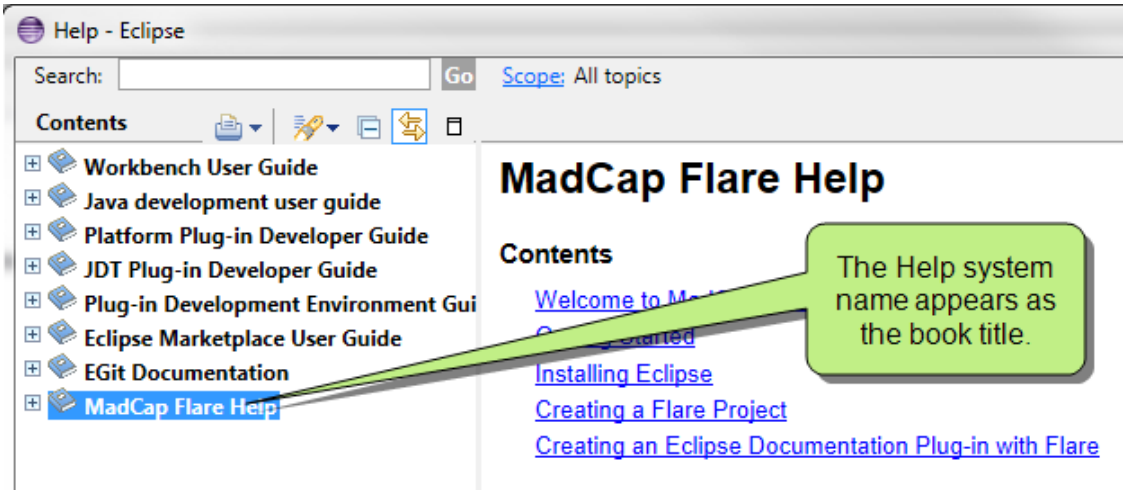
Following are some of the key features of the DotNet Help format:

- **.NET Application Support**
- **Accessible** This is a good choice if you do not want users to be burdened by the security warnings and limitations that are often encountered with Microsoft HTML Help and WebHelp.
- **Customizable/Modern Interface** You can produce a customizable interface that is much more modern looking than the aging Microsoft HTML Help. It will blend easily into a modern environment.
- **Embedded Context-Sensitive Help (CSH)** You can create embedded CSH in a .NET application. This includes the ability to produce Dynamic Help.
- **Multiple Languages** Users can select between multiple languages for the interface when viewing your output. This is possible because the freely distributable MadCap Help Viewer lets users select English, French, German, or Japanese for the interface.
- **Boolean Operator and Asterisk (*) Wildcard Character Searches** These are supported on Microsoft Windows XP (and newer operating systems) when SQL Compact is installed on the local computer building the Help.

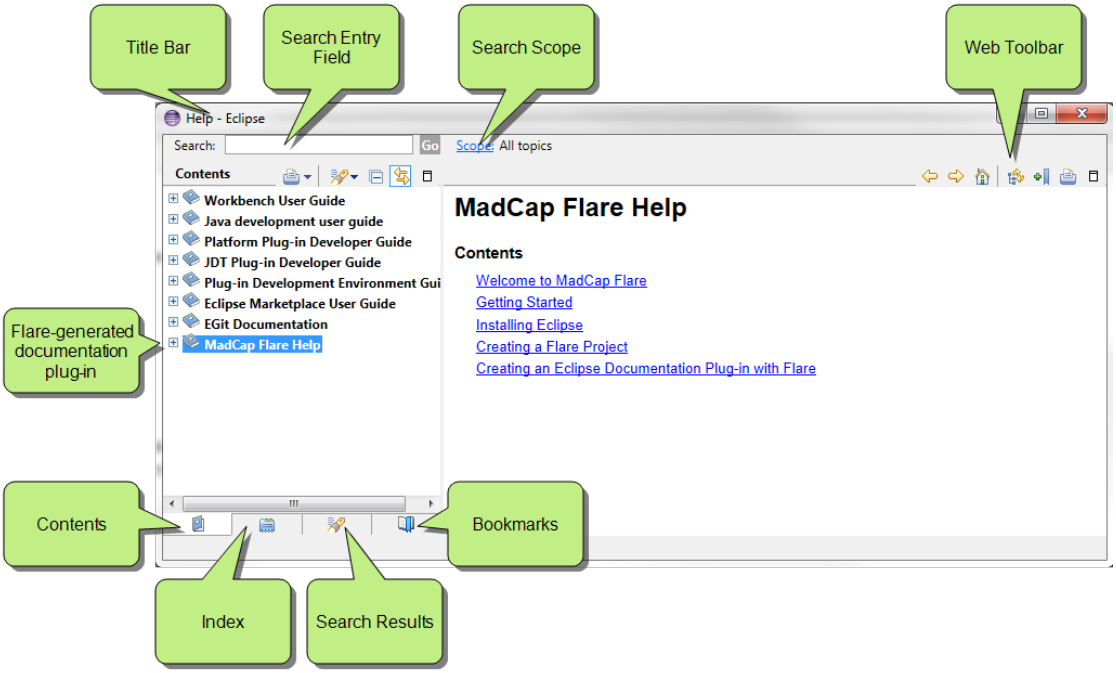
Eclipse Help

ECLIPSE HELP

The Eclipse Help output lets you generate Eclipse Help plug-ins using your Flare content as its source material. It displays your output content in the Eclipse Help Viewer.



The Eclipse Help Viewer uses an embedded Jetty server to provide your users with navigation features similar to other online viewers. Jetty is a Java-based web server implementation that tightly integrates with Eclipse and other open source projects.



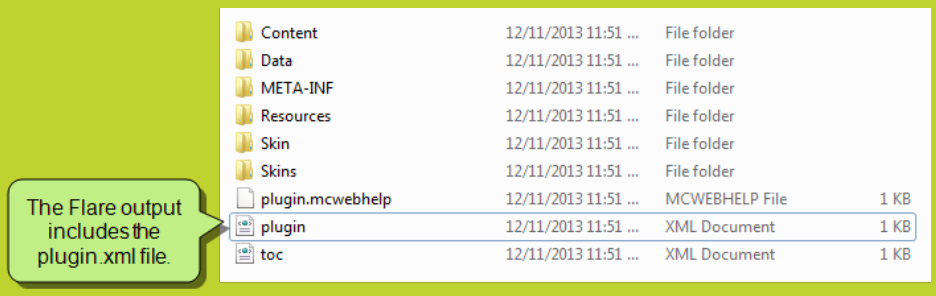
The viewer's features and user interface must be customized in Eclipse. To learn about your customization options, visit this link: **http://help.eclipse.org/kepler/index.jsp**.

The Eclipse Help output type (called a "plug-in" in Eclipse) consists of the following components:

» **Plug-In XML File** This is an XML-based main file that defines the name and ID of the Eclipse help plug-in that you generate with Flare.

E X A M P L E

The output generates a plugin.xml file for Eclipse:



The Flare output includes the plugin.xml file.

The plugin.xml file contains the extension point definitions that the Eclipse runtime environment needs to activate your help plug-in:

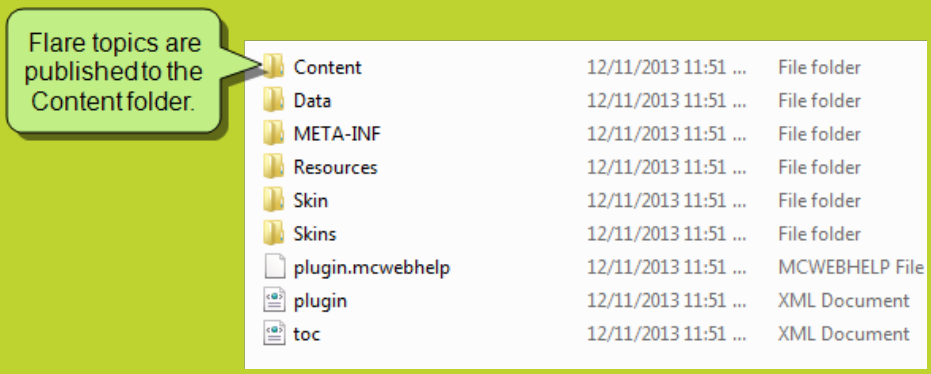
```
  <?xml version="1.0" encoding="UTF-8"?>
  <?eclipse version="3.0"?>
- <plugin provider-name="EclipseHelp_01" name="EclipseHelp_01" ve
  - <extension point="org.eclipse.help.contexts">
        <contexts file="csh.xml"/>
    </extension>
  - <extension point="org.eclipse.help.index">
        <index file="index.xml"/>
    </extension>
  - <extension point="org.eclipse.help.toc">
        <toc file="toc.xml" primary="true"/>
    </extension>
    <extension point="org.eclipse.help.toc"/>
  </plugin>
```

Eclipse extension point definitions.

» **HTML Content Files** As with other output types, Flare publishes a separate HTML file for each documentation topic. These are contained in the Content folder.

E X A M P L E

As with other output types, the Content folder contains your Flare target's topics:

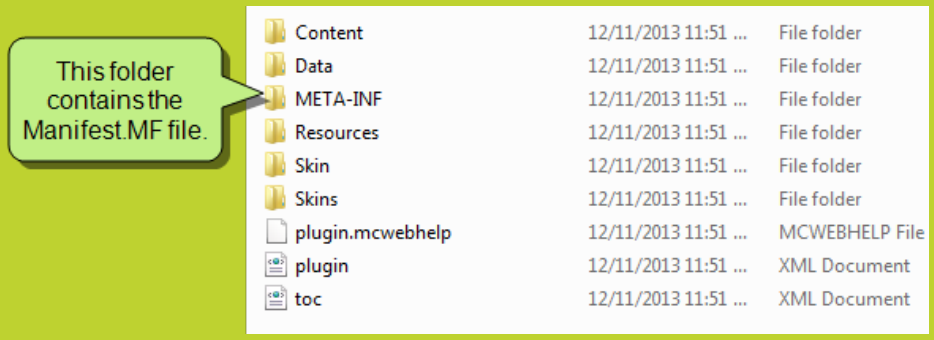


» **Other XML Files** Flare publishes separate XML files for the table of contents, keyword index, and context-sensitive help. These correspond to the **org.eclipse.help.toc**, **org.eclipse.help.index**, and **org.eclipse.help.contexts** extension points.

» **Manifest.MF File** The manifest file is used only when you package your Eclipse help plug-in as a Java archive (JAR) file. This file is required by Java when creating a JAR file.

E X A M P L E

If you package your Eclipse help plug-in as a JAR file, the META-INF folder contains the required Manifest.MF file.



KEY FEATURES

Following are some of the key features of the Eclipse Help format:

- **Eclipse Help Plug-In and Extension Points** This output type is particularly useful when you want to create dynamic, searchable content that integrates into the Eclipse workbench as a platform plug-in. Flare generates an XML-based plug-in file (plugin.xml) that specifies the context Help (csh.xml), index (index.xml), and table of contents (toc.xml) files that Eclipse needs to load your Flare content in the Eclipse Help system.
- **Eclipse Help Viewer and Dynamic Help Window** You can generate the Eclipse Help output for viewing with the Eclipse Help Viewer and the Eclipse Dynamic Help window, which displays context-sensitive Help and search results.
- **Distribute Server-based or Local Content** Your Eclipse content can be distributed over the web via an Eclipse InfoCenter server or on local systems with a pared down version of Eclipse, which provides the search capabilities and skin for the Eclipse Help Viewer.
- **Publish Content as a Collection of Files or Single JAR File** This format lets you publish the output for the Eclipse Help plug-in as a series of files in a folder or bundled in a single Java archive (JAR) file.

Note: Eclipse provides its own breadcrumb and topic toolbar support. See the Platform Plug-in Developer Guide from Eclipse at: **http://help.eclipse.org/kepler/index.jsp**.

Note: If your Eclipse Help target contains non-web safe images (e.g., WMF, EMF, BMP, TIF, TIFF, XPS, EXPS), be sure to select **Generate "web-safe"images** in the Advanced tab of the Target Editor.

Note: The Eclipse Viewer cannot display PDFs opened from links such as those in TOCs and browse sequences.

Note: When you create a TOC or browse sequence intended for Eclipse output, you should make sure all items are linked. Unlinked items may not display in the output.

Note: Eclipse does not use the Flare skin, so there will be no glossary pane in the output. However, you can insert a glossary proxy (which is normally used for print-based outputs) as an alternative.

ADDITIONAL INSTALLATIONS

To take advantage of this output type, you must perform these additional installations:

JAVA RUNTIME ENVIRONMENT (JRE)

As the individual compiling the output, you need to install the required Java Runtime Environment (JRE), which is part of the "Eclipse IDE for Java EE Developers" download. For more information see the Flare online Help.

JAVA

As the individual compiling the output, you must download and install the Java Version 7 (Update 45) or later on your Windows operating system. For more information see the Flare online Help.

RECOMMENDED READING

If you are new to the Eclipse platform, be sure to visit the links below to learn more about it:

- **Eclipse Foundation** http://www.eclipse.org/
- **Eclipse Documentation** http://help.eclipse.org/kepler/index.jsp
- **Eclipse Doc Style Guide** http://wiki.eclipse.org/Eclipse_Doc_Style_Guide

Microsoft HTML Help

Microsoft HTML Help is an HTML-based Help format that runs on Windows 32-bit platforms and requires Internet Explorer on the end users' systems. You can use Microsoft HTML Help to create Help for Windows desktop applications.

The Microsoft HTML Help output consists of a single CHM file that you will distribute to users.

KEY FEATURES

Following are some of the key features of the Microsoft HTML Help format:

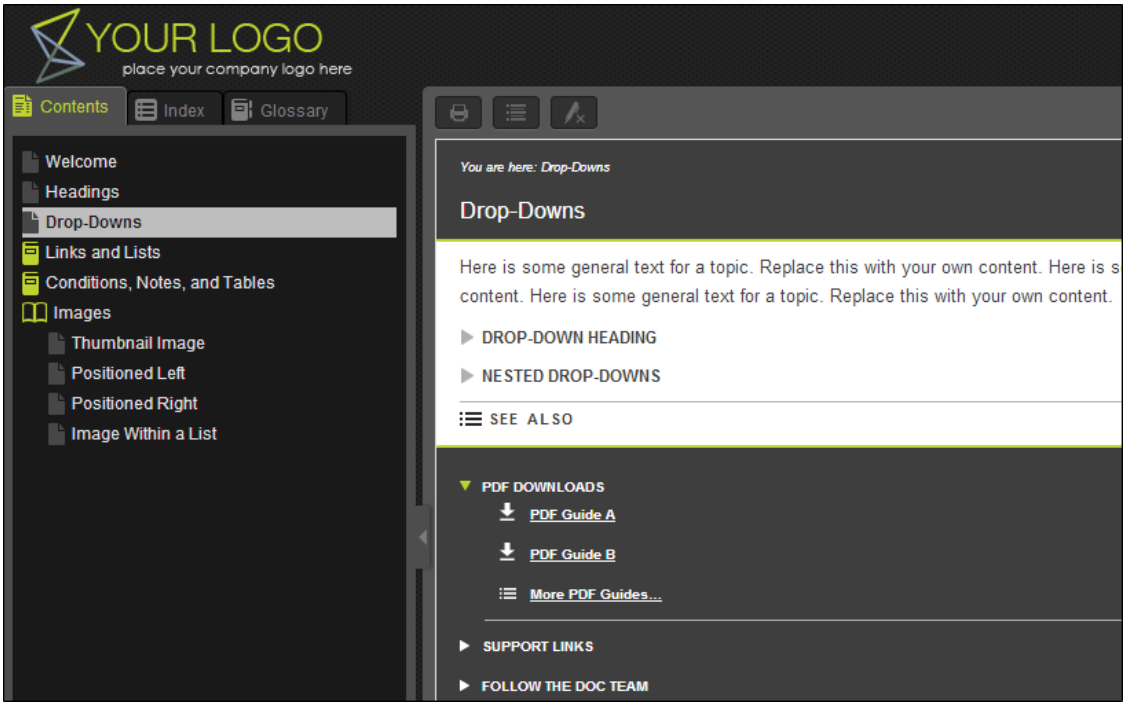
- **32-bit Windows Application Support**
- **Internet Explorer Support**
- **Local** This format is a good choice if you want users to store and open the output locally, rather than from a server, such as a website.
- **Single File** This format is a good choice if you want to create output that has just one file.

Note: Your users need Internet Explorer (4.0 or later) installed and a 32-bit Windows operating system (Windows 95 or later).

HTML5

HTML5 output is an online format that is the next generation of WebHelp. It is similar to the regular WebHelp format; however, whereas the original WebHelp format works with HTML4, this output format supports the HTML5 specification developed by the Web Hypertext Application Technology Working Group (WHATWG—**http://whatwg.org**) and the World Wide Web Consortium (W3C—**http://w3.org**). Therefore, the HTML5 format results in better markup and offers additional features not found in the other WebHelp outputs.

The HTML5 output consists of a collection of files that you will distribute to users. The output will be displayed in the user's Internet browser window. The main entry file has an .htm extension.



The online Help on the madcapsoftware.com website was created with HTML5.

KEY FEATURES

Following are some of the key features of the HTML5 format:

- **WebHelp Features** This format lets you take advantage of features and benefits available with the regular WebHelp output type.
- **Frameless Output** Pages in HTML5 output do not use <frameset> or <frame> tags, like they do in regular WebHelp output. This means that, behind the scenes, HTML5 output can be said to be "frameless," even though the user still sees distinct areas that display content, navigation, and toolbars. By being frameless, HTML5 output is much easier for web crawlers to find, thus improving search engine results.
- **Responsive Output** Responsive output is essentially "smart" output. When you generate HTML5 targets using a skin enabled with responsive output, the output display changes depending on the size of your browser or device you are using. So instead of seeing the entire HTML5 output on a tablet or smart phone, you will see a condensed look that is more appropriate for that device. You get the same effect if you shrink your browser to a smaller size.
- **Specific Page Addresses** With the regular WebHelp output, you can only give readers a single URL path, which opens the starting page for the output. With HTML5 output, you can point end users to a specific page in the output by giving them the exact URL path.

E X A M P L E

Let's say you generate regular WebHelp output, with the primary output file called "MyOutput.htm." When you view the output with the initial topic shown (say it's called "Welcome.htm"), the URL path in the browser might look something like this.

http://www.mycompany.com/documentation/MyOutput.htm

And if you then click on another topic (say, "WhatsNew.htm"), the same URL path will be shown. It does not change, even though you opened a different page.

On the other hand, suppose you generate HTML5 output. The URL in the browser may look the same as the path for regular WebHelp when the output is first displayed.

http://www.mycompany.com/documentation/MyOutput.htm

But if you click on the "WhatsNew.htm" page, the URL will change to this.

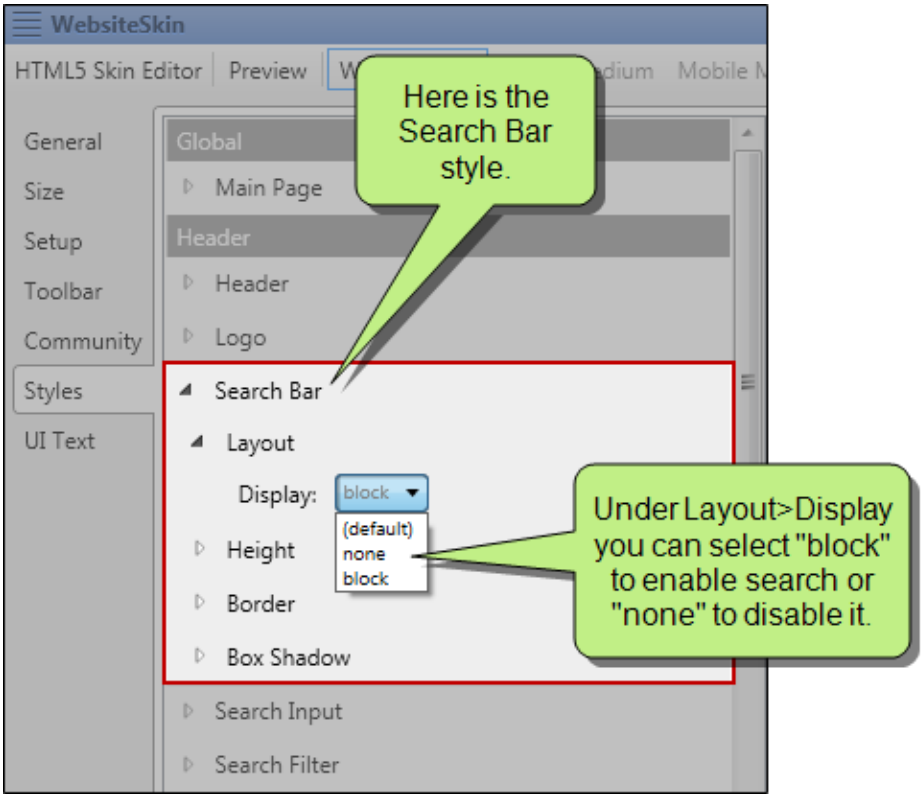
http://www.mycompany.com/documentation/MyOutput.htm#WhatsNew.htm

And if you click on the "Welcome.htm" page to return to it, the URL will change to this.

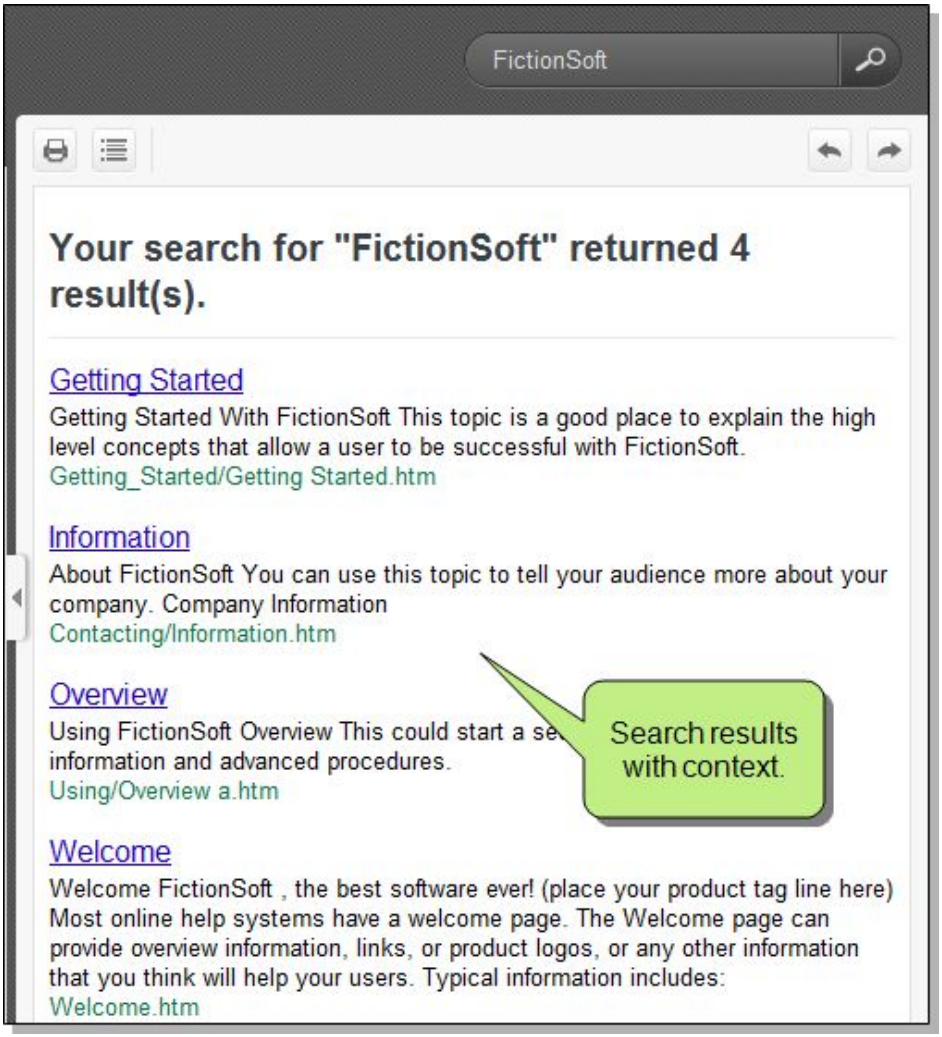
http://www.mycompany.com/documentation/MyOutput.htm#Welcome.htm

- **Sitemap/Search Engine Optimization (SEO)** You can generate a sitemap when compiling your output. This helps with search engine optimization (SEO), making it easier for search indexing services (i.e., spiders, crawlers, or bots) to find your output. Therefore, the entire output is indexed and search engine results are improved.
- **Modern Looking** When you build HTML5 output, you will notice that it has a more modern look, compared with the older WebHelp formats.
- **Search Bar** In the other online outputs, the search feature is displayed in an accordion or tab. In HTML5 output, the search feature is included in a search bar above the topics.

In addition, rather than enabling or disabling the search feature on the General tab of the Skin Editor, you do so on the Styles tab by editing the Search Bar property.

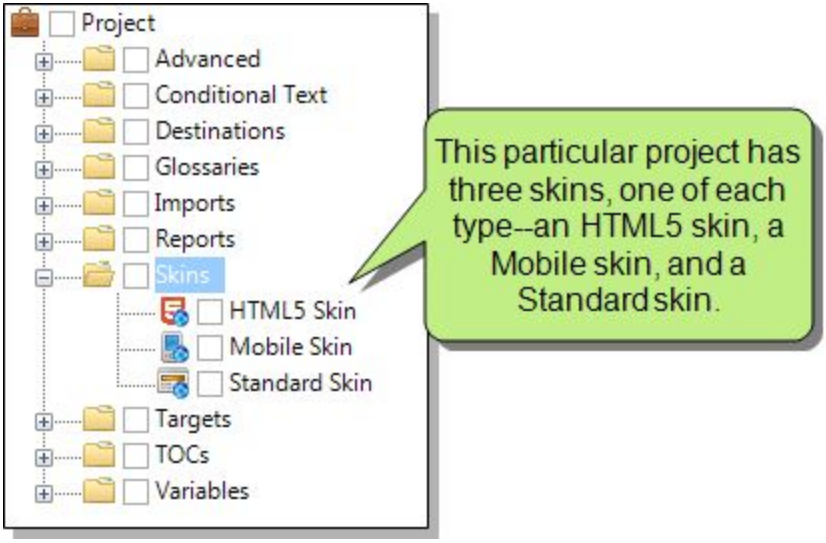


» **Search with Context** If end users perform a search in your output, they will see context next to each result, rather than terms only.



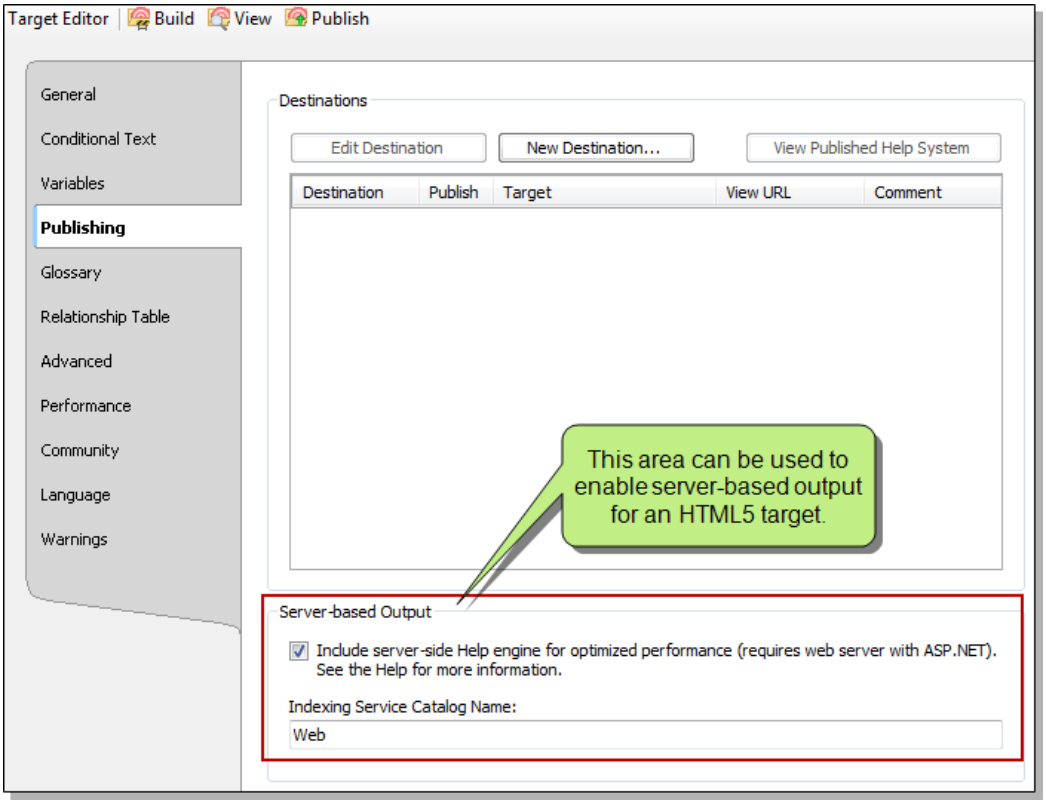
In addition, you can create topic meta descriptions, which will be used in the search results abstract. Also, the meta description itself is searchable.

- **Smoother Dynamic Effects** Features such as drop-down text, expanding text, and popups have a smoother appearance when they are activated.
- **Unique Skin** Because HTML5 output is quite different from the other online formats, it requires its own skin.



Note: Unlike other output types, separate language skins are not used for HTML5 targets. Instead, a UI Text tab is located in the Skin Editor, which lets you perform all of the localization tasks that you would normally accomplish in a language skin.

» **Server-based Output** You can create HTML5 output in its regular state, or you can select an option in the Publishing tab of the Target Editor to enable server-based functionality. This allows you to accomplish the same results as WebHelp Plus output—searching of non-XHTML content, faster server-side search, and automatic runtime merging.



Following are benefits of server-based HTML5 output;

» **Multiple Platforms** You can publish to a machine running Windows XP, Windows Server 2003, Windows Server 2008, Windows 7, or Windows Vista.

» **Searching of Non-XHTML Content** When end users perform a search in your online output, you can ensure that non-XHTML files (e.g. PDF, DOC, XLS) are included in that search. When you build WebHelp

Plus or HTML5 server-based output, a subfolder named "AutoSearch" is created and placed in the generated output folder. You can place non-XHTML files within the published AutoSearch subfolder (whether the non-XHTML files are linked to content from your Flare project or not). When users perform a search in your output, those non-XHTML files will also be scanned and become accessible to the users.

- **Faster Searching** Another benefit of generating and publishing WebHelp Plus or HTML5 server-based output to a web server running Microsoft IIS is that users will find the task of performing a search to be much faster than it is otherwise. This is especially useful if you have a very large Help system.
- **Automatic Runtime Merging of Flare Projects** This is an easy way to merge the output from multiple WebHelp Plus or HTML5 server-based Flare targets into one Help system. These targets can be originated from the same Flare project or from different Flare projects. You simply place the output files in the correct location on the server (i.e., within your master project's AutoMerge folder). Flare then automatically merges the output from all of the targets when users access the Help. From the end user's perspective, the results are seamless, appearing as one large Help system. All of the TOCs, browse sequences, indexes, glossaries, and search capabilities for the projects are merged.

WebHelp

WebHelp is a Web-based Help format that can run on almost any browser or platform. You can use WebHelp to create online documentation for the Internet or an intranet, as well as for desktop applications.

The WebHelp output consists of a collection of files that you will distribute to users. The output will be displayed in the user's Internet browser window. The main entry file has an .htm extension.

KEY FEATURES

Following are some of the key features of the WebHelp format:

- **Multiple Browsers** This format is a good choice if your users have different Internet browsers on their systems.
- **Multiple Languages** You can produce an output interface in various languages. This is possible through the use of language skins.
- **Multiple Platforms** This format is a good choice if your users are working on different platforms (operating systems).
- **Web-Based** This format is a good choice if you are writing online documentation for distribution on the Internet or an intranet.
- **Sitemap/Search Engine Optimization (SEO)** You can generate a sitemap when compiling your output. This helps with search engine optimization (SEO), making it easier for search indexing services (i.e., spiders, crawlers, or bots) to find your output. Therefore, the entire output is indexed and search engine results are improved.

Note: The following browsers are supported for WebHelp: Internet Explorer 7.0 or later, Firefox 3.5 or later, Safari 4.0 or later, Google Chrome, and other browsers that support scripts and framesets.

WebHelp AIR

WebHelp AIR is a Web-based Help format that is identical to the regular WebHelp output in most ways. However, WebHelp AIR uses direct integration with Adobe AIR, which is designed to bring Web-related content to a desktop environment by taking web files and incorporating them into a single file to be opened locally, rather than from a server.

The WebHelp AIR output that you generate consists of a single file with an .air extension, which you distribute to users. When users access this file, they are taken through an installation process, and as a result an executable file with an .exe extension is created on their local drive and saved in their C:\Program Files directory. In other words, the output becomes its own application. The output will be displayed in the application window that is part of the AIR installation.

KEY FEATURES

Following are some of the key features of the WebHelp AIR format:

- **WebHelp Features** This format lets you take advantage of features and benefits available with the regular WebHelp output type.
- **Local** This format is a good choice if you want users to store and open the output locally, rather than from a server, such as a website.
- **Single File** This format is a good choice if you want to create output that has just one file.
- **Windows, Mac, and Linux** This format is a good choice if you want users to be able to install the file not only in Windows, but in a Macintosh and Linux environment as well.

ADDITIONAL INSTALLATIONS

In order for you and your end users to take advantage of this output, both you and they must perform additional installations.

JAVA RUNTIME ENVIRONMENT INSTALLATION (YOU)

As the individual compiling the output, you need to install the Java Runtime Environment (JRE) before generating output.

You can download the JRE from: http://java.sun.com/javase/downloads/index.jsp

ADOBE AIR INSTALLATION (YOU AND END USERS)

Anyone who wants to view the generated WebHelp AIR output needs to install Adobe AIR first. This means that both you and your end users must run this installation.

You can download Adobe AIR from: http://get.adobe.com/air/

WebHelp Mobile

WebHelp Mobile is an output type that lets you deploy Web-based, XHTML output to mobile devices. WebHelp Mobile maintains an easy and intuitive interface that fits on a very small screen. The Home page in WebHelp Mobile output contains navigation links to access the various panes that you can include: TOC, Index, Glossary, Search, Browse Sequences.

The WebHelp Mobile output consists of a collection of files that you distribute to users by placing them on a web server. End users then use their mobile device to open those files, just as they would open any website. The output will be displayed in the user's mobile browser. The main entry file has an .htm extension.

KEY FEATURES

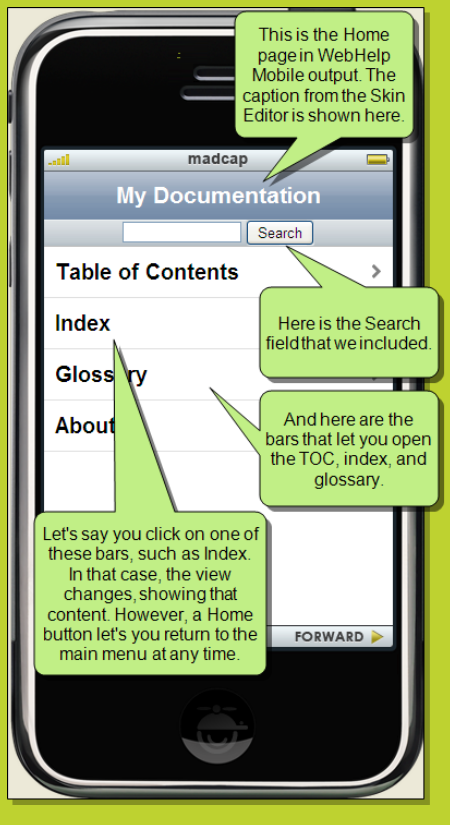
Following are some of the key features of the WebHelp Mobile format.

- **WebHelp Features** This format lets you take advantage of features and benefits available with the regular WebHelp output type.
- **Mobile** This format is a good choice if your end users are on the move and need to be able to access your documentation on their mobile devices.
- **Multiple Languages** You can produce an output interface in various languages. This is possible through the use of language skins.
- **Multiple Platforms** This format is a good choice if your users are working on different mobile platforms. Some of the major platforms supported are iPhone, iPad, Microsoft Mobile, LG WebOS (Palm WebOS), Blackberry, and Android.
- **Sitemap/Search Engine Optimization (SEO)** You can generate a sitemap when compiling your output. This helps with search engine optimization (SEO), making it easier for search indexing services (i.e., spiders, crawlers, or bots) to find your output. Therefore, the entire output is indexed and search engine results are improved.

Note: Some older mobile browsers do not support certain features (e.g., DHTML, search), whereas newer mobile browsers do support them. Other features, such as "mouse over," are not supported in either older or newer mobile browsers.

EXAMPLE

Let's say you want to create WebHelp Mobile output so that end users can see your documentation on their cell phones. In the skin associated with the target, you specify that the output should include a TOC, an index, a glossary, and search. After generating the output, the home page might look something like this:



madcap
Home
Index
A
Address
C
Company
Concepts
Contact
Address
Email
Phone
BACK
FORWARD
Here is the Home button.
Index entries let you quickly jump to associated topics.

madcap
Home
Getting Started With FictionSoft
This topic is a good place to explain the high level concepts that allow a user to be successful with FictionSoft.
Overall work flow for FictionSoft
Important concepts necessary to be successful
Special language or jargon that a user will need to understand
Any other information that a user would need
Tip: If you want to see the underlying structure of the document you are working on, then select the Show Tags icon on the tool bar.
Copyright © 2009 MegaSoft Fictional Software.
BACK
FORWARD
Here is what a topic might look like in WebHelp Mobile output.

WebHelp Plus

WebHelp Plus is a Web-based Help format that is identical to the regular WebHelp output in most ways. However, WebHelp Plus is designed to work on a Web server running Windows 8, Windows 7, Windows Server 2012, Windows Server 2008 R2, Windows Server 2008, Windows Server 2003, Windows XP, or Windows Vista. It also uses Microsoft Internet Information Services (IIS) and ASP.NET. To provide faster search, WebHelp Plus uses Microsoft Indexing Service or Windows Search. The benefit of publishing WebHelp Plus output is that you and your users can take advantage of some advanced features, including searching of non-XHTML content, faster server-side search, and automatic runtime merging.

The output consists of a collection of files that you will distribute to users by publishing output to a Microsoft IIS web server. The output will be displayed in the user's Internet browser window. The main entry file has an .htm extension.

KEY FEATURES

Following are some of the key features of the WebHelp Plus format.

- **WebHelp Features** This format lets you take advantage of features and benefits available with the regular WebHelp output type.
- **Sitemap/Search Engine Optimization (SEO)** You can generate a sitemap when compiling your output. This helps with search engine optimization (SEO), making it easier for search indexing services (i.e., spiders, crawlers, or bots) to find your output. Therefore, the entire output is indexed and search engine results are improved.
- **Multiple Platforms** This format is a good choice if you publish your output to a machine running Windows XP, Windows Server 2003, Windows Server 2008, Windows 7, or Windows Vista.
- **Searching of Non-XHTML Content** When end users perform a search in your online output, you can ensure that non-XHTML files (e.g. PDF, DOC, XLS) are included in that search.When you build WebHelp Plus or HTML5 server-based output, a subfolder named "AutoSearch" is created and placed in the generated output folder. You can place non-XHTML files within the published AutoSearch subfolder (whether the non-XHTML files are linked to content from your Flare project or not). When users perform a search in your output, those non-XHTML files will also be scanned and become accessible to the users.
- **Faster Searching** Another benefit of generating and publishing WebHelp Plus or HTML5 server-based output to a web server running Microsoft IIS is that users will find the task of performing a search to be much faster than it is otherwise. This is especially useful if you have a very large Help system.
- **Automatic Runtime Merging of Flare Projects** This is an easy way to merge the output from multiple WebHelp Plus or HTML5 server-based Flare targets into one Help system. These targets can be originated from the same Flare project or from different Flare projects. You simply place the output files in the correct location on the server (i.e., within your master project's AutoMerge folder). Flare then automatically merges the output from all of the targets when users access the Help. From the end user's perspective, the results are seamless, appearing as one large Help system. All of the TOCs, browse sequences, indexes, glossaries, and search capabilities for the projects are merged.

Note: If you want to test WebHelp Plus on your local computer,the advanced search features of WebHelp Plus are not operable.

Note: The following browsers are supported for WebHelp Plus: Internet Explorer 7.0 or later, Firefox 3.5 or later, Safari 4.0 or later, Google Chrome, and other browsers that support scripts and framesets.

Flare's Print-based Output Types

Following are overviews of each of Flare's print-based output types.

PDF

Short for "Portable Document Format," PDF is an open standard format for electronic documentation exchange invented by Adobe. PDF files are used to represent a two-dimensional document in an device- and resolution-independent fixed-layout format. You can generate PDF output from your project directly, or you can generate a PDF while simultaneously building FrameMaker or Word output.

You can also set PDF options in the Target Editor. These options let you specify the way that images, document properties, the initial view, and security are handled in the output.

PDF output consists of a file with a **.pdf** extension that you can print or distribute to users.

XHTML

XHTML is a browser-based output type that consolidates project content into an XML file. It can be viewed online or printed.

XHTML is a good choice if you need an "intermediary" format for your large, custom, enterprise level proprietary systems. You can easily transform Flare-authored content into your own system. By creating the single file output, you can feed it into your own parser/transform and convert all of the Flare content to your internal formats. This is part of the flexibility that allows Flare to be integrated into just about any tool chain or work flow. If you do not have a situation like that, you may find one of the other formats to be a more suitable option for you.

XHTML consists of a collection of XHTML files that you can print or distribute to users. This includes the following.

- A file with an **.htm** extension. This is the XHTML file that contains the consolidated topic content from your project.
- A file with an **.mcbook** extension. This file is used to display the chapters in the MadCap Book Viewer.
- A Resources folder with **various ancillary files**, such as stylesheets and images.

If you want to make XHTML output accessible for others, you need to include all of the files in the output mentioned in this list. Otherwise, when they view the output, certain elements (e.g., images) might be missing from the pages.

XPS

Microsoft's XML Paper Specification (XPS) is a document format with a markup language that is a subset of XAML for Windows Presentation Foundation. XPS is an alternative to Adobe's Portable Document Format (PDF). You can generate XPS output from your project directly. Make sure you have the latest version of Microsoft .NET Framework installed on your computer. This is a free download from **microsoft.com**. Alternatively, you can generate XPS output while simultaneously building Word 2007 output (by installing a free add-in download from Microsoft).

XPS output consists of a collection of XPS files that you can print or distribute to users. This includes:

- A file with an **.xps** extension. This is the file that contains the consolidated topic content from your project. This is the main file and the only one that is essential. This is the file that you would provide to a printer or distribute to end users.
- A Resources folder with **various ancillary files**, such as stylesheets and images.

If you want users to download an XPS document from a server, you need to enable the server to do this by registering the MIME types and file extensions. For steps see the online Help.

Word

Word is an output type where the generated project is exported to Microsoft Word in one of the following formats: XML (default), DOC, DOCX, XPS, PDF. However, you can also create PDF or XPS output directly, without going through Word.

Note: Unless you specify otherwise, the Word target will create files with an .xml extension only. If you want to use one of the other formats see the online Help.

Note: Flare supports Microsoft Word 2003 and newer versions.

Note: To create output in DOCX or XPS format, you need to have Microsoft Word 2007 installed. Also, Word 2007 allows you to create PDF files from Word without needing to have the Adobe Distiller installed.

FrameMaker

FrameMaker is an output type where the generated project is exported to Adobe FrameMaker in one of the following formats: BOOK, FM, PDF. However, you can also create PDF output directly, without going through FrameMaker.

Note: Flare supports FrameMaker 7.0 and newer versions.

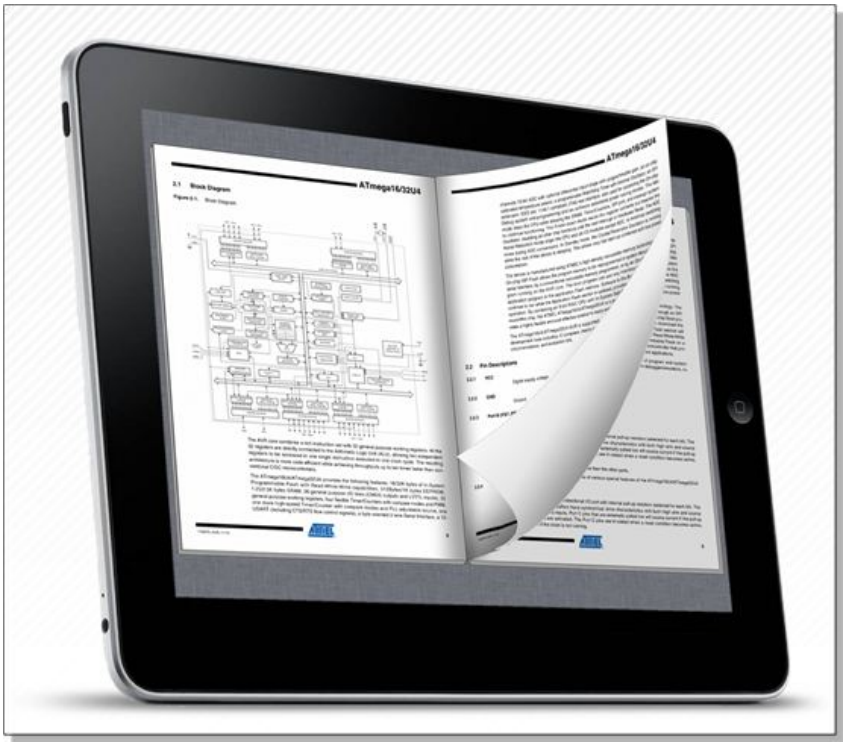
Other Output Types

Following are overviews of two additional output types—EPUB and DITA.

EPUB

EPUB is an online format for creating reflowable digital books (ebooks). This format is developed and maintained by the International Digital Publishing Forum (IDPF). For additional information about this group and the EPUB format, see **http://idpf.org/**.

EPUB is sort of a hybrid format. The output is designed to be viewed electronically; in that way, it is like an online format. On the other hand, its structure is like a book or manual; in that way, it is like a print-based format. In fact, you can follow many of the same steps for creating PDF output to produce EPUB output.



FEATURES

Following are some of the features of the EPUB format in Flare:

- **Reflowable** EPUB output allows for reflowable content, such as word wrap and resizeable text. Therefore, the content can be displayed optimally on different devices.
- **Single File** You only need to distribute a single EPUB file, rather than a collection of files and folders.

  Note: If you extract the EPUB file using another tool, you will find that it contains one or more XHTML files. If you have used chapter breaks in the outline TOC, a new XHTML file is created at each point where a chapter break occurs. If you do not use chapter breaks, the EPUB contains just one XHTML file.

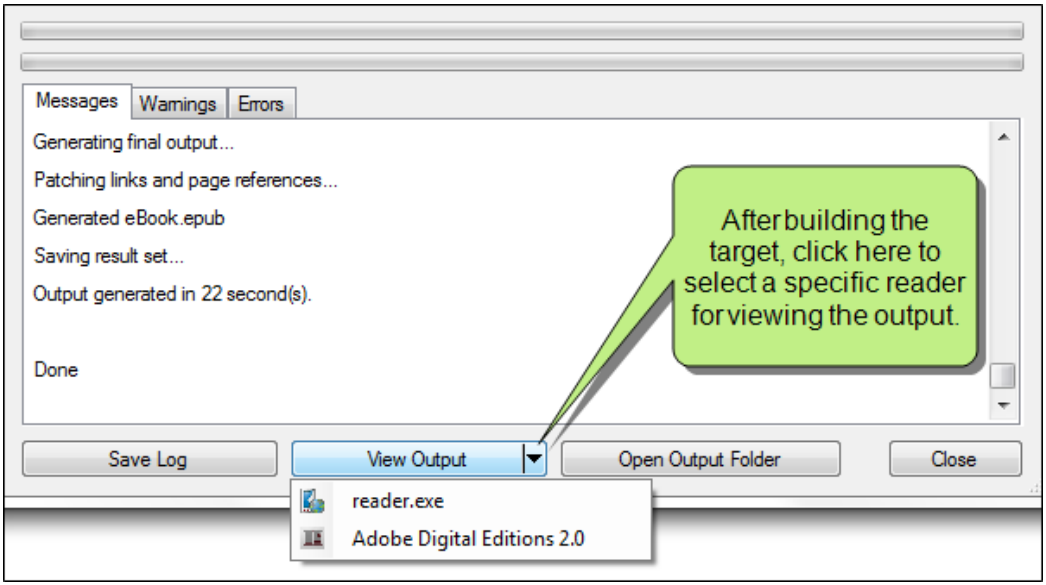
- **Embedded Metadata** In the EPUB target you can provide information such as a title, author, publisher, rights, ISBN, identifier, which become embedded in the final output. The document title you provide will usually be displayed at the top of the EPUB reader. If you do not provide a title, the target name will automatically be used instead.

  For more information about EPUB metadata and the fields supported in Flare (except "Tag"), see:

  http://idpf.org/epub/20/spec/OPF_2.0.1_draft.htm#Section2.2

- **Embedded Fonts** Flare includes all of your used fonts in the final EPUB output file. Therefore, if someone has a reader that does not have your fonts installed, that person will still be able to see your content as you designed it.
- **Cover Page** You can use the Target Editor to quickly and easily select an image file to be used as the cover for your ebook.
- **Validate EPUB Output** You can validate EPUB output so that it will be acceptable to most online stores, such as amazon.com.
- **MOBI Output** MOBI is the standard format used for the Amazon Kindle. You can use settings in the Target Editor in Flare to generate EPUB output that uses MOBI content.
- **MathML to PNG Conversion** If you use the Equation Editor to insert equations, you are in essence creating MathML markup. This type of content is supported for EPUB output. If you need to produce output that can be displayed on older readers, you can select an option in the Target Editor to convert all of your MathML content to PNG image files.

» **Select Reader to View Output** You can select a specific EPUB reader (if you have more than one installed) to view the output.



READING SYSTEMS

EPUB output can be viewed on many kinds of software and hardware, across many platforms. Following are just a few examples of reading systems for EPUB output:

- Amazon Kindle (MOBI)
- Barnes & Noble Nook
- Bluefire Reader (Android, iOS)
- EPUBReader (Firefox)
- Duokan (Kindle)
- iPad, iPhone, iPod Touch
- Lexcycle Stanza (iOS, Windows, Mac OS X)
- Mobipocket (Windows, Blackberry, Symbian, Windows Mobile)
- Sony Reader

Important: Some readers handle EPUB output and respect your formatting better than others.

DITA

Darwin Information Typing Architecture (DITA) file content is supported in Flare. DITA is an XML-based markup language with its own schema for authoring, producing, and delivering technical information. It is a standard of the Organization for the Advancement of Structured Information Standards (OASIS), and it consists of a set of design principles for creating "information-typed" modules at a topic level and for using that content in various delivery modes.

DITA allows companies (especially larger ones) to maintain better consistency throughout its documentation by establishing structural rules and standards for all of its authors to follow. The idea is that writers will spend more of their time authoring content, rather than worrying about the presentation of that information.

In Flare you can generate output that produces DITA files. When you build this type of output, a DITA map file is generated, with multiple DITA files in it. The XHTML tags are converted to DITA elements. In other words, although it is considered an "output" from the standpoint of the Flare process, the end result is actually a collection of "source" files, which you can later use in another tool (or import back into Flare) to produce the final output.

CHAPTER 3

Adding Targets

You can add as many targets as you want to a project.

HOW TO ADD TARGETS

1. Do one of the following, depending on the part of the user interface you are using:

   - **Ribbon** Select the **Project** ribbon. In the **Content** section select **New>Target**.

     You can use the Options dialog to switch between ribbons and the classic tool strip layout. For more information see the online Help.

     Keep in mind that the smaller the application window becomes, the more the options in a ribbon shrink. Therefore, you might only see a small icon instead of text, or you might see only a section name displayed with a down arrow to access the options in it. You can hover over small icons to see tooltips that describe them. You can also enlarge the application window or click one of the section drop-downs in the ribbon to locate a hidden feature.

   - **Menu** Select **Project>Add Target**.
   - **Right-Click** In the Project Organizer, right-click on the **Targets** folder and from the context menu select **Add Target**.

   The Add File dialog opens.

2. In the **File Type** field at the top, make sure **Target** is selected.

3. In the **Source** area select one of the following:

   - **New from template** This lets you choose either the factory template file or one of your own customized template files as a starting point. The new file will take on all of the settings contained in the template. If you want to use the factory template provided by Flare, expand the **Factory Templates** folder and click on a template file. If you want to use your own customized template file, expand the appropriate folder and click on a file. For more information about templates, see the online Help.

     > **Note:** In some dialogs and wizards you can click the **Manage Templates** button if you want to open the Template Manager. This lets you manage any of your template files (e.g., add new templates, enter descriptions for templates). For more information see the online Help.

   - **New from existing** This lets you choose an existing file of the same type as a starting point for your new file. As with template files, your new file will take on all of the settings contained in the file you select. To use this option, click the browse button ..., use the Open File dialog to find a file, and double-click it.

4. (Optional) If you want to place the file into a subfolder that you previously created in the Project Organizer, in the **Folder** field click ... and select the subfolder. Otherwise, keep the default location.
5. In the **File Name** field, type a new name for the target.
6. From the **Output Type** field, select one of the available output formats. You can always change the output type later in the Target Editor.
7. Click **Add**. The target is added to the Targets folder in the Project Organizer. The Target Editor opens to the right.

HOW TO MAKE COPIES OF TARGETS

1. Make sure the Project Organizer is open.
2. Double-click the **Targets** folder. The available targets are shown.
3. Click on the target that you want to copy.
4. Press **CTRL+C** on your keyboard.
5. Press **CTRL+V** on your keyboard. A copy of the target is added to the Targets folder. The new target is given the same name as the target you copied, with the words "Copy of" before it.

CHAPTER 4

Importing Targets

Not only can you add a new target, but you can also import an existing target (FLTAR file).

HOW TO IMPORT A TARGET

1. Do one of the following, depending on the part of the user interface you are using:
   - **Ribbon** Select the **Project** ribbon. In the **Content** section select **New>Target**.
   - **Menu** Select **Project>Add Target**.
   - **Right-Click** In the Project Organizer, right-click on the **Targets** folder and from the context menu select **Add Target**.

   The Add File dialog opens.
2. Select **New from existing** and click ...
3. Find and select the FLTAR file that you want to import.
4. Click **Open**. The Source File field now contains the path to the file that you are importing. Also, the name of the file is displayed in the File Name field.
5. If you want to give the target a different name than that for the imported file, click in the **File name** field and replace the text.
6. Click **Add**. The target is added and opens in the Target Editor.

Note: As an alternative to importing single files from another location, you can import linked files from a Flare source project using a featured called "Global Project Linking." This is different than a simple import process, because in this case, the imported files remain linked to the source project. This allows you to make future updates to those files in just one place—in the source project file. When you perform ongoing imports using your previous settings, Flare recognizes changes to the source files. Therefore, the new files can be brought over, replacing the outdated files.

CHAPTER 5

Renaming Targets

When you create a new project using one of Flare's factory templates, one or more new targets are added to your project.

It is usually helpful to rename the targets that you use to reflect the nature of your project (especially if you are using multiple targets with the same output type).

EXAMPLE

If you are producing a Help system for a software application called FictionSoftPro, you might want to rename the target "FictionSoftPro." If you are creating a beginner's version and an advanced version of your Help system, you might want to name one target "Beginner FictionSoftPro" and the other target "Advanced FictionSoftPro." That would be much easier than trying to distinguish between targets called "MyDotNetHelp" and "Copy of MyDotNetHelp."

HOW TO RENAME A TARGET

1. Make sure the Project Organizer is open.
2. Double-click the **Targets** folder. The available targets are shown.
3. Click on the target that you want to rename.
4. Press **F2**. The name in the target is highlighted.
5. Type a new name for the target and press **Enter** on your keyboard. The target is renamed.

CHAPTER 6

Opening Targets

When you want to modify an existing target, use the following steps to open it. You can use the first set of steps to quickly open the primary target, or you can use the second set of steps to open a specific target (not necessarily the "primary" target).

HOW TO OPEN THE PRIMARY TARGET

Do one of the following, depending on the part of the user interface you are using:

- **Ribbon** Select the **Project** ribbon. In the **Primary Files** section select **Primary Target**.
- **Project Toolbar** Click Primary Target.

  To open the Project toolbar from the menu view, select **View>Toolbars>Project**.
- **Keyboard Shortcut** Press **CTRL+F9**.

HOW TO OPEN A SPECIFIC TARGET

1. Make sure the Project Organizer is open.
2. Double-click the **Targets** folder. The targets in your project are displayed.
3. Double-click the target that you want to open. The Target Editor opens to the right, with the target page shown.

CHAPTER 7

Setting a Primary Target

You can have as many targets in a project as you want, and any of your targets can be built (generated) whenever you like. However, chances are that you will have one target that you work with more than the others. Therefore, you can set it as your primary target.

When you first create a project, the Start New Project Wizard asks you to specify a primary target. The steps below show you how to change which target is the primary one.

A primary target is treated just like any of your other targets, with one exception. There are certain shortcuts in Flare that let you build, view, or publish your primary target more quickly.

HOW TO SET A PRIMARY TARGET

1. Make sure the Project Organizer is open.
2. Double-click the **Targets** folder. The available targets are shown.
3. Right-click on the target that you want to make the primary target. In the context menu, select **Make Primary**. The new primary target has a green arrow next to it with "(Primary)" added after the name of that target.

CHAPTER 8

Editing Target Settings

Using the Target Editor, you can edit properties for any of your targets.

This chapter discusses the following:

Applying Empty Alt Text 62
Associating a Browse Sequence with a Target 64
Associating Condition Tags with Targets 65
Associating an Alias File with a Target 84
Adding the DOCTYPE Declaration to Generated Topics 85
Specifying EPUB Options 87
Enabling Feedback in Flare 93
Associating Glossaries with Targets 94
Specifying Buttons for HTML Help Output 96
Displaying Merged Navigation in HTML Help Child Outputs 98
Patching Stylesheets and Image Links for HTML Help 100
Setting Up an HTML5 Target 101
Embedding Images in Microsoft Word Output 102
Generating Web-Safe Images 103
Disabling Auto-Sync of Import Files 104
Excluding Index Entries From Search 105

Selecting a Language 106
Adding Mark of the Web 108
Associating Master Pages with Targets 110
Disabling the Use of Master Pages 111
Adding Custom Meta Tags 112
Specifying the Output File and Folder in a Target 115
Replacing Characters and Spaces with Underscores 116
Using Custom File Extensions 117
Excluding Content not Linked Directly or Indirectly from Targets 118
Converting Output File Names and Folders to Lowercase 121
Omitting the Content Folder from Output 122
Changing the Output Type for a Target 123
Associating Master Page Layouts with Targets 124
Specifying PDF Options 125
Specifying PDF Output Via Word or FrameMaker 136
Collapsing Top Margins 138
Improving the Processing Performance of the Target 140
Specifying How Online Features are Converted in Printed Output 143
Associating Publishing Destinations with Targets 145
Integrating Flare and Pulse 146
Setting Redacted Text on Targets 174
Associating Relationship Tables with Targets 175
Associating a Search Filter Set with a Target 176
Enabling Partial-word Search 177
Generating Sitemaps for SEO 179
Associating Skins with Targets 180
Setting All Skins to be Generated 181

Getting the Latest Version 182
Specifying the Startup Topic in a Target 184
Applying Master Stylesheets 185
Associating a Medium with a Target 187
Associating a Master Table of Contents with a Target 188
Using the TOC Depth for Heading Levels in Print Output 189
Creating Headings for Unlinked Books in a Generated TOC 192
Removing Images from Headings in a Generated TOC 195
Overriding Variable Definitions in Targets 196
Specifying WebHelp AIR Options 198
Setting Up a WebHelp Plus Target 199
Enabling Mirror Margins in Word Output 201
Specifying XPS Output Via Microsoft Word 202
Ignoring Warnings When Building Output 204

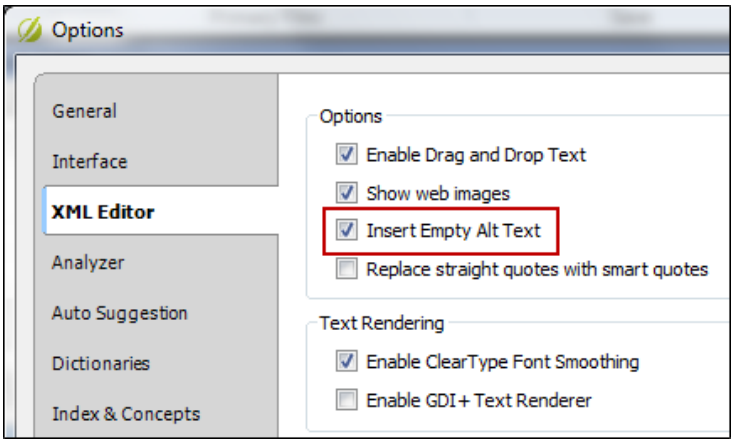
Applying Empty Alt Text

This feature is supported in Adobe PDF and in all online outputs (DotNet Help, Eclipse Help, Microsoft HTML Help, HTML5, WebHelp, WebHelp AIR, WebHelp Mobile, and WebHelp Plus.

By default, a new image does not contain an "alt" (alternate text) attribute when you insert it into a topic or snippet, unless you specifically add alt text in the Insert Image dialog. However, alt attributes on images are required for Section 508 compliance, even if the string is empty or null (alt=""). Also, by adding empty alt attributes to images, you can search the files to find the images without alternate text and then add it as needed.

There are a couple of ways to quickly add empty alt tags to images:

- **When Output is Generated** You can select an option in the **Advanced** tab of the Target Editor. As a result, empty alt tags are automatically added to images that do not already have them. The empty alt attribute gets added to the output when the target is generated.
- **When Images are Inserted** You can use a global setting in the Options dialog. This automatically adds an empty alt tag to images when you insert them into a topic or snippet (unless you specifically add an alt tag to an image).



In addition to images, this also works for QR codes and equations, since those elements are converted to images in the output.

HOW TO APPLY EMPTY ALT TEXT WHEN OUTPUT IS GENERATED

1. Open a target.
2. On the **Advanced** tab of the Target Editor, select **Use empty ALT text for images that do not have ALT text**.
3. Click to save your work.

HOW TO APPLY EMPTY ALT TEXT WHEN IMAGES ARE INSERTED

1. Do one of the following, depending on the part of the user interface you are using:
   - **Ribbon** Select **File>Options**.

     You can use the Options dialog to switch between ribbons and the classic tool strip layout. For more information see the online Help.

     Keep in mind that the smaller the application window becomes, the more the options in a ribbon shrink. Therefore, you might only see a small icon instead of text, or you might see only a section name displayed with a down arrow to access the options in it. You can hover over small icons to see tooltips that describe them. You can also enlarge the application window or click one of the section drop-downs in the ribbon to locate a hidden feature.
   - **Menu** Select **Tools>Options**.

   The Options dialog opens.
2. Select the **XML Editor** tab.
3. Select the **Insert Empty Alt Text** check box. A check mark indicates the feature is enabled.
4. Click **OK**.

Associating a Browse Sequence with a Target

This feature is supported in DotNet Help, Microsoft HTML Help, HTML5, WebHelp, WebHelp AIR, WebHelp Mobile, and WebHelp Plus output.

After you enable a browse sequence in a skin, you need to associate it with a target (unless it is linked to another browse sequence that is associated with a target).

If you have created only one browse sequence for your project, you do not need to associate it with a target. It will display automatically after you build the target. However, if you have added more browse sequences, you need to specify which one will serve as the "master browse sequence." This is the browse sequence that will be displayed in the output. The additional browse sequences will also be displayed if you have linked to them from the master browse sequence. Use the following steps to specify a master browse sequence, associating it with a target.

HOW TO ASSOCIATE A BROWSE SEQUENCE WITH A TARGET

1. In the Project Organizer, open the target.
2. In the Target Editor, click the **General** tab.
3. Click the drop-down arrow in the **Browse Sequence** field, and select the browse sequence to associate with the target.
4. Click to save your work.

Associating Condition Tags with Targets

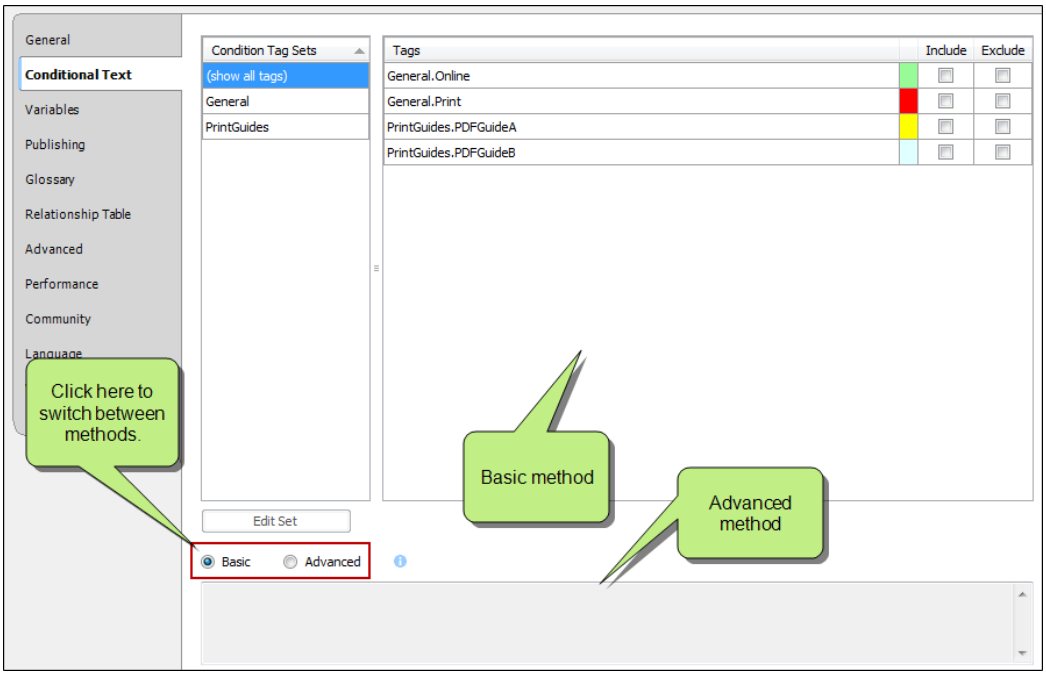
This feature is supported in all outputs.

After creating and applying condition tags, you need to tell Flare what your target should do with the condition tags that you have created and applied. Should content with a particular condition tag be included in or excluded from that target?

HOW TO ASSOCIATE CONDITION TAGS WITH A TARGET

1. Make sure the Project Organizer is open.
2. Double-click the **Targets** folder. The folder opens, showing the targets in your project.
3. Double-click the target that you want to associate with one or more condition tags. The Target Editor opens to the right.
4. Click the **Conditional Text** tab.
5. There are two ways to use the Conditional Text tab—Basic and Advanced. By default when you open this tab for the first time, the **Basic** option is selected. You can click **Advanced** to switch to that mode if you are experienced at writing condition tag expressions by hand. While the Basic section is easier for most people to use, it is also more limited in the type of expressions it can create.

   With the Basic method, all of your work is done on the top half of the user interface, by just clicking check boxes. On the other hand, with the Advanced method, most of your work is done on the bottom portion of the user interface.



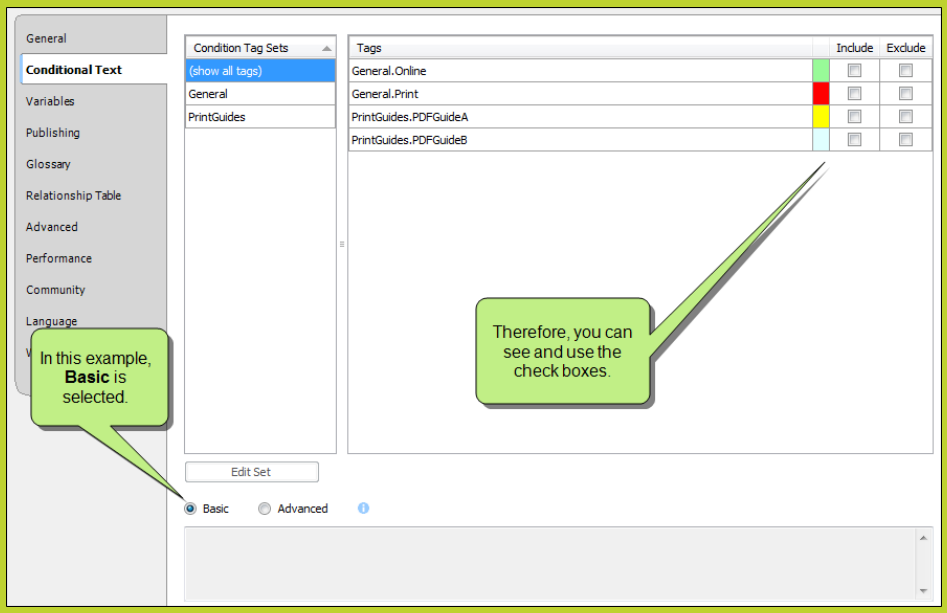
Note: If you are an experienced user and want to use the Advanced mode, you might find it helpful to begin with the Basic mode to create the initial expression and then switch to Advanced when you are ready.

INSTRUCTIONS FOR THE BASIC METHOD

In the **Condition Tag Sets** area, you can choose to view tags for all condition tag sets or you can select a specific set. The tags associated with the selected set are shown to the right, with their associated colors. An Include and Exclude check box appears next to each condition tag.

When Basic is selected, the check boxes are shown and the Advanced section is disabled.

E X A M P L E



If you want to exclude a condition tag, click the **Exclude** check box next to it. If you want to make sure a condition tag is included, click the **Include** check box next to it. The primary reason for having Include check boxes is to account for possible conflicts.

E X A M P L E

Suppose you have two condition tags—one called "Beginner" and another called "Advanced." Let's say that you have a topic containing three paragraphs. You apply the Advanced tag to the first two paragraphs, and you apply the Beginner tag to the last two paragraphs.

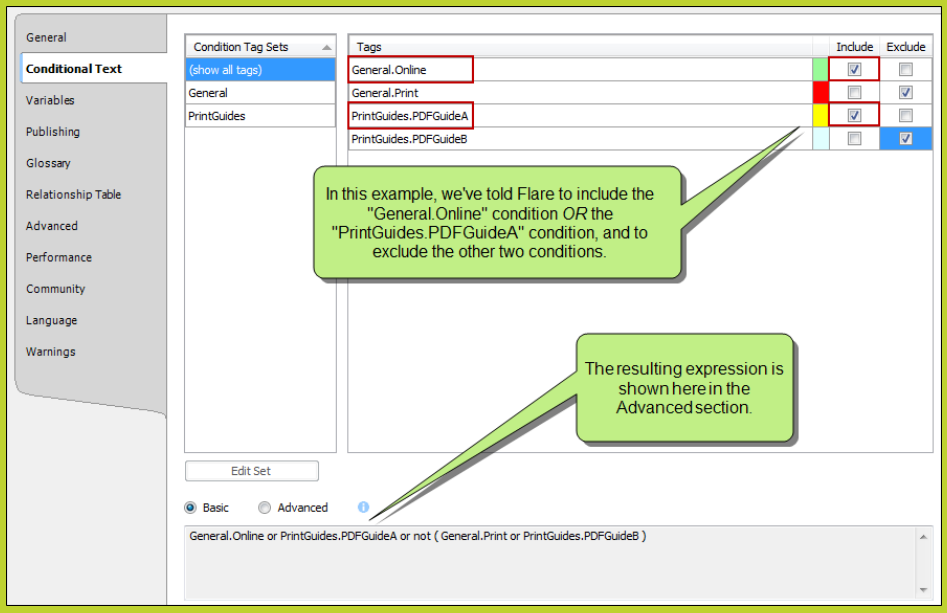
You have created a target called "Advanced Set Up." For this target, you obviously want to include all content associated with the Advanced tag, but you want to exclude content associated with the Beginner tag.

By default, Flare will include content associated with both tags, unless you tell it not to. So you tell Flare to exclude the content associated with the Beginner tag. The problem is the middle paragraph from the topic mentioned above. It is associated with both tags. You have told Flare to exclude content associated with the Beginner tag, and it will do so, overriding the default. But you want to make sure that paragraph is included in the "Advanced Set Up" output. That is why you need to make sure you select the Include check box next to the Advanced tag.

Warning: If you do not select any check boxes at all, all of the tags will automatically be *included*. If you include only some tags and exclude others, any remaining tags without check marks for either Include or Exclude will automatically be *included*. However, if you select Include for any of the tags (even just one) and do not select Exclude for any tags, all of the other tags that do not have Include check marks will automatically be *excluded*. If you want to be safe and always know that the correct tags are included or excluded, you can make sure that all of the tags have check marks with either Include or Exclude.

When you click check boxes in the Basic section, you are creating expressions with "or" statements (e.g., "include this condition tag *or* that one"). And when you do this, you can see the resulting expression displayed in the Advanced section below.

EXAMPLE



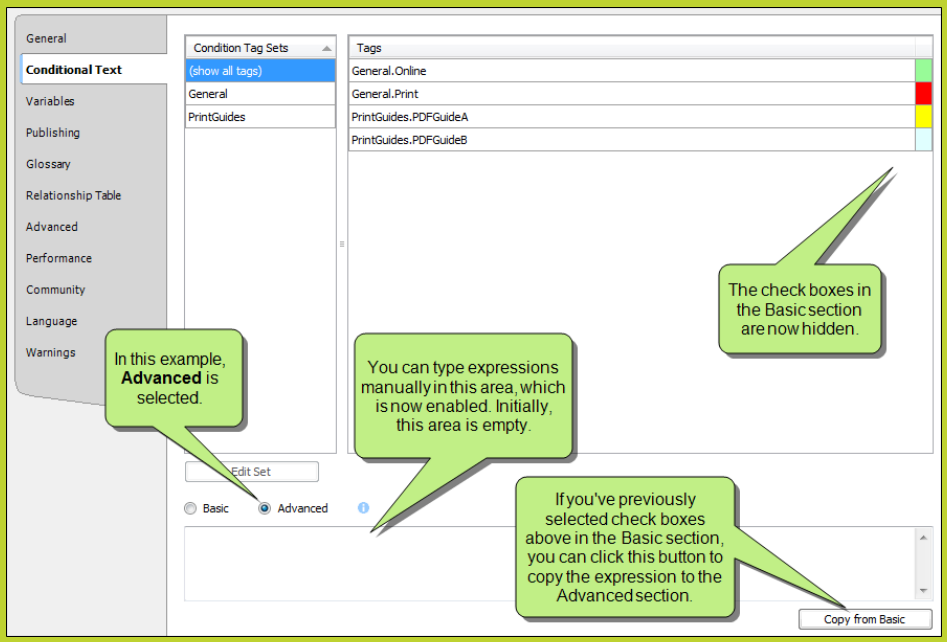
INSTRUCTIONS FOR THE ADVANCED METHOD

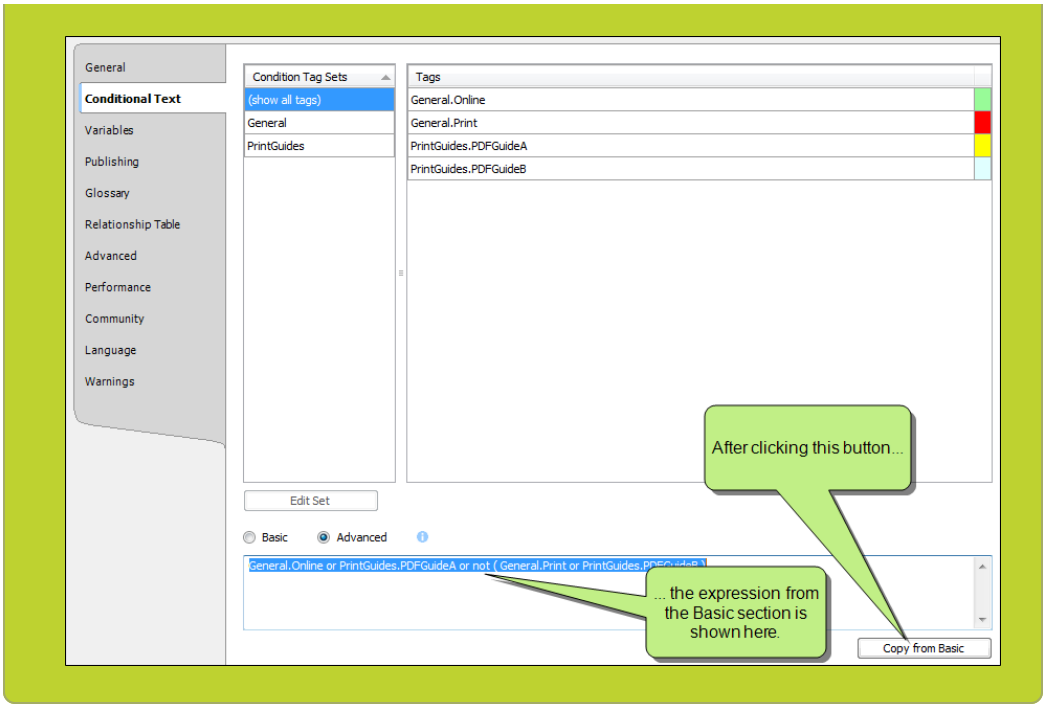
If you are a more experienced user, you may decide to use the Advanced section to write expressions manually, rather than clicking check boxes. You can type the names of the condition tags as well as any of the following tokens: OR, AND, NOT, (). This allows you to produce more complex, robust expressions to control your output (i.e., "and" statements, as well as "or" statements).

COPY FROM BASIC

When you select Advanced for the first time, the Basic section becomes disabled (i.e., the check boxes are hidden), and the Advanced section is enabled but empty (even if you've previously selected conditions in the Basic section). However, you can copy the expression from the Basic section to the Advanced section. This is a quick and easy way to create an initial expression and then tweak it with "and" statements.

E X A M P L E

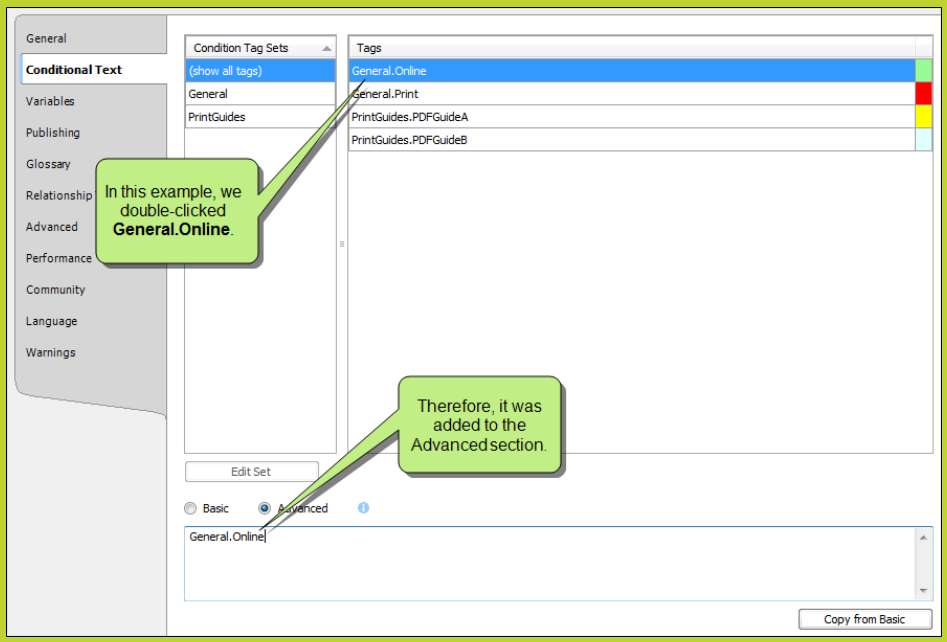


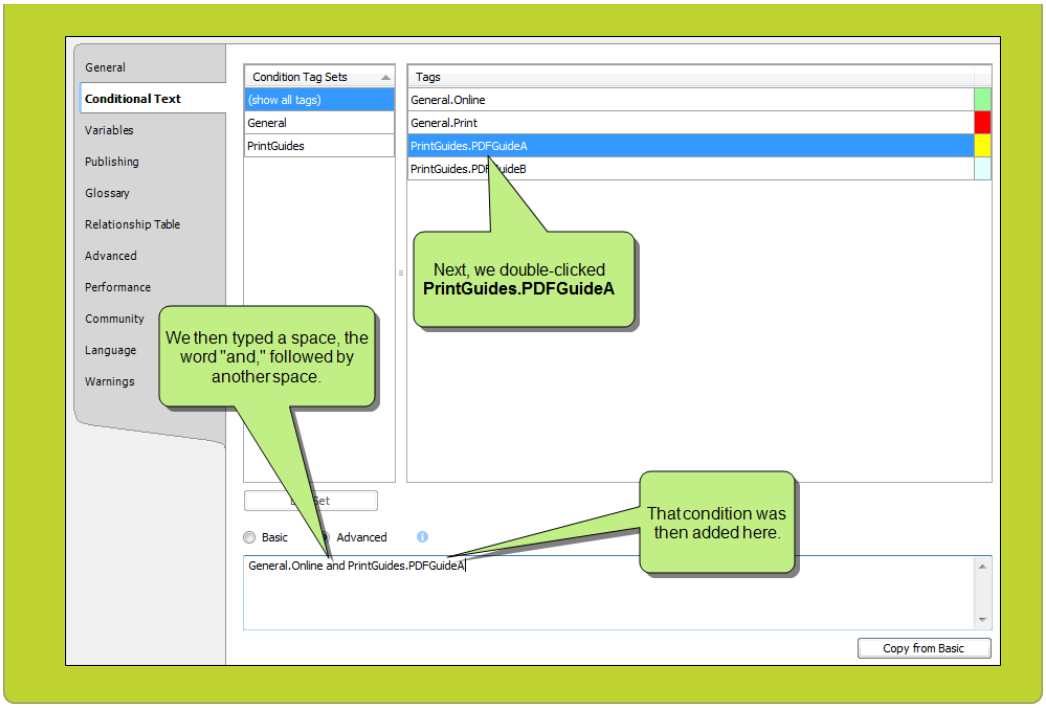

General
Conditional Text
Variables
Publishing
Glossary
Relationship Table
Advanced
Performance
Community
Language
Warnings
Condition Tag Sets
(show all tags)
General
PrintGuides
Tags
General.Online
General.Print
PrintGuides.PDFGuideA
PrintGuides.PDFGuideB
Edit Set
Basic
Advanced
General.Online or PrintGuides.PDFGuideA or not (General.Print or PrintGuides.
After clicking this button...
... the expression from the Basic section is shown here.
Copy from Basic

DOUBLE-CLICK TAGS

Another trick you can use while working in the Advanced section is to double-click any tag from the Basic section. This adds the condition to the Advanced section so you do not have to type it. *Just make sure your cursor has been placed in the Advanced section before double-clicking a condition tag. If you don't click in the Advanced section first, the newly added condition tag will overwrite all of the text you've entered previously in the Advanced section.*

EXAMPLE

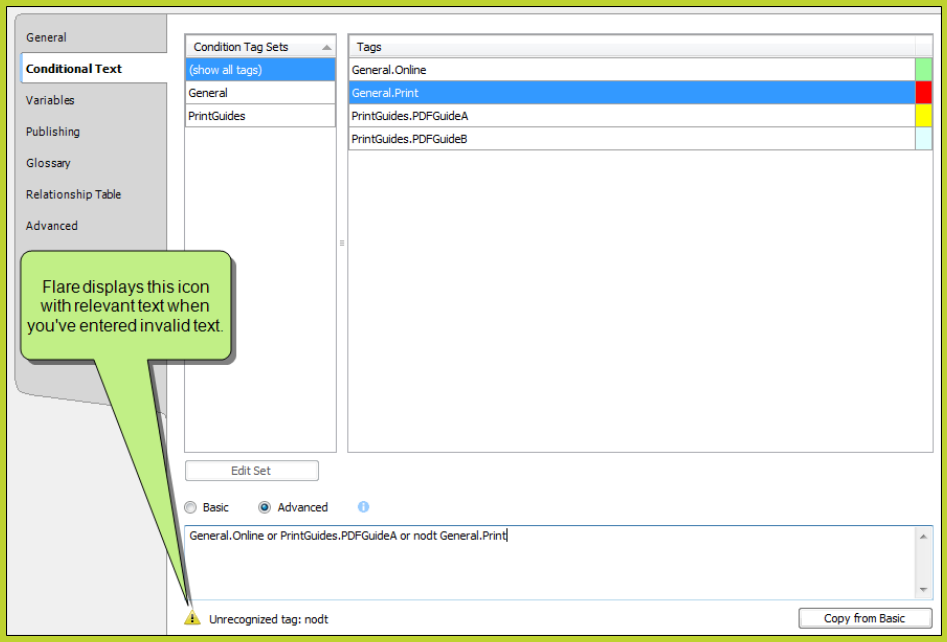



General
Conditional Text
Variables
Publishing
Glossary
Relationship Table
Advanced
Performance
Community
Language
Warnings
Condition Tag Sets
(show all tags)
General
PrintGuides
Tags
General.Online
General.Print
PrintGuides.PDFGuideA
PrintGuides.PDFGuideB
Next, we double-clicked PrintGuides.PDFGuideA
We then typed a space, the word "and," followed by another space.
That condition was then added here.
Basic
Advanced
General.Online and PrintGuides.PDFGuideA
Copy from Basic

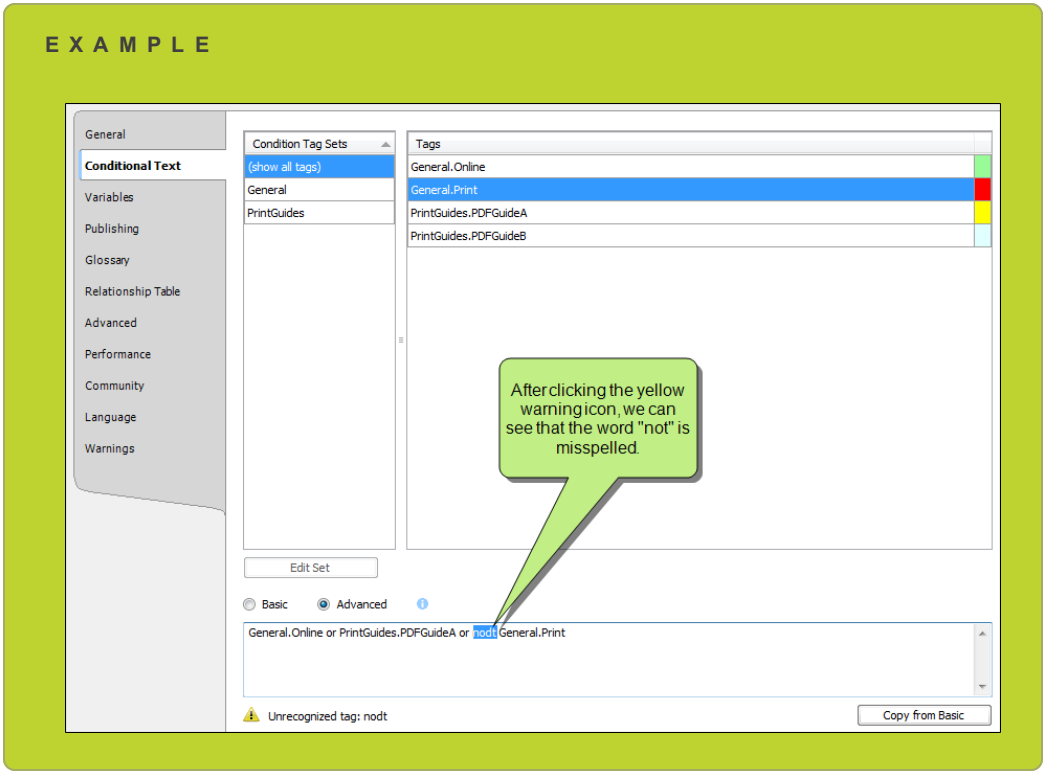
WARNING ICON

When you type in the Advanced section, a yellow warning icon displays with relevant text if you have typed an error.

EXAMPLE



You can expect to see this icon quickly appear and then disappear as you type. However, if the icon remains after you finish, you can click it. The error will then be highlighted in the text below.



6. Click to save your work.

EXAMPLES OF CONDITION EXPRESSIONS

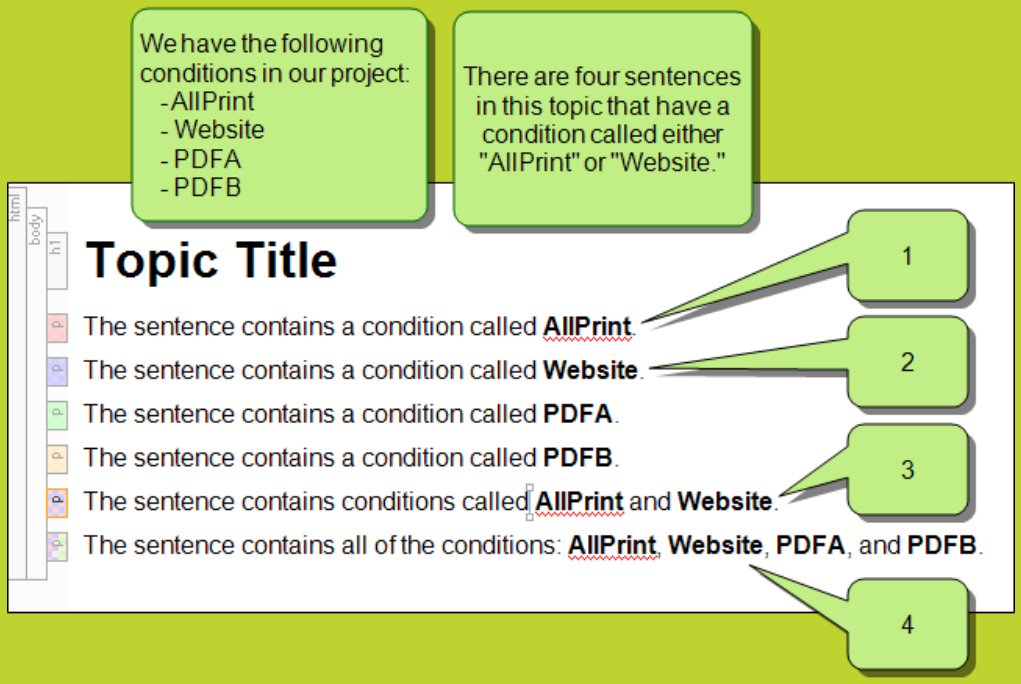
Following are some examples of condition expression strings.

E X A M P L E — O R

```
Primary.Website OR Primary.AllPrint
```

This is a simple expression that includes content that has either Website *or* AllPrint conditions applied to it.

So let's say you have a topic that looks like this.



In the output, you will see this.

Topic Title

The sentence contains a condition called **AllPrint**.

The sentence contains a condition called **Website**.

The sentence contains conditions called **AllPrint** and **Website**.

The sentence contains all of the conditions: **AllPrint**, **Website**, **PDFA**, and **PDFB**.

EXAMPLE — AND

```
Primary.Website AND Primary.AllPrint
```

This expression includes content that has Website *and* AllPrint conditions applied to it. *However, the output can also include other conditions not listed in the expression, since they are not explicitly excluded.*

So let's say you have a topic that looks like this.

We have the following conditions in our project:
- AllPrint
- Website
- PDFA
- PDFB

There are two sentences that have both conditions "AllPrint" and "Website."

Topic Title

The sentence contains a condition called **AllPrint**.

The sentence contains a condition called **Website**.

The sentence contains a condition called **PDFA**.

The sentence contains a condition called **PDFB**.

The sentence contains conditions called **AllPrint** and **Website**. 1

The sentence contains all of the conditions: **AllPrint**, **Website**, **PDFA**, and **PDFB**. 2

In the output, you will see this.

Topic Title

The sentence contains conditions called **AllPrint** and **Website**.

The sentence contains all of the conditions: **AllPrint**, **Website**, **PDFA**, and **PDFB**.

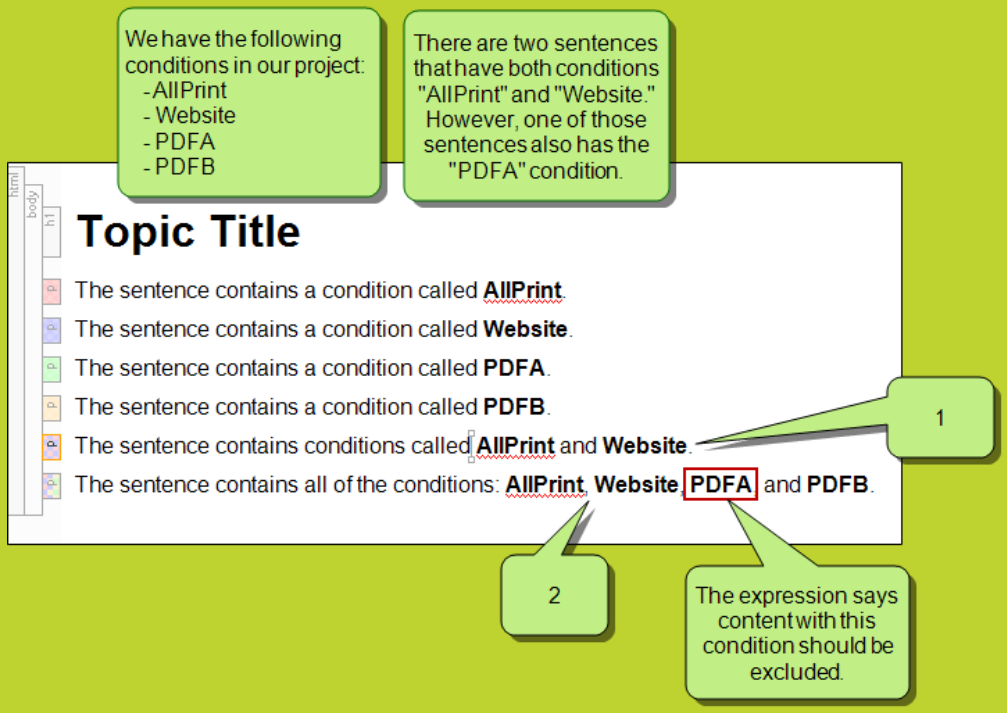
Notice that the second sentence is included even though it contains not only the AllPrint and Website conditions, but also the PDFA and PDFB conditions.

EXAMPLE—AND/AND NOT

```
Primary.Website AND Primary.AllPrint AND NOT Primary.PDFA
```

This expression includes content that has Website *and* AllPrint conditions applied to it. In addition, it specifically excludes content with the PDFA condition.

So let's say you have a topic that looks like this.



In the output, you will see this.

Topic Title

The sentence contains conditions called **AllPrint** and **Website**.

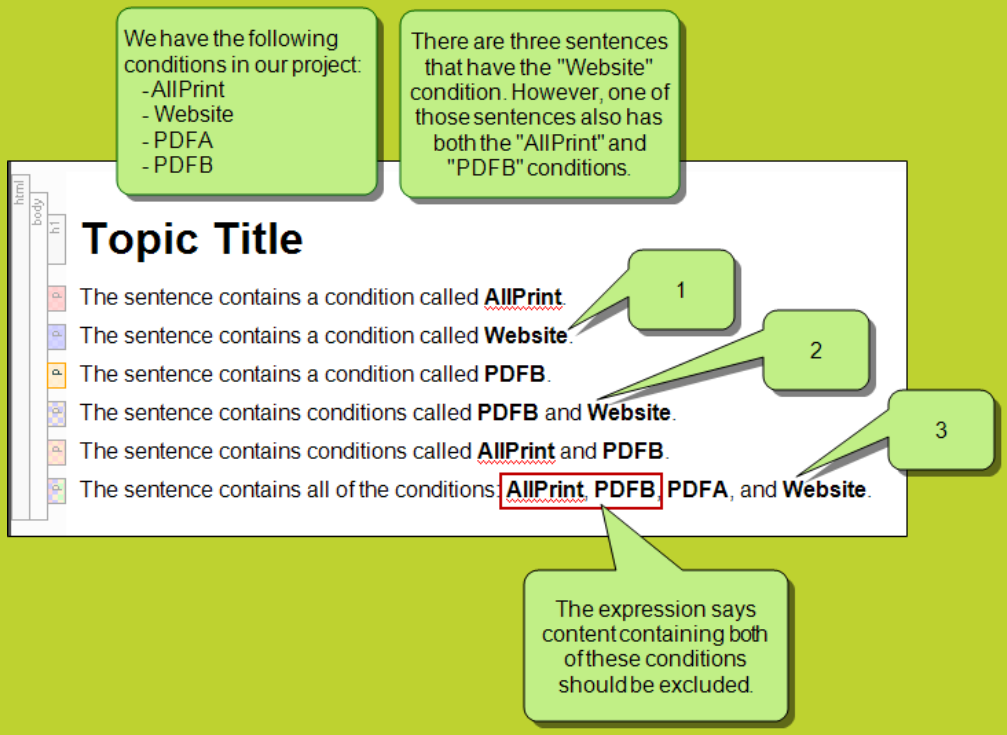
The second sentence with both Website and AllPrint does not display in the output because the PDFA condition is in that sentence too, and it was set to be excluded.

EXAMPLE—AND NOT (AND)

```
Primary.Website AND NOT (Primary.AllPrint AND Primary.PDFB)
```

This expression includes any content with the Website condition applied to it. In addition, it specifically excludes content with both the AllPrint and PDFB conditions.

So let's say you have a topic that looks like this.



In the output, you will see this.

Topic Title

The sentence contains a condition called **Website**.

The sentence contains conditions called **PDFB** and **Website**.

The sentence that had both the AllPrint and PDFB conditions was excluded. However, another sentence that has the Website and PDFB conditions was included. That's because the exclusion only occurs if *both* the AllPrint and PDFB conditions have been applied to the sentence.

Warning: When you switch between the Basic and the Advanced view, your work is preserved independently in both. However, be cautious when using the copy button. Because of the way this feature is designed, it is a best practice to use *either* the Basic section or the Advanced section in the long term, *but not both*. Although it is possible to copy expressions from the Basic to the Advanced section, and this can save you lots of time, you should not try to switch between sections over time. The reason for this is that every time you make changes in the Basic section and click the button to copy the expression to the Advanced area, your previous advanced expression will be overwritten.

Note: When you generate the target, Flare uses the expression from the section that is selected—Basic or Advanced.

Associating an Alias File with a Target

This feature is supported in DotNet Help, Microsoft HTML Help, HTML5, WebHelp, WebHelp Mobile, and WebHelp Plus output.

If you have added more than one alias file for your project, you need to associate the appropriate alias file with the target that you plan to build. If you do not specify an alias file in a target, Flare uses the first alias file listed in the Project Organizer.

HOW TO ASSOCIATE AN ALIAS FILE WITH A TARGET

1. Open the target.
2. In the Target Editor, click the **Advanced** tab.
3. Click the drop-down arrow in the **Alias File** field, and select the alias file that you want to associate with the target.
4. Click to save your work.

Adding the DOCTYPE Declaration to Generated Topics

This feature is supported in DotNet Help, Microsoft HTML Help, WebHelp, WebHelp AIR, and WebHelp Plus output.

You can add the DOCTYPE declaration to topic files when you generate online output. This allows browsers to render the topics in strict mode. If you do not select this option, generated topics will not have this declaration and will be rendered by browsers in quirks mode.

Quirks mode and strict mode have to do with the evolution of web browsers and the rules they use to interpret styles in cascading stylesheets (CSS). Quirks mode follows the old rules, and strict mode follows the new rules. If you are not concerned about which mode is used for your online output, you do not need to add the DOCTYPE declaration to topics. However, if you want to ensure that your output is interpreted and displayed using the newer strict mode, you should use this option.

Why use the DOCTYPE declaration? You might find the need to use the DOCTYPE declaration feature if you have text boxes, images, tables, or other objects that have float, margin, or padding settings applied to them. With settings such as this, you might notice slight alignment issues when generating online output. For example, margin or padding settings might be pushing aligned text a bit further than you want. To fix this, add the DOCTYPE declaration to the target.

HOW TO ADD THE DOCTYPE DECLARATION TO GENERATED TOPICS

1. Open a target used for online output.
2. On the **Advanced** tab of the Target Editor, select **Add DOCTYPE declaration to generated topics**.
3. Click to save your work.

Note: Enabling this feature for HTML Help targets may be necessary to avoid certain custom JavaScript errors. However, for some languages (e.g., right-to-left, Russian, Asian languages), this option may not function properly in HTML Help. Therefore, it is recommended that you disable the option for non-English language content when generating HTML Help.

Note: The DOCTYPE declaration will exist in the main output file whether or not this option is enabled.

Note: This option is not necessary for WebHelp Mobile output because that output type always includes the standard mobile DOCTYPE.

Specifying EPUB Options

If you are sending output to EPUB, you can access EPUB options in the Target Editor. These options let you provide metadata, select a cover image, validate output, generate MOBI output, and enable MathML to PNG conversion.

HOW TO SPECIFY EPUB OPTIONS

1. Open a target that is using the EPUB format. If necessary, you can change the output type for a target. See "Changing the Output Type for a Target" on page 123.
2. Select the **EPUB Options** tab.
3. Complete the options as necessary.

   METADATA

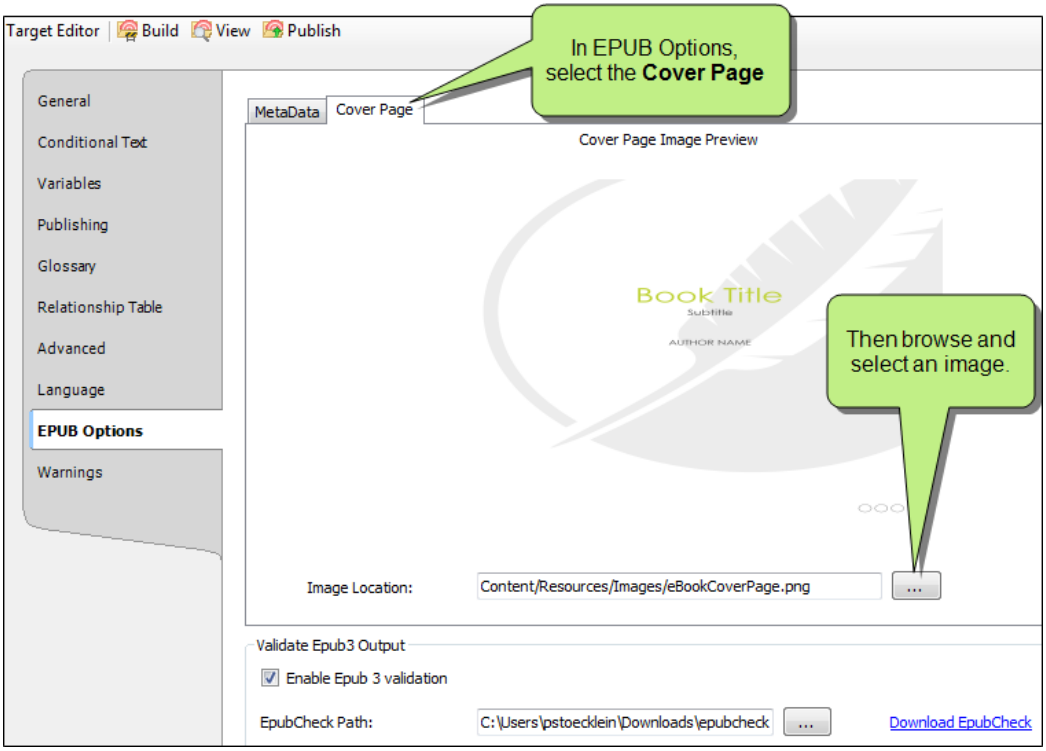
   Select the **MetaData** tab and complete any of the following. The information becomes embedded in the final output.

   - **Title** The document title you provide is usually displayed at the top of the EPUB reader. If you do not provide a title, the target name is automatically used instead.
   - **Author**
   - **Publisher(s)**
   - **Tag** This is a custom field that you can use for your own purposes.
   - **Rights**
   - **Description**
   - **ISBN**
   - **Publish Date**
   - **Identifier**

   For more information about EPUB metadata and the fields supported in Flare (except "Tag"), see this link: **http://idpf.org/epub/20/spec/OPF_2.0.1_draft.htm#Section2.2**.

COVER IMAGE

Select the **Cover Page** tab and click ... to find and select an image file. This is used as the cover for your ebook.



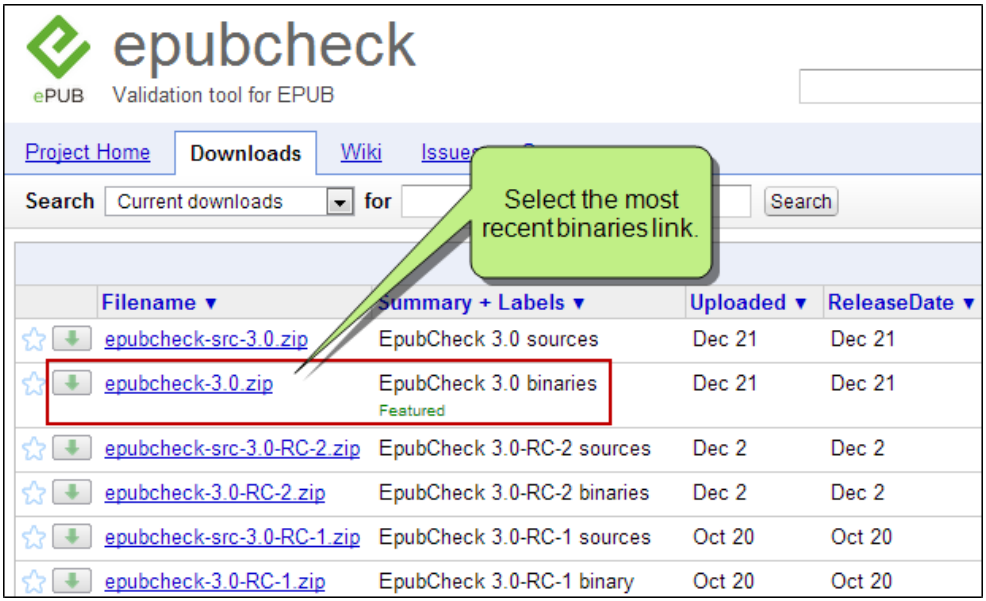
VALIDATE EPUB3 OUTPUT

Use these fields to validate EPUB output so that it will be acceptable to most online stores, such as amazon.com.

a. Select **Enable Epub 3 validation**.

b. Click the **Download EpubCheck** link. A website opens.

c. Click the link you want to download. Newer versions are typically at the top. Select the most recent link for binaries and follow the steps online to download it to your computer.



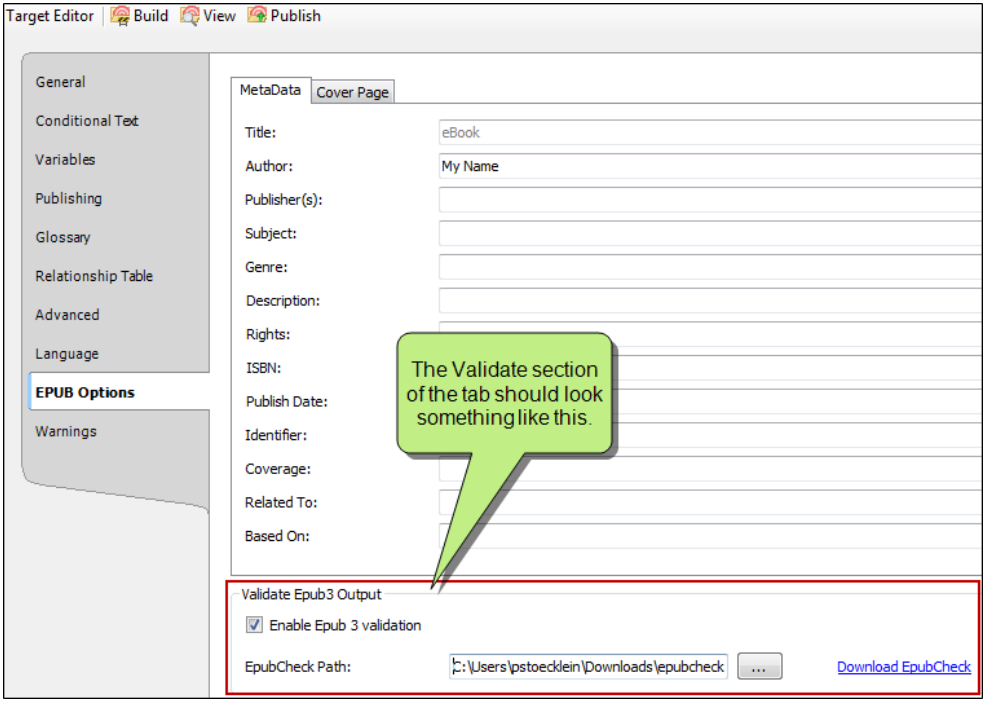
d. The file that you downloaded is a ZIP file, so you need to extract its contents.

e. Return to the EPUB Options tab in Flare.

f. Next to the **EpubCheck** field, click ...

g. Navigate to the extracted folder on your computer.

h. In the Open dialog, double-click the file with the **.jar** extension (e.g., **epubcheck-3.0.jar**).



i. Click to save your work.

When you are finished setting all EPUB options, you can generate the EPUB target. If validation fails, you should see related errors during the compilation process. You can then fix the issues and try to build again.

GENERATE MOBI OUTPUT

MOBI is the standard format used for the Amazon Kindle. You can use these fields settings to generate EPUB output that uses MOBI content.

a. Select **Generate MOBI file**.

b. Click the **Download KindleGen** link. A website opens.

c. Click the link you want and follow the steps online to download it to your computer.

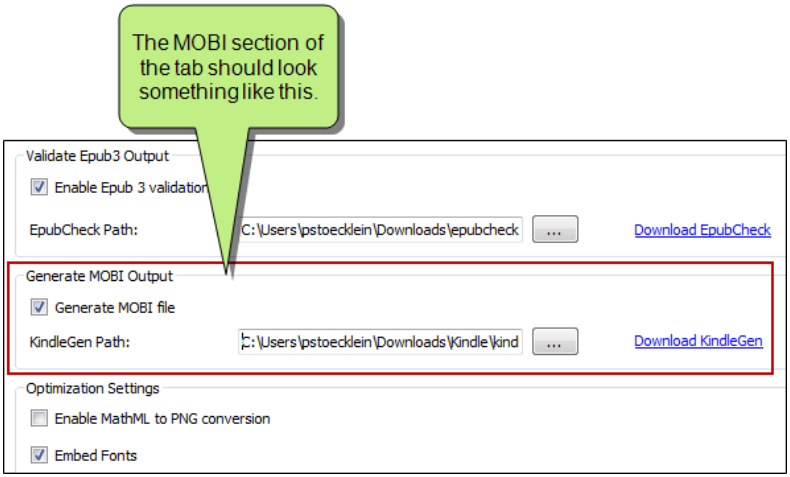
d. The file that you downloaded is a ZIP file, so you need to extract its contents.

e. Return to the EPUB Options tab in Flare.

f. Next to the **KindleGen Path** field, click ...

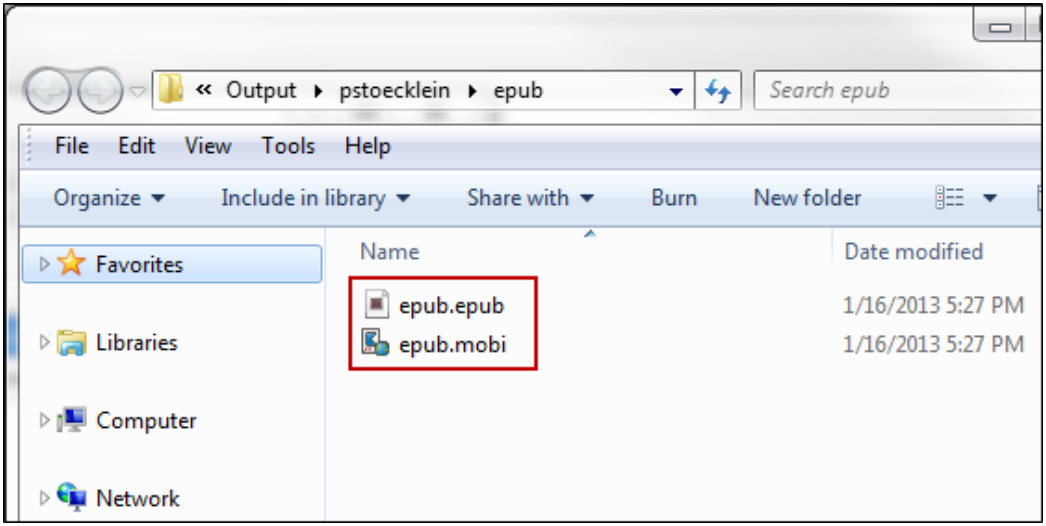
g. Navigate to the extracted folder on your computer.

h. In the Open dialog, double-click the file with the **kindlegen.exe** file.



i. Click to save your work.

When you are finished setting all EPUB options, you can generate the EPUB target. If you open the output folder, you will see that Flare has created a MOBI file in addition to an EPUB file. The MOBI file is all you need to view the ebook on a Kindle. The EPUB file can be used to view the ebook on other readers.



ENABLE MATHML TO PNG CONVERSION

If you use the Equation Editor to insert equations, you are in essence creating MathML markup. This type of content is supported for EPUB output.

If you need to produce output that can be displayed on older readers, you can select this option. All of your MathML content will be converted to PNG image files.

4. Click to save your work.

Note: For all supported image types (PNG, JPEG, JPG, GIF), the cover art in MOBI output has a minimum width or height of 300 pixels.

Enabling Feedback in Flare

This feature is supported in DotNet Help, Microsoft HTML Help, HTML5, WebHelp, and WebHelp Plus output.

After obtaining the necessary Feedback license(s), you need to enable the Feedback functionality in Flare. This is done on the Community tab in the targets.

HOW TO ENABLE FEEDBACK IN FLARE

1. From the Project Organizer, open the target.
2. Select the **Community** tab.
3. Click the **Enable Pulse/Feedback Server** check box.
4. In the **Feedback Server URL** field, enter the path to the server hosting the data. If you purchased the MadCap Feedback Service (as opposed to the MadCap Feedback Server), enter `madcap`, which maps to feedbackserver.madcapsoftware.com.

   Note: It is highly recommended that you purchase an SSL certificate and install it on your web server. SSL uses encryption for certain processes, which helps to ensure that the data transfer is secure. If you have an SSL certificate installed on your web server, make sure you begin the path to the server with https://.

5. Enter your license key using one of the following methods:
   - **If Using MadCap Feedback Service (i.e., MadCap does the hosting)** Click the **Enter Key** button to open the Feedback Service End User License Agreement dialog. After reviewing the agreement in the dialog, enter or copy the Feedback license that you have been issued into the **Feedback Service License Key** field. Then click **Accept**. This button is enabled only if "madcap" is entered in the Feedback Server URL field (i.e., you are using MadCap Software's hosted service, as opposed to your own server).
   - **If Using MadCap Feedback Server (i.e., you do the hosting)** Copy the Feedback license that you have been issued and paste it into the Feedback Service License Key field.
6. Click to save your work.

Associating Glossaries with Targets

This feature is supported in all outputs.

One of the final steps to creating a glossary is to associate it with a target. After you build the target, the glossary will be incorporated into the output.

HOW TO ASSOCIATE A GLOSSARY WITH A TARGET

1. Open the target from the Project Organizer.
2. In the Target Editor, click the **Glossary** tab.
3. (Optional) In the **Glossary Term Conversion** section, select one of the options if you want to make use of glossary term links:
   - **Do not convert terms** In the output, none of the glossary terms that appear in topics will be converted to hyperlinks, popups, expanding text, or footnotes in the output (even if a term was inserted into a topic as a glossary term link).
   - **Convert only marked terms** In the output, only glossary terms that were inserted into topics as glossary term links will be converted to hyperlinks, popups, expanding text, or footnotes in the output. Glossary terms that happen to exist in topics as normal text will not be converted.
   - **Convert first occurrence of term** In the output, only the first occurrence of a glossary term in a topic will be converted to a hyperlink, popup, expanding text, or footnote in the output (per your instructions). This includes terms that were inserted as glossary term links, as well as glossary terms that happen to exist in topics as normal text.
   - **Convert all occurrences of term** In the output, every occurrence of a glossary term in a topic will be converted to a hyperlink, popup, expanding text, or footnote in the output (per your instructions). This includes terms that were inserted as glossary term links, as well as glossary terms that happen to exist in topics as normal text.

   Note: The conversion of a glossary term to a link is case-sensitive. For example, if a term starts with an uppercase letter in the glossary, it will not be converted to a link in the output if the term is all lowercase in the topic.

4. Click the check box next to each glossary that you want to associate with the target. A check mark is added to the box.

5. (Optional for online output) In the **Glossary Master Page** section, select a master page to be associated with your glossary definitions.

   The master page will be used to display the glossary in the output.

   E X A M P L E

   You might use a glossary master page to add custom text at the bottom of the glossary. Or perhaps you want to add a watermark image to the background of the glossary. To do that, you could create a class of the <html> style, add a background image to that style class, and select that class in the master page file.

   Note: A glossary master page is not supported in HTML5 output.

6. Click to save your work.

Specifying Buttons for HTML Help Output

You can select the Help buttons that you would like to include in the output window (e.g., Hide, Forward, Back, Print, customized buttons).

HOW TO SPECIFY BUTTONS FOR HTML HELP OUTPUT

1. Open a Standard skin.
2. Select the **HTML Help Setup** tab.
3. In the **Html Help Buttons** section, select the buttons that you want to include in the output window.
   - **Hide** Opens and closes the navigation pane.
   - **Locate** Synchronizes the TOC with the current topic (identifying the location of the current topic in the TOC hierarchy). This is different from the option "Automatically Synchronize TOC" on the General tab. This option is a manual method (requires action from the user), where as the other option is automatic.
   - **Back** Opens the topic that was viewed previously.
   - **Forward** Opens the next topic in a previously viewed sequence.
   - **Stop** Prevents a page from downloading file information.
   - **Refresh** Updates the content of the topic being viewed.
   - **Home** Opens a designated home page (such as your company's website).
   - **Font** Opens a font selection popup. You can choose a typeface from among those listed in the left panel. The base size can be selected in the right panel. There are two separate font families used by the viewer: the normal, proportional-spaced font, and a fixed-pitch font for preformatted and "typewriter" text. The latter does not actually have to be fixed-pitch; however, preformatted columns will not line up unless this is so. Pressing **Apply Font** will set the currently selected font as the viewer's proportional font, and **Apply Fixed Font** will set the currently selected font as the viewers "fixed" font. In either case, the display will be redrawn using the new font. The new font selection will be updated in the .mozyrc file if it exists, so that subsequent windows will use the new font.
   - **Print** From the Contents tab, prints pages, headings, and subtopics, or the entire TOC. From the Index or Search tab, opens the Print dialog for printing the current topic.
   - **Next** Opens the next topic in the TOC. In order to include this button in the output, you must also make sure Binary TOC and Binary Index are selected (use the TOC Options and Index Options buttons). Please note that these buttons may not work correctly if the topic file names contain spaces.
   - **Previous** Opens the previous topic in the TOC. In order to include this button in the output, you must also make sure Binary TOC and Binary Index are selected (use the **TOC Options** and **Index Options** buttons). Please note that these buttons may not work correctly if the topic file names contain spaces.
   - **Options** Opens a menu that provides access to Home, Show, Back, Stop, Refresh, and Search Highlight On/Off.

- **Jump1** Opens a customized destination. You can use this to let users open a website (defined with a URL) or a Help topic (defined by its file name if included in the same CHM file). You can specify the destination and label for the button by selecting **Jump Button Options**.
- **Jump2** Opens a customized destination. You can use this to let users open a website (defined with a URL) or a Help topic (defined by its file name if included in the same CHM file). You can specify the destination and label for the button by selecting **Jump Button Options**.

4. If you want to include labels below each of the buttons in the output, select **Display button captions**.
5. If you selected the "Jump1" or "Jump2" button options, click **Jump Button Options** to specify the button text and destination. In the HTML Help Jump Button Options dialog, enter the appropriate destination URL (e.g., http://www.acme.com) in the **Jump1 URL** and/or **Jump2 URL** fields. Then enter the label for the button in the appropriate **Button text** field. When you are finished, click **OK** to close the dialog. (Make sure **Display button captions** is selected so that users can see the button text.)

> **Note:** You can also set the jump button options on the General tab in the Target Editor. The reason these button options are available in both the Skin Editor and Target Editor is this: If you want multiple targets to use one skin, with each target using the same destinations for the jump buttons, you should set the jump button URLs in the Skin Editor. If you want multiple targets to use the same skin, but you want them to use different URLs for the jump buttons, set the URLs in the Target Editor. If URLs are set in both the Skin Editor and the Target Editor, Flare uses the settings from the Skin Editor.

6. Click to save your work.

Displaying Merged Navigation in HTML Help Child Outputs

You can merge HTML Help so that the navigation (table of contents, index, search) for each CHM file is displayed, regardless of which CHM file you open (parent or child). This can be done in a couple of ways, depending on whether you have access to all of the Flare projects or to the CHM files only.

HOW TO DISPLAY MERGED NAVIGATION IN HTML HELP—ACCESS TO ALL FLARE PROJECTS

1. Open the target to be used as the parent CHM.
2. On the **Advanced** tab of the Target Editor, select **Display merged navigation in HTML Help**.
3. Click to save your work.
4. Follow the steps for merging based on HTML Help targets.
5. Generate the parent HTML Help target.

   The parent CHM file and all child CHM files are located in the same output folder, and when you open any of those CHM files, you can see the navigation for any of the other outputs.

HOW TO DISPLAY MERGED NAVIGATION IN HTML HELP—ACCESS TO CHM FILES ONLY

Use this method if you *do not* have access to all of the Flare projects, but you *do* know the names of the other CHM files that will be created. This method is more of a manual setup and is useful if you are part of a team where different authors have access to different Flare projects that need to be merged. This method also lets you add or remove child CHM files without needing to recompile the parent project.

1. The author for each project (parent *and* child) places a simple TXT file named *merge.txt* in the same folder where the Flare project file (.flprj) is located. This TXT file can be created with Notepad and should simply list the names of all CHM files to be included in the merged output (one on each line), with the name of the parent CHM appearing first.

   E X A M P L E

   MyParent.chm

   MyFirstChild.chm

   MySecondChild.chm

   MyThirdChild.chm

2. The author for each child project generates his output and provides the author of the parent project with the CHM file(s).

3. The author for the parent project brings the child CHM files into the parent project and follows the steps for merging based on CHM files.
4. The author for the parent project generates the main HTML Help target.

   The parent CHM file and all child CHM files are located in the same output folder, and when you open any of those CHM files, you can see the navigation for any of the other outputs.

Patching Stylesheets and Image Links for HTML Help

If you generate Microsoft HTML Help output, some topics may not look as intended when they are printed from the CHM file, due to stylesheet-related problems. You can use this feature to "patch" those problems, ensuring the printed topics will look as intended.

Why would you not use this option? The only reason not to use this option is when you plan to rename the generated CHM file. If this option is enabled and you rename the CHM file, styles in the output are broken. This happens because, when the option is enabled, the file name of the CHM is hardcoded into the CHM itself.

HOW TO PATCH STYLESHEETS AND IMAGE LINKS FOR HTML HELP

1. Open a target.
2. On the **Advanced** tab of the Target Editor, select **Patch stylesheet and image links for HTML Help printing**.
3. Click to save your work.

Setting Up an HTML5 Target

The following steps show you how to set up your HTML5 target for server-based output.

HOW TO SET UP THE HTML5 TARGET

1. In Flare, open the HTML5 target.The Target Editor opens.
2. Select the **Publishing** tab.
3. In the **Server-based Output** section, click the check box.
4. In the **Indexing Service Catalog Name** field, type the catalog that you are using for the output. In most cases, this will be `Web`, which is the default value. However, if you or someone in your company (e.g., network administrator) creates a custom catalog, you need to enter that name in the field.

   What is a catalog? Microsoft Indexing Service stores all of its index information in catalogs. A catalog comprises index information and stored properties for a particular group of file system directories. When Indexing Service is installed with Windows XP, it automatically builds a catalog, called the System catalog, listing contents of all permanently attached disk drives. The System catalog contains an index for all documents except certain system and temporary files. If Internet Information Services (IIS) is installed, the Indexing Service also creates a web catalog, which contains an index of IIS, the default virtual server of the World Wide Web.

   Note: This step is necessary only for Windows XP and Windows Server 2003.

5. Click to save your work.
6. Generate the target.
7. Publish the output to any location on the web server. If you want to use the default location that was created after you installed IIS, you can publish the output to C:\Inetpub\wwwroot. If you do this, you can use the "Default Web Site" folder that you will encounter when performing the next set of steps (i.e., configuring IIS on the production server).

   If you are working on your local machine for testing purposes, you do not need to publish the output. You can simply view your WebHelp Plus output. When you view HTML5 output on your local computer, you need to create a special folder called "MCPreview" within your "C:\Inetpub\wwwroot" folder. Place a copy of your HTML5 output files in it. This enables you to test the advanced features of HTML5 on your local machine. (For Windows Vista users testing locally, the physical location is C:\MadCap\[Name of Project]\[Name of Main Entry File].htm.)

Embedding Images in Microsoft Word Output

An option on the Advanced tab of the Target Editor lets you ensure that images are embedded in Microsoft Word output.

HOW TO EMBED IMAGES IN MICROSOFT WORD OUTPUT

1. Open the target being used for Word output.
2. Select the **Advanced** tab.
3. Click the check box next to **Embed images in output** so that it contains a check mark.
4. Click to save your work.

Generating Web-Safe Images

This feature is supported in all online outputs (DotNet Help, Eclipse Help, Microsoft HTML Help, HTML5, WebHelp, WebHelp AIR, WebHelp Mobile, and WebHelp Plus).

If you have used non–web-safe image formats (e.g., WMF, EMF, BMP, TIF, TIFF, XPS, EXPS) in your project and want those images to be converted to web-safe formats (e.g., GIF, JPEG, PNG) when you generate online output—e.g., DotNet Help, Eclipse Help, Microsoft HTML Help, HTML5, WebHelp, WebHelp AIR, WebHelp Mobile, WebHelp Plus—you can use an option on the Advanced tab of the Target Editor. For print-based output types (Adobe FrameMaker, Adobe PDF, Microsoft Word, Microsoft XPS, XHTML), the original image file formats will be used when you generate output.

HOW TO GENERATE WEB-SAFE IMAGES

1. Open a target.
2. Select the **Advanced** tab.
3. Select **Generate "web-safe" images**.
4. From the drop-down list, select the image format that you want to use.
5. Click to save your work.

Note: If you are using this option, images that link to external locations (such as websites), rather than to images within the project, are not affected.

Note: Vector images, such as SVG, are preserved in the output, unless you use this option to generate web-safe image formats.

Disabling Auto-Sync of Import Files

This feature is supported in all outputs.

Let's say you have imported Microsoft Word, Adobe FrameMaker, or Flare files from another project, and when doing so, you selected the "Easy Sync" option to automatically re-import the files when you generate output. If you want to override that Easy Sync setting for a specific target, you can do so by using the steps below. Therefore, if you want to re-import any of the files, you will need to do so manually.

You might decide to use this option, for example, if you are testing the generation of output and do not want to wait the extra time for the files to be imported. After you finish your testing, you can deselect this option to return to the automatic imports of the files.

HOW TO DISABLE AUTO-SYNC OF IMPORT FILES

1. Open the target.
2. In the Target Editor, select the **General** tab.
3. Select the check box labeled **Disable auto-sync of all import files**.
4. Click to save your work.

Excluding Index Entries From Search

This feature is supported in DotNet Help and in all WebHelp outputs (HTML5, WebHelp, WebHelp AIR, WebHelp Mobile, and WebHelp Plus).

If you insert index markers in your project, those markers by default are included in searches that users perform in your output. If you want to exclude index entries from searches, you can do so.

HOW TO EXCLUDE INDEX ENTRIES FROM SEARCH

1. Open the target.
2. Select the **Performance** tab.
3. In the **Search Database** section, click **Exclude Index Entries from Search**.
4. Click to save your work.

Selecting a Language

You can select a language at any of the following levels:

- Project
- Target
- Topic
- Content within a topic or snippet

The broader the level where the language is set, the lower precedence it has in case of conflicts. In other words, the content level has precedence over everything. The topic level has precedence over the target and project levels. And the target level has precedence over the project level.

HOW TO SELECT A LANGUAGE FOR A SPECIFIC TARGET

1. Open the target.
2. In the Target Editor, select the **Language** tab.
3. Select the language from the list (those in bold font are supported in spell check).
4. If you selected a right-to-left (RTL) language, you may see the following options, which are enabled by default for RTL languages. Leave them selected to automatically invert the following: (1) language-related style rules locally, (2) language-related style rules in the stylesheet, (3) image callouts from MadCap Capture, and (4) page layout settings.
   - **Invert left/right local styles**
   - **Invert left/right stylesheet rules**
   - **Invert Capture captions**
   - **Invert Page Layouts**

   The options that are seen depend on which output type you are using.

   - **PDF/XPS/XHTML/Word** All four options are shown.
   - **HTML5/HTML Help/EPUB/Eclipse Help/All WebHelps** Local styles, CSS styles, and image callout options are shown.
   - **DITA/DotNet Help/FrameMaker** No options are shown (FrameMaker and DotNet Help do not support RTL languages).

   These options can be quite useful after you receive the project back from a translator. When you generate the output, the inversion of the styles and page layouts takes place.

EXAMPLE

Let's say you have a project in English (a left-to-right language), and you need to have it translated into Arabic (a right-to-left language).

For your <p> style, suppose you have a left margin of 5 px and a right margin of 30 px.

Those settings work fine for your targets using the LTR language. But for the RTL language, you need the settings to be flipped so that the left margin is 30 px and the right margin is 5 px.

So after you receive the project back from the translator, you make sure the RTL language in Flare is set at either the project or target level.

As a result, the inversion options become selected automatically, and when you generate the output, paragraphs will have a left margin of 30 px and a right margin of 5 px.

Note: The option to invert page layouts controls every aspect of the page layout file (e.g., it inverts not only frames and content within them, but also styles applied to frame content). The other invert options have no effect on page layouts.

Note: If your selected language is LTR, these options cannot be accessed in the Target Editor.

5. Click to save your work.

Adding Mark of the Web

This feature is supported in Eclipse Help, HTML5, WebHelp, and WebHelp Plus output.

According to Microsoft:

> *"The Mark of the Web (MOTW) is a feature of Windows Internet Explorer that enhances security. MOTW enables Internet Explorer to force webpages to run in the security zone of the location the page was saved from—as long as that security zone is more restrictive than the Local Machine zone—instead of the Local Machine zone. When you are developing webpages, the MOTW enables you to test your HTML documents in the security zone where you intend the pages to run. Adding the MOTW to your webpages also enables you to fully test their compatibility with users' security settings."*

So MOTW is a comment added to the HTML markup for a web page. When users open a web page from their local machine, Internet Explorer references this comment to determine the security zone in which it should run the page.

Therefore, if you *do not* have MOTW enabled in Flare, the following message displays when HTML5, WebHelp, or WebHelp Plus output is opened locally on Internet Explorer.

If you *do* have MOTW enabled in Flare, this security message is not displayed.

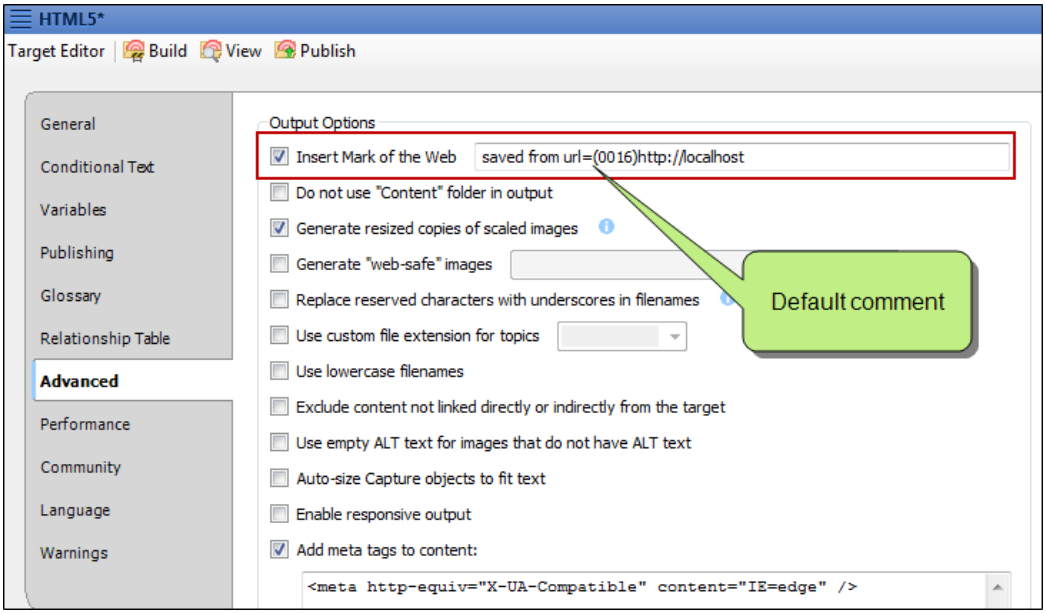
Without MOTW, you would have to click on this message and select **Allow Blocked Content** to view the output locally on Internet Explorer. By adding MOTW, you can avoid this situation.

However, please note that if you have external links in your project to non-HTML files (e.g., DOC, PDF, JPG, GIF), those links will not work with MOTW enabled. Therefore, MOTW should not be used unless the links in your project are limited to HTML documents only.

HOW TO ADD MARK OF THE WEB

1. Open a target.
2. On the **Advanced** tab of the Target Editor, select **Insert Mark of the Web**.

   The default comment in this field is a generic one that specifies the Local Intranet zone and is used in situations where the domain is not yet known. You can replace it with another comment if necessary.



   If you plan to use MOTW, the text field should contain some content. You should either keep the default content or replace it with your own. If you remove any content from the field, MOTW cannot be used.

3. Click to save your work.

Note: With Mark of the web enabled in your target, some links to external files may not work properly in some versions of Internet Explorer when pages are viewed locally (file://). This is not an issue if pages are viewed online (http://). For more information about MOTW, see Microsoft's MSDN website.

Associating Master Pages with Targets

This feature is supported in Adobe FrameMaker, Microsoft Word, and in all online outputs (DotNet Help, Eclipse Help, Microsoft HTML Help, HTML5, WebHelp, WebHelp AIR, WebHelp Mobile, and WebHelp Plus).

If you want a master page to be applied to all topics in the output, you would associate that master page with the target that you are building. This is useful, for example, if you want to create breadcrumbs, a mini-TOC, header content, or footer content for your online outputs.

HOW TO ASSOCIATE A MASTER PAGE WITH A TARGET

1. Open the target.
2. In the Target Editor, click the **Advanced** tab.
3. Click the drop-down arrow in the **Master Page** field, and select the master page that you want to associate with the target.
4. Click to save your work.

Disabling the Use of Master Pages

This feature is supported in Adobe FrameMaker and Microsoft Word output.

Let's say you have used master pages in earlier versions of Flare when creating Word or FrameMaker output, but now you decide to use the newer page layouts instead. You can manually remove all links to master pages in your target and table of contents. However, another alternative is to automatically disable the use of master pages in your project. By using this feature, Flare will ignore all links to master pages when you generate the Word or FrameMaker target. It will instead use links that you provide to any page layouts.

HOW TO DISABLE THE USE OF MASTER PAGES

1. Open the target.
2. In the Target Editor, select the **Advanced** tab.
3. Click the check box next to **Disable Master Pages for FrameMaker output** or **Disable Master Pages for MS Word output**.
4. Click to save your work.

Adding Custom Meta Tags

This feature is supported in all online outputs (DotNet Help, Eclipse Help, Microsoft HTML Help, HTML5, WebHelp, WebHelp AIR, WebHelp Mobile, and WebHelp Plus).

For online outputs, you can add custom meta tags in the Advanced tab of the Target Editor. These are added between the <head> and </head> tags in your content when you generate output.

There is a default meta tag provided. You can keep this tag, remove it, and/or add other valid tags.

E X A M P L E

Let's say you have embedded HTML5 movies in your topics, and you are generating DotNet Help. However, DotNet Help uses Internet Explorer 7, which does not support HTML5 movies. Therefore, you use the default meta tag in the DotNet Help target to ensure that the latest version of Internet Explorer is used, therefore allowing the movies to play.

E X A M P L E

You might want to enhance search engine optimization for your output, so you add the <meta name="googlebot"/> tag to the target.

HOW TO ADD CUSTOM META TAGS

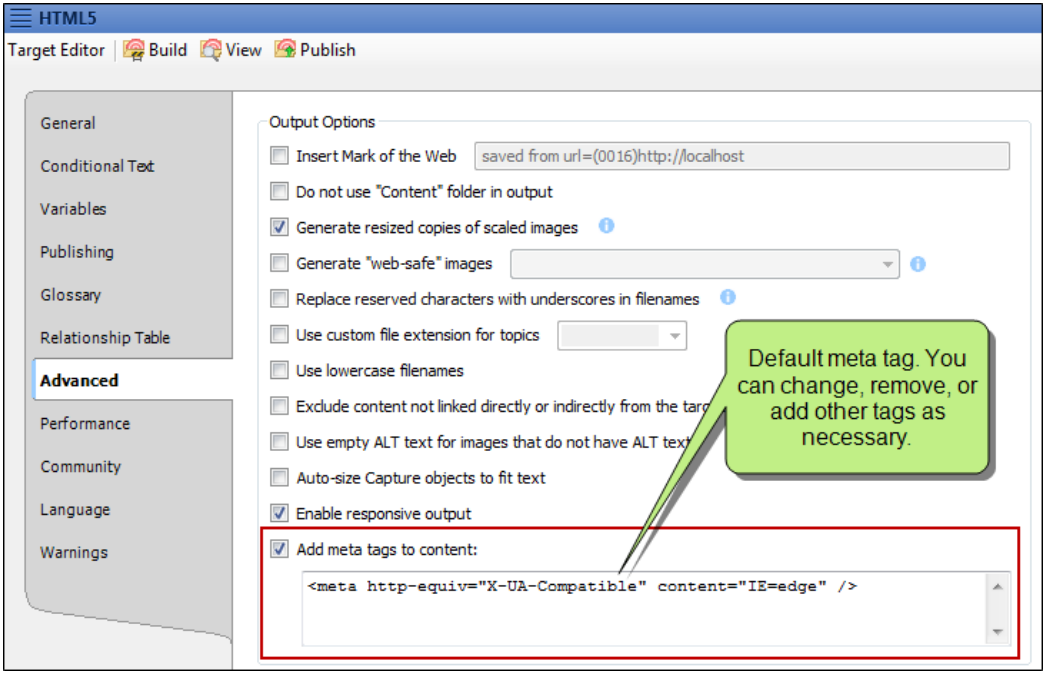
1. Open a target.
2. On the **Advanced** tab of the Target Editor, select **Add meta tags to content**.
3. In the area below, you can add any valid meta tags. You can also keep the default tag.

   DEFAULT TAG FOR MOST OUTPUT FORMATS

   For most output formats, the default tag is as follows:

   <meta http-equiv="X-UA-Compatible" content="IE=edge" />

   This tag tells Internet Explorer to render with the latest version.

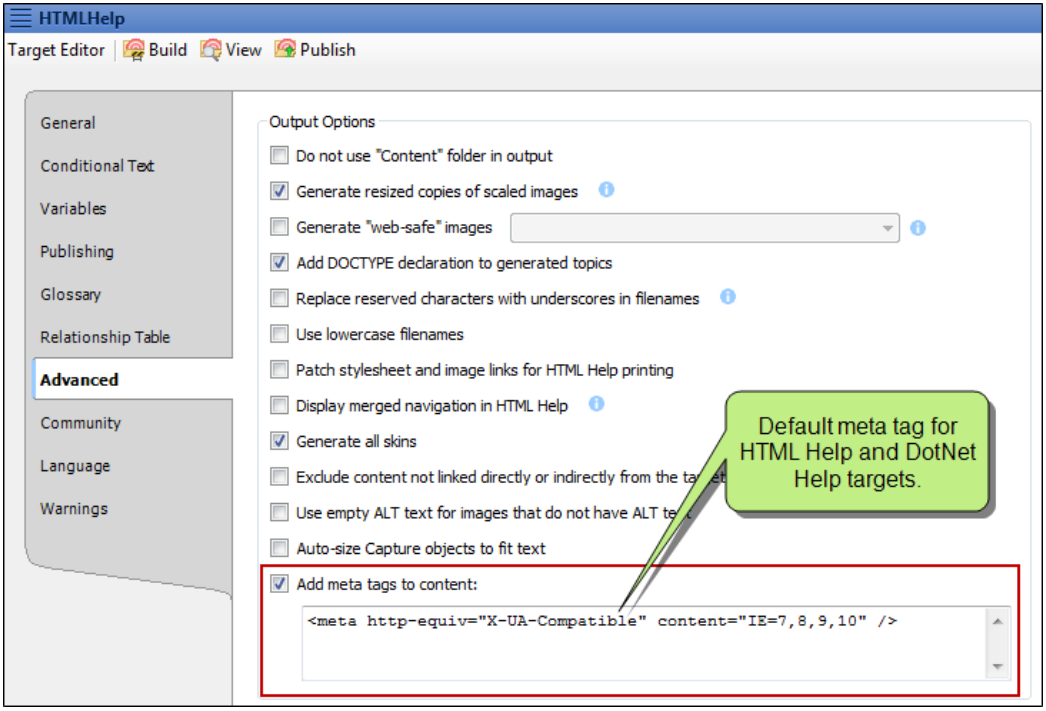


DEFAULT TAG FOR DOTNET HELP AND HTML HELP

For DotNet Help and Microsoft HTML Help targets, the default tag looks like this:

<meta http-equiv="X-UA-Compatible" content="IE=7,8,9,10" />

That's because IE=Edge can cause rendering issues on some systems.



4. Click to save your work.

Specifying the Output File and Folder in a Target

When you build the output for a target in your project, Flare sends the output files to the Output folder where your project is located. It also uses a default name for the main entry file to the output.

However, you can specify a different location and main entry file name for a target's output files.

EXAMPLE

Let's say you have a project called "FictionSoftPro" and have stored it in your root C:\ directory. If you create a project based on the "Empty" template and then build a DotNet Help target named "MyDotNetHelp,"Flare places the output files in C:\FictionSoftPro\Output\MyName\MyDotNetHelp, with the default main entry file named "Manual.mchelp." If you build the output for an Microsoft HTML Help target, Flare places the output files in C:\FictionSoftPro\Output\MyName\MyHtmlHelp, with the main entry file (actually, the only file) named "Manual.chm."

HOW TO SPECIFY THE OUTPUT FILE AND FOLDER IN A TARGET

1. Open the target.
2. On the **General** tab of the Target Editor, do one or both of the following.
   - To specify the main entry file name, type the desired name in the **Output File** field.

     For Word output, the default file type is XML. However, you can enter one of the following extensions at the end of the file name to use that file type: .doc, .docx, .pdf, .xps.

     For FrameMaker output, the default type is BOOK and/or FM. However, you can enter .pdf in the field to use that file type.

     Note: This field affects the name of the .mcbook file (used for viewing XHTML output in the MadCap Book Viewer). It also affects the name of the .book file for FrameMaker output. If you are creating multiple output files using chapter breaks, you can specify the name of each chapter file in the Printed Output tab of the TOC Properties dialog.

     Note: The Output File field is not available for Eclipse Help targets.

   - To specify the location of the output files, click ... in the **Output Folder** field. Then navigate to and select the location in your computer where you want to place the files.
3. Click to save your work.

Replacing Characters and Spaces with Underscores

This feature is supported in all online outputs (DotNet Help, Eclipse Help, Microsoft HTML Help, HTML5, WebHelp, WebHelp AIR, WebHelp Mobile, and WebHelp Plus).

If you have spaces or unusual characters in your file names or folders, you can convert these spaces and characters to underscores in the output. This feature is primarily useful for individuals working in a UNIX environment. In addition to spaces, the characters that are converted to underscores include: ()&;,!'.

HOW TO REPLACE CHARACTERS AND SPACES WITH UNDERSCORES

1. Open a target.
2. On the **Advanced** tab of the Target Editor, select **Replace reserved characters with underscores in file-names**.
3. Click to save your work.

Note: This feature applies to the files and folders within the root output folder, but not to the root output folder itself.

Note: You should use caution with this feature because some international characters are also converted to underscores. This includes characters such as ç, ß, and ñ. All varieties of non-English vowels are affected, as are all Asian characters.

Using Custom File Extensions

This feature is supported in DotNet Help, Eclipse Help, HTML5, WebHelp, WebHelp AIR, WebHelp Mobile, and WebHelp Plus output.

You can use specific file extensions for topics for WebHelp output types. If you do not use this feature, the output topic files will have an .htm extension. The most common alternative extensions are .html and .aspx.

HOW TO USE CUSTOM FILE EXTENSIONS

1. Open a target.
2. Select the **Advanced** tab.
3. Select **Use custom file extensions for topics**.
4. In the field below, enter the extension that you want to use. Do not type the period, but rather the characters only (e.g., aspx).
5. Click to save your work.

Excluding Content not Linked Directly or Indirectly from Targets

This feature is supported in all online outputs (DotNet Help, Eclipse Help, Microsoft HTML Help, HTML5, WebHelp, WebHelp AIR, WebHelp Mobile, and WebHelp Plus).

One way to completely exclude content from online output is to place condition tags on the topic files and include or exclude them from the targets. Another method—discussed here—is to use an option in the Target Editor to include only content files that are directly or indirectly referenced from the target. This means that if the target is using particular files such as a table of contents (TOCs), master page, and so on, other files linked directly or indirectly from them will be part of the output.

E X A M P L E

Let's say your target points to a TOC that includes Topic A, Topic B, and Topic C. If you generate an online output and use this option in the Target Editor, all three of those topics will be included in the output. In addition, Topic D will be included in the output. Although Topic D is not found in the TOC, a cross-reference to it is found in Topic B; therefore, it is linked to the TOC (and the target) indirectly. However, Topic E is not included in the output, because it is not found in the TOC and none of the topics to be included in the output contain a link to it.

E X A M P L E

Let's say you have four topics in your project: Topic A, Topic B, Topic C, and Topic D.

Topic A, Topic B, and Topic C have been included in your TOC. Topic D has not been included, and it does not contain any links to or from any other topics.

Furthermore, let's say you want to use the "list-of" proxy feature to automatically generate a list of all H2 headings in the project. All four of the topics have H2 headings in them, and you've inserted the list-of proxy somewhere in Topic A.

So what happens when you generate the output?

If the "Exclude..." target option is not enabled in the target, all four topics will be included in the output. In addition, the generated list of H2 headings will show heading text from all four of those topics.

However, if the "Exclude..." target option is enabled, only Topic A, Topic B, and Topic C will be included in the output. This also means that the generated list of H2 headings will show heading text only from those three topics. The H2 heading text from Topic D will not be included.

E X A M P L E

Let's say your primary target explicitly links to the following.

- A TOC
- A browse sequence
- A stylesheet
- One or more glossary files
- A master page
- A startup topic

A target can also link explicitly to one or more relationship tables and one alias file. However, these files are not used to determine whether files should be included when this "Exclude…" target option is enabled.

- The TOC and the browse sequence can link to topics, other TOCs, or other browse sequences. These files would then be included. You could even place a link to multimedia from a TOC or browse sequence if you wanted (although this cannot be done in the Flare user interface). It will work and those explicitly linked resources would also be included.
- A stylesheet may link to image resources. These resources are included.
- A glossary file may link explicitly to a topic if a term uses a topic as its definition. These topics are included.
- A master page may link to a topic or resource (image/multimedia). These linked items would be included.
- The startup topic is, of course, included.
- Any included topic can explicitly link to additional topics via hyperlinks, cross-references, image maps, or related topics controls (*not* concept links or keyword links).

Using this option can help prevent bloated output by excluding unused files.

E X A M P L E

Let's say you have 1000 images in your project, but only 500 of them are used in the Microsoft HTML Help output you want to generate. By using this feature, only those 500 images are included in the output, thus keeping the output file size down to a minimum.

HOW TO EXCLUDE CONTENT NOT LINKED DIRECTLY OR INDIRECTLY FROM A TARGET

1. Open a target used for online output.
2. On the **Advanced** tab of the Target Editor, select **Exclude content not linked directly or indirectly from the target**.
3. Click to save your work.

Converting Output File Names and Folders to Lowercase

You can convert online output file names and folders to all lowercase letters, even if the corresponding file names and folders in the project are not all lowercase. This option is useful, for example, if you are working in a UNIX environment, which is case-sensitive.

HOW TO CONVERT OUTPUT FILE NAMES AND FOLDERS TO LOWERCASE

1. Open a target.
2. On the **Advanced** tab of the Target Editor, select **Use lowercase filenames**.
3. Click to save your work.

Note: This feature applies to the files and folders within the root output folder, but not to the root output folder itself.

Omitting the Content Folder from Output

This feature is supported in DotNet Help, Eclipse Help, Microsoft HTML Help, HTML5, WebHelp, WebHelp AIR, WebHelp Mobile, and WebHelp Plus output.

The Content subfolder in a project is normally used to hold all of your content files. If you do not want this subfolder to be created when you generate online output, you can omit it. When you use this option and build output, the content files will be placed at the root of the output folder instead, rather than within the Content subfolder.

HOW TO OMIT THE CONTENT FOLDER FROM OUTPUT

1. Open a target used for online output.
2. On the **Advanced** tab of the Target Editor, select **Do not use "Content" folder in output**.
3. Click to save your work.

Changing the Output Type for a Target

Each target in your project is based on one of the output types that is available in Flare (Adobe FrameMaker, Adobe PDF, DITA, DotNet Help, Eclipse Help, EPUB, Microsoft HTML Help, HTML5, Microsoft Word, Microsoft XPS, WebHelp, WebHelp AIR, WebHelp Mobile, WebHelp Plus). If you want a particular target to use a different output type than is currently specified, use the following steps. You can also add an internal comment to describe the output.

HOW TO CHANGE THE OUTPUT TYPE FOR A TARGET

1. From the Project Organizer, open the target.
2. Click the **General** tab.
3. From the **Output Type** field make a selection.
4. In the message that opens click **Yes**.
5. (Optional) In the **Comment** field, you can enter an internal comment that describes the output you are generating from the target.
6. Click to save your work.

Note: For help in deciding which output type(s) you should use, see "Determining the Output Type" on page 13.

Associating Master Page Layouts with Targets

This feature is supported in all print outputs (Adobe FrameMaker, Adobe PDF, Microsoft Word, Microsoft XPS, and XHTML).

After you create a page layout and configure its frames and settings as necessary, you need to associate the page layout with the appropriate content. In most cases, you will probably want to associate different page layouts with various entries in your outline TOC (so that different page layouts can be used for different parts or chapters in a manual). Otherwise, you would associate a single "master" page layout with an entire target or project; in that case, the same page layout will be applied to all topics in that target or project. You can associate a page layout with an outline TOC entry with or without creating a chapter break at the same time.

The following steps show how to associate a "master" page layout at the target level.

HOW TO ASSOCIATE A MASTER PAGE LAYOUT WITH A TARGET

1. Open the target. The Target Editor opens.
2. Select the **General** tab.
3. Click in the **Master Page Layout** field, and from the drop-down select the page layout that you want to use for all topics in the target.
4. Click to save your work.

Note: There is an order of precedence with the three levels where you can specify a page layout. Specific entries in an outline TOC have the highest precedence, followed by page layouts at the target level, followed by page layouts at the project level. So if you've got a page layout set at the project or target level and it doesn't seem to be working the way you think it should, you might check to see if you have any page layouts set on specific TOC entries that are overriding the target- or project-level setting.

Specifying PDF Options

If you are sending output to Adobe PDF, you can access PDF options in the Target Editor. These options let you specify the way that images, document properties, PDF tagging, the initial view, and security are handled in the output.

HOW TO SPECIFY PDF OPTIONS

1. Open a target that is using the PDF format. If necessary, you can change the output type for a target. See "Changing the Output Type for a Target" on page 123.
2. Select the **PDF Options** tab.
3. Complete the options on the tab as necessary.

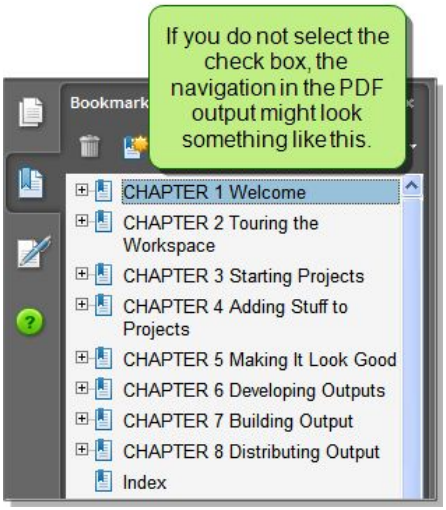
   IMAGE COMPRESSION

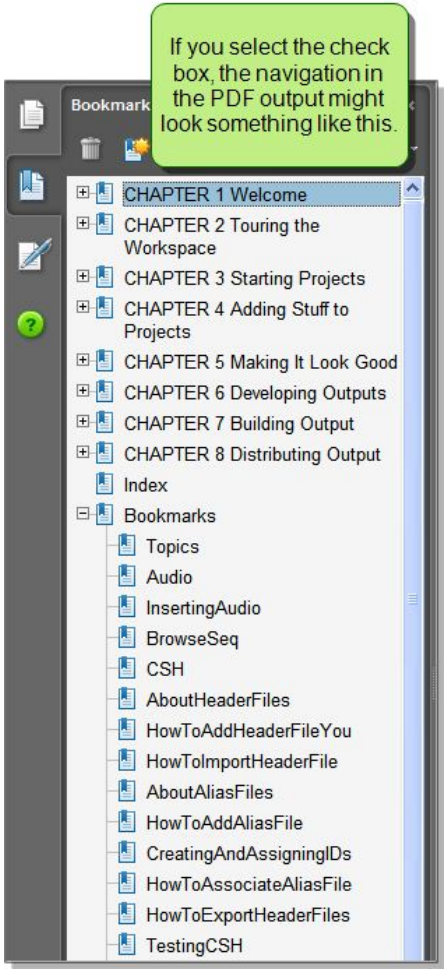
   - **Downsample images above** Select this if you want to downsample images and then enter the number of pixels per inch. Downsampling means to reduce the number of pixels in an image, which can significantly reduce the size of the PDF.
   - **Compression** Select an option for the compression.
     - **Automatic** This uses Flare's internal lossless compression algorithm on all images. Lossless compression allows you to reduce file size without sacrificing image quality. JPEGs remain as they are.
     - **JPEG** This converts all images to JPEG with some compression.
   - **Quality** If you select the "JPEG" option, you can also choose an option from this field. Maximum uses the least amount of compression and results in a higher file size.

   DOCUMENT PROPERTIES

   - **Title** Enter the title for the PDF document. This information is included in the final PDF output (e.g., shown in the file properties). The same is true for the Author, Subject, and Keywords fields.
   - **Author** Enter the name of the author.
   - **Subject** Enter the subject of the document.
   - **Keywords** Enter any keywords for the document.

» **Include non-TOC bookmarks in the bookmarks pane** If you have inserted bookmarks randomly in some of your topics, by default these will not appear as links in the output navigation pane. If you want these bookmarks to be shown in the navigation pane (under your table of contents links) you can click this check box.





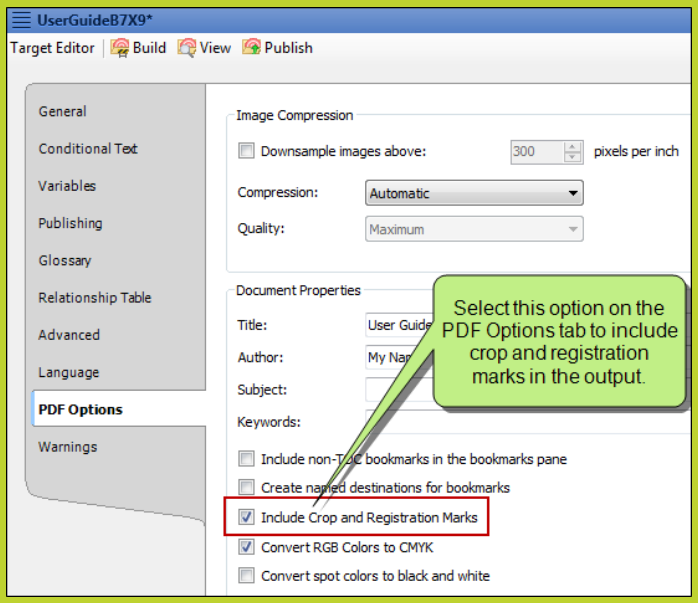
» **Create named destinations for bookmarks** Select this check box if you want to automatically create named destinations everywhere a bookmark exists. This lets you create links from a file to a specific location in a PDF. You can also insert named destinations manually.

» **Include Crop and Registration Marks** Select this check box if you want to include crop and registration marks in the output. These are elements that are often used when documents are sent to a professional printer.

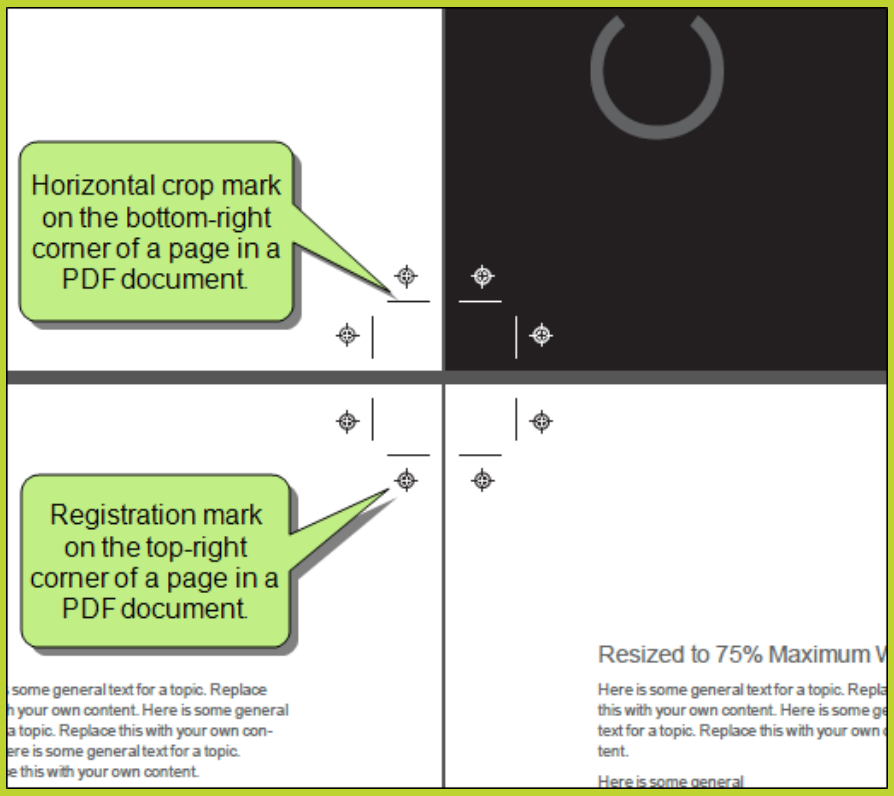
  » **Crop marks** These are thin vertical and horizontal lines that show the edges of the page, where it will be trimmed by the printer. The location of crop marks are determined by the page size you specify.

  » **Registration marks** These are small cross-hair target icons shown next to crop marks. Each color in the final output is printed on its own layer. Therefore, in order to make sure the colors are matched up accurately, printers use registration marks to align plates on the press.

EXAMPLE



Here is PDF output showing crop and registration marks.



» **Convert RGB Colors to CMYK** Select this check box if you have used only RGB colors in your content and want all of those occurrences to be converted to the CMYK model in the output. CYMK is an acronym for the four primary colors used in printing (Cyan, Magenta, Yellow, and Key—or black). Whereas CMYK is often the preferred model for print-based output, the RGB (Red, Green, Blue) model is primarily intended for electronic content.

» **Convert spot colors to black and white** Select this option to convert spot colors to black and white.

Spot colors include things such as font color, background color, and so on (basically, everything except images). This is a quick way to produce a PDF that does not contain color. The alternative is to make sure that you do not have any color settings locally in topics, in your stylesheet (or medium), in page layouts, etc. For example, you might create some PDFs in full color, but you might need to send one to a professional printer to be processed in black and white. So for that target, you select the option to convert spot colors to black and white.

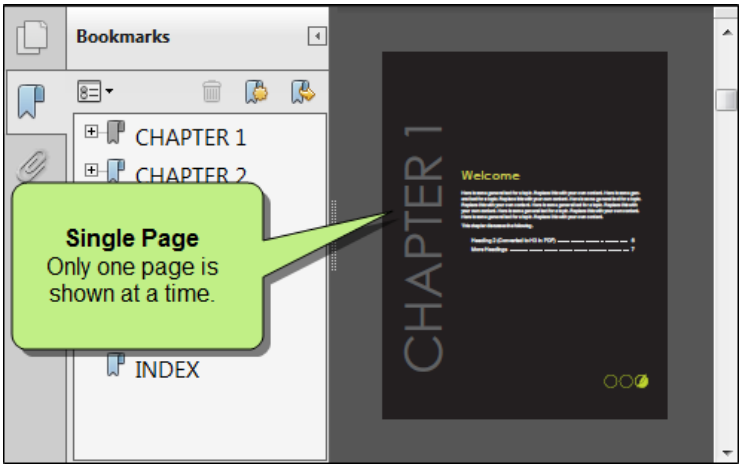
Note: If you also need the PDF to show images in gray scale, you have a couple of options. First, you can insert one image in color and another in gray scale, then apply conditions to them. Second, you can create and insert just one image, but single-source it so that online outputs show it in color but print outputs show it in gray scale.

PDF TAGGING

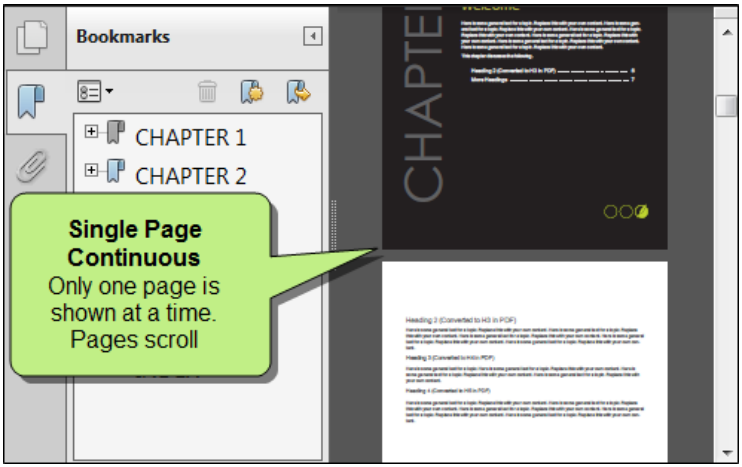
- **Generate tagged PDF** Select this to generate a tagged PDF. This gives the file a structure similar to that of the source XHTML documents. This structure is necessary for certain accessibility applications, including screen readers. For more information about viewing the tag structure in Adobe Acrobat Pro, please refer to the documentation provided with that application.

INITIAL VIEW

- **Magnification** Select the default size for displaying the PDF output when an end user opens it (e.g., actual size, fit page, fit width, a specific percentage).
- **Navigation** Select how you want the output to be displayed by default when it comes to navigation features (Page Only, Bookmarks Panel, and Page).
- **Page layout** You can select any of the following options to set the page layout that the reader initially sees when opening the PDF.
  - **(default) Single Page** This displays one page at a time, with no portion on other pages visible. When you slide the scroll bar, the current page does not scroll; instead, the next page appears quickly.

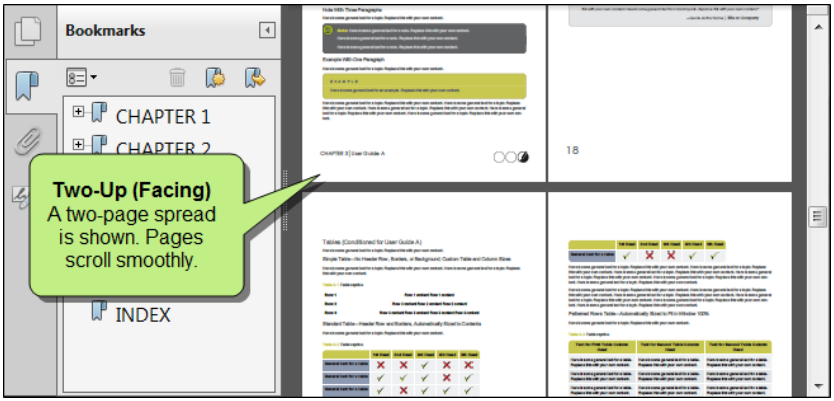


» **Single Page Continuous** This displays pages in a continuous vertical column that is one page wide. When you slide the scroll bar, the current page scrolls until the next page smoothly comes into view.

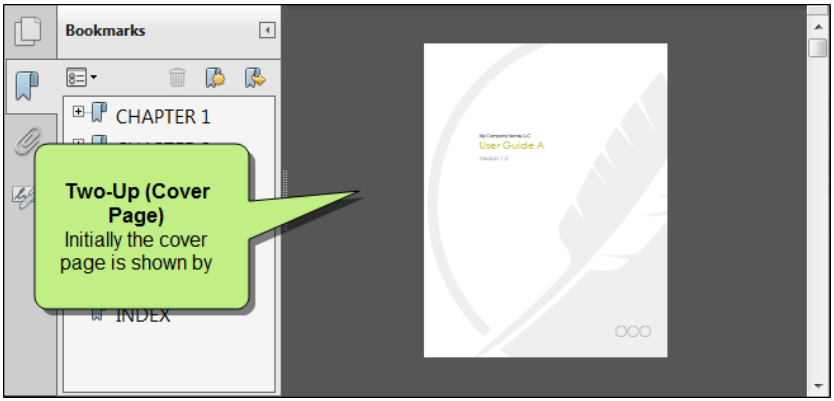


» **Two-Up (Facing)** This displays each two-page spread with no portion of other pages visible. When you slide the scroll bar, the current pages do not scroll; instead, the next pages appear quickly.

» **Two-Up Continuous (Facing)** This displays facing pages side by side in a continuous vertical column. When you slide the scroll bar, the current pages scroll until the next pages smoothly come into view.



» **Two-Up (Cover Page)** This displays the first page by itself. After that it displays each two-page spread with no portion of other pages visible. When you slide the scroll bar, the current pages do not scroll; instead, the next pages appear quickly.



- **Two-Up Continuous (Cover Page)** This displays the first page by itself. After that it displays each two-page spread with no portion of other pages visible. When you slide the scroll bar, the current pages scroll until the next pages smoothly come into view.

- **Title bar** You can select one of the following to determine the text shown in the title bar of the PDF output.
  - **File Name** This is the text specified in the Output File field on the General tab of the Target Editor.
  - **Document Title** This is the text specified in the Title field on the PDF Options tab of the Target Editor.
- **Collapsed bookmarks** Select this if you want the bookmark folders to be collapsed initially in the bookmark panel of the output.
- **Open in full screen mode** Select this if you want the output to be displayed initially in full screen mode, taking over the reader's entire screen.

SECURITY

- **Require password to open** Select this if you want users to be required to enter a password in order to open the document. Then type the password you want to use.

  Specify the amount of security for the document.

  - **40-bit** Not as secure, but can be viewed with Adobe Reader 3.0 and higher
  - **128-bit** More secure, but can be read only with Adobe Reader 5.0 and higher

» **Restrict permissions** Select this if you want to limit what readers of the document can do unless they have the permissions password. Then type the password you want to use. When you set only a permissions password (also known as a "master password"), recipients don't need a password to open the document. However, they must type the permissions password to set or change the restricted features. This option works with Adobe Acrobat Reader 9.0 and higher.

In the **Open Permissions** section, specify the actions that you want to restrict.

> **Note:** If the PDF is secured with both types of passwords, it can be opened with either password. However, only the permissions password allows the user to change the restricted features. Because of the added security, setting both types of passwords is often beneficial. All Adobe products enforce the restrictions set by the permissions password. However, if third-party products do not support or respect these settings, document recipients are able to bypass some or all of the restrictions you set.

4. Click to save your work.

> **Note:** File size results with these options may vary.

Specifying PDF Output Via Word or FrameMaker

If you are building either Microsoft Word or Adobe FrameMaker targets, you can send the output to Adobe's Portable Document Format (PDF). When you use this feature, Flare generates both the native Word or FrameMaker documents, in addition to a PDF file.

Creating a PDF Via Microsoft Word

In order to send output to PDF via Microsoft Word, you need to have the following setup:

- You must have Microsoft Word 2007 installed. Previous versions of Word do not support this functionality.
- You must download and install the following free add-in from Microsoft:

  http://www.microsoft.com/downloads/details.aspx?familyid=4D951911-3E7E-4AE6-B059-A2E79ED87041&displaylang=en

> **Note:** With Word 2007 and this add-in from Microsoft, there is no need to have the Adobe Distiller installed.

HOW TO SPECIFY PDF OUTPUT VIA MICROSOFT WORD

1. Open the Word target.
2. In the Target Editor, do one of the following:

   USE THE OUTPUT FILE FIELD ON THE GENERAL TAB

   a. Select the **General** tab.

   b. In the **Output File** field, enter the name that you want to use for the main output file and add the .pdf extension at the end of it. For example, if you want your main output file to be called "MyNewManual," you would enter the following in this field: `MyNewManual.pdf`

USE THE CHECK BOX ON THE ADVANCED TAB

a. Select the **Advanced** tab.

b. Click the check box next to **Generate PDF for MS Word Output**.

> **Note:** If you use this second check box option, it will overrule any file extension specified for the output file name on the General tab.

3. Click to save your work.

Creating a PDF Via Adobe FrameMaker

In order to send output to PDF via Adobe FrameMaker, you need to have the Adobe Distiller installed.

HOW TO SPECIFY PDF OUTPUT VIA ADOBE FRAMEMAKER

1. Open the FrameMaker target.
2. In the Target Editor, do one of the following:

   USE THE OUTPUT FILE FIELD ON THE GENERAL TAB

   a. Select the **General** tab.

   b. In the **Output File** field, enter the name that you want to use for the main output file and add the .pdf extension at the end of it. For example, if you want your main output file to be called "MyNewManual," you would enter the following in this field: `MyNewManual.pdf`

   USE THE CHECK BOX ON THE ADVANCED TAB

   a. Select the **Advanced** tab.

   b. Click the check box next to **Generate PDF for FrameMaker Output**.

   > **Note:** If you use this second check box option, it will overrule any file extension specified for the output file name on the General tab.

3. Click to save your work.

> **Note:** You can also create PDF output directly, without going through Word or FrameMaker.

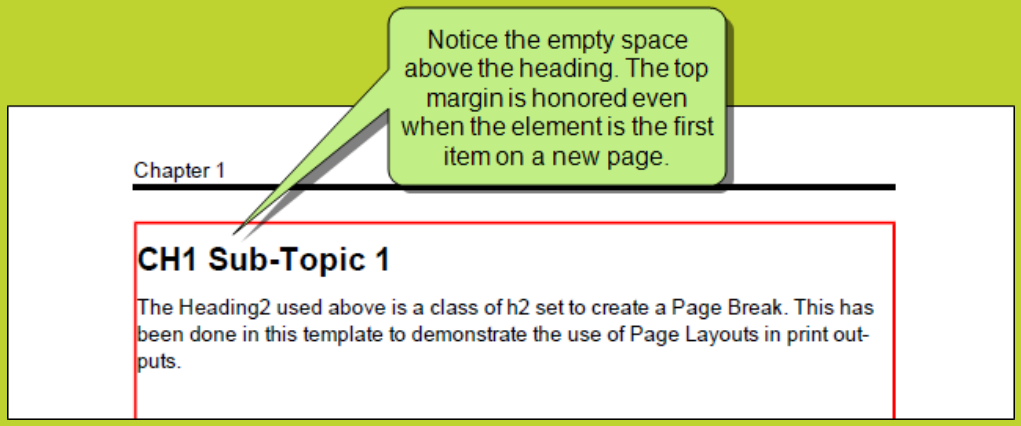
Collapsing Top Margins

This feature is supported in Adobe PDF and Microsoft XPS output.

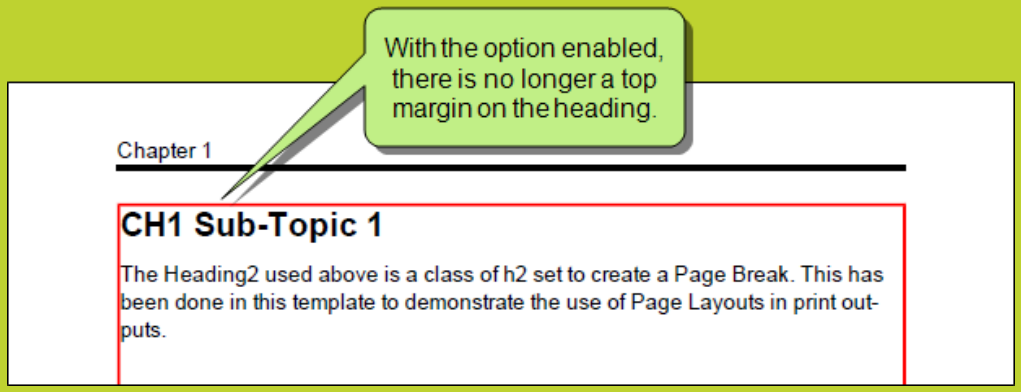
In PDF and XPS targets you can collapse the top margin for all elements occurring at the top of a new page or column.

E X A M P L E

Let's say you have top margins set on your headings. If the "collapse top margin" option is not enabled, the results are something like this (a red border has been added to the body frame so you can see the subtle difference):



If the option is enabled, the results are something like this:



HOW TO COLLAPSE TOP MARGINS

1. Open the target.
2. In the Target Editor, select the **Advanced** tab.
3. In the **Margins** section, select **Collapse top margin at top of page/column**.
4. Click to save your work.

Improving the Processing Performance of the Target

This feature is supported in DotNet Help, HTML5, WebHelp, WebHelp AIR, WebHelp Mobile,and WebHelp Plus output.

You can improve the speed that your output loads for end users in several ways.

HOW TO IMPROVE THE PROCESSING PERFORMANCE OF A TARGET

1. Open a target.
2. In the Target Editor select the **Performance** tab
3. Select the option(s) that you want to use:
   - **Condense JavaScript Files** You can condense any JavaScript files in your output. This causes all of the output's runtime scripts to be merged into one file and improves web server performance by causing there to be fewer requests when loading the output.
   - **Pre-load Images** You can pre-load your images that are not initially visible when output is opened. When those images are triggered to be displayed, they will appear faster since they were pre-loaded. This happens at the cost of the output taking slightly longer to initially open since it must download more content. On the other hand, disabling pre-loading images will cause the output to open faster. Instead they will have to be downloaded when they are triggered to be displayed.
   - **Pre-merge Index File/Pre-merge Search Database File** If you have multiple projects whose outputs are set to be merged when you build the "master" project, you can pre-merge index files and search database files. If you select to pre-merge the index or search database file, you disable runtime merging, which means that you cannot compile and publish a "sub-project" later. You must generate and publish all linked Help systems at the same time. Selecting this option may slow down the generation of the output, but it will significantly increase the speed with which the output loads for the end user.

- **Chunk Large Index/TOC/Glossary/Search Database Files** You can control the size of "chunks" used when generating the index, TOC, glossary, and search database files for a target. After you choose this option, select the number of bytes for each chunk. The chunk size affects the speed that a Help system loads for a user. Depending on the Internet connection, different results will be seen. The smaller the chunk size, the greater the number of chunks that will be sent. A large number of chunks may result in a slower speed, especially if you have many merged "sub-projects" and the index is being merged. Chunking may slow down the generation of your output somewhat, but it increases the speed that the output loads for end users. For best results, you should test the different chunk sizes.

  Note: The option for chunking large glossary files is only available in HTML 5 targets.

- **Include "Stop-Words" in Search** Flare has a hard-coded stop words list that works behind the scenes to filter out common structure-class words (i.e., function words) from search operations. This means the Flare search engine excludes words (e.g., "an," "for," "of, and "the") by default. However, if you want the search engine to search for queries with these words, you can include the stop words list in search.
- **Exclude "Non-Words" from Search** If you have a project that contains a lot of content that is not composed of regular letters and numbers (e.g., angled brackets), you can set your target to avoid these "non-words" when users perform a search in your Help system. This will improve the search performance that your users experience.
- **Exclude Index Entries from Search** If you insert index markers in your project, those markers by default are included in searches that users perform in your output. If you want to exclude index entries from searches, you can do so. See "Excluding Index Entries From Search" on page 105.
- **Include SQL Search Database** For DotNet Help you can create an SQL database when generating the output. This results in faster searches. It also allows people to use asterisk wildcards when performing searches. For example, if you type `Mad*` in the Search field, Flare will list all topics with words that have "Mad" followed by any other letters (e.g., MadCap, Madagascar, madness, Madrid).

  Note: For more information about wildcards, see http://tinyurl.com/279p2nf.

  Note: This option and wildcard searches are supported in DotNet Help output only.

- **N-Gram Size** You can adjust the n-grams, which are character chunks used to separate words in Chinese, Japanese, and Korean projects. Chinese, Japanese, and Korean languages do not have spaces

as word boundaries, so n-grams are used to separate the words. Smaller n-gram values produce smaller search database file sizes. They also lead to more search results, which may be less accurate. On the other hand, larger n-gram values produce larger search database file sizes. They also lead to fewer search results, which may be more accurate. The default size in this field is optimal for most situations.

4. Click to save your work.
5. Build and publish the output.
6. Open the output from the published destination to test the loading performance of the output.

Specifying How Online Features are Converted in Printed Output

The online features from your project (e.g., text hyperlinks, expanding text, Help controls) are converted to print-based output in various ways, as described in the following table.

Online feature	How it is converted in print-based output
Breadcrumbs	Not included in the document
Browse sequences	Not included in the document
Context-sensitive Help	Not included in the document
Cross-references	You can convert the cross-references to page number references (e.g., "See My Topic" can become "See page 36"). This is done by using a stylesheet medium and changing the property settings for the cross-reference style for the print medium. You also need to make sure the print medium is associated with the target you wan to build.
Drop-down text	Hotspot converted to text; body expanded inline
Expanding text	Hotspot converted to text; body can be converted (in the Target Editor) to (1) footnotes, (2) inline text, or (3) removed. See steps below.
Favorites	Not included in the document
Glossary term links	Converted to footnotes at the bottom of pages
Help controls (related topics, concept links, keyword links)	Not included in the document
Popups	Hotspot converted to text; body can be converted (in the Target Editor) to (1) footnotes, (2) inline text, or (3) removed. See steps below.
Rules (horizontal lines)	Not included in the document
Scripts	Not included in the document

Online feature	How it is converted in print-based output
Search	Not included in the document
Text hyperlinks	Converted; for some outputs (e.g., Word), users can press CTRL key and click link to go to destination
Togglers	Hotspot converted to text; destination converted to inline text
Topic popups	Converted to text with link removed

HOW TO SPECIFY THE WAY ONLINE FEATURES ARE CONVERTED IN PRINTED OUTPUT

Some online features in your project are automatically handled in one way or another when you produce printed output. However, there are some features (expanding text and popup effects) where you can specify how you'd like them to be treated in the printed output.

1. Open the target that you plan to build for your printed output. The Target Editor opens.
2. Click the **Advanced** tab.
3. In the **Expanding Text Effects** and **Text Popup Effects** sections, select an option for each type of online feature:
   - **Convert to footnote** The body for the expanding text or popup effect is displayed as a footnote in the printed output.
   - **Expand text inline** The body for the expanding text or popup effect is displayed as a text (after the "hotspot" text) in the printed output.
   - **Remove the "body"** The body for the expanding text or popup effect is not included in the printed output.
4. Click to save your work.

Associating Publishing Destinations with Targets

After you create a publishing destination, you need to associate it with a target that you plan to build. You can associate the same publishing destination with as many targets as you want.

HOW TO ASSOCIATE A PUBLISHING DESTINATION WITH A TARGET

1. Make sure the Project Organizer is open.
2. Double-click the **Targets** folder. The existing targets are shown.
3. Double-click the target that you want to associate with the destination. The target opens in its own page in the Target Editor.
4. Click the **Publishing** tab. All publishing destinations that have been added to the project are displayed. If a destination contains a check mark next to it, that means it is associated with the target. If it does not contain a check mark, it is not yet associated with the target.
5. Click the check box next to any destinations that you want to associate with the target. A check mark appears next to any destinations that you select.
6. Click to save your work.

Integrating Flare and Pulse

To integrate the embedded discussion forms and other Pulse features with your Flare output, add a new community in Flare. This action automatically creates a new system in Pulse, allowing your web server to host, store, and serve the embedded discussion forms that appear below the topics in your Flare output. This system also gives you and your team members the ability to moderate and administer the Pulse features and content from the Pulse dashboard.

Integration Options

If you are developing an online Flare target for a single product, service, or purpose, integrating Flare is a straightforward process. You simply install Pulse and then add a new Flare community to create a corresponding system in Pulse. However, because many organizations develop multiple Flare targets, you also have these options:

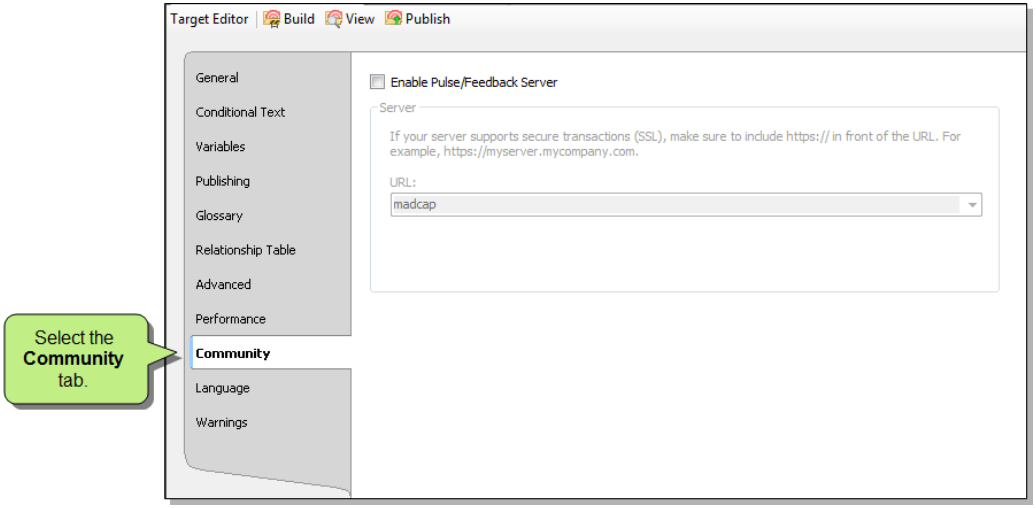
- **Add a Single Flare Community for Use With Multiple Targets** In this scenario, you add one Flare community which creates a corresponding Pulse system. Then you associate that community with multiple Flare targets. This method gives you less-granular reporting capabilities in Pulse.
- **Add a Unique Flare Community for Each Different Target** In this scenario, you add a unique Flare community for each different target. This creates a separate system in Pulse. The advantage of using this option is that it gives you more granular reporting capabilities, letting you generate usage reports for each community.

Enabling the Pulse Server

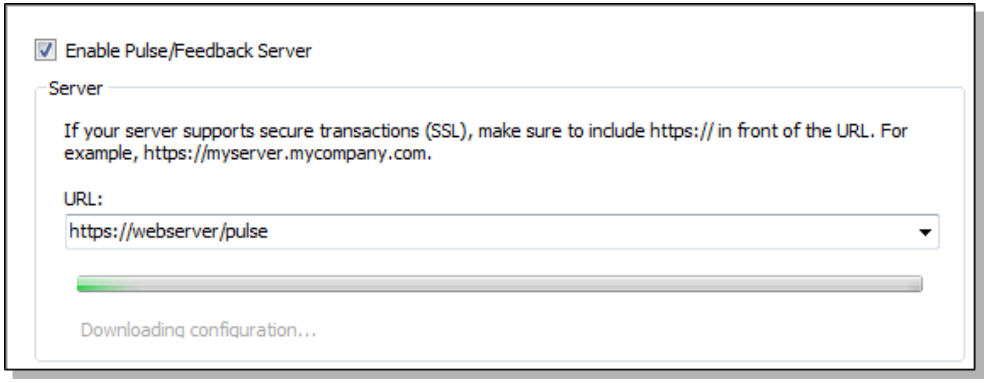
To give Flare the ability to interact with Pulse, you must supply the Flare community in the target with the appropriate connection information and administrative access credentials for the Pulse web server.

HOW TO ENABLE THE PULSE SERVER

1. Open the desired Flare project.
2. In the Project Organizer, open the desired WebHelp or HTML5 target.
3. In the Target Editor, select the **Community** tab.



4. Enable the server as follows:
   a. At the top of the **Community** tab, place a check mark in the **Enable Pulse/Feedback Server** box.
   b. In the **URL** field, type the URL for the Pulse web server.



Note: To find the URL, log in to the Pulse Dashboard as an administrator. Then select **Administration>Settings>General Settings**. The URL above corresponds to the value in the **Site URL** box.

Note: If you have installed a Secure Sockets Layer (SSL) certificate on the Pulse web server, ensure that the URL starts with the https:// protocol (instead of http://).

c. Enter the access credentials as appropriate:

   - If this is the first time you are enabling the Pulse community in the Flare target, the Log In dialog automatically appears.

     OR

   - If you want to change the existing login information, click the **Logout** button to clear the existing credentials. Then click **Login**. The Log In dialog appears.

d. In the Log In dialog, type the appropriate **User name** and **Password**. The account that you use must have administrative permissions to Pulse. You have these options:

   - **Use the PulseAdmin Account** This is the default administrator account that was created by the person who installed Pulse.

» **Use an Account that is a Member of the Administrators Group** This can be any account belonging to a registered Pulse user who is a member of the Administrators group.

e. Click **OK** or press **Enter** to close the Log In dialog.

f. Before you can save the Flare target, you must associate a Flare community with the Pulse system.

Setting Up the Community in Flare

After enabling the Pulse server in the desired target, you can set up the community in Flare. If you are setting up Flare for the first time, you will add a new Flare community.

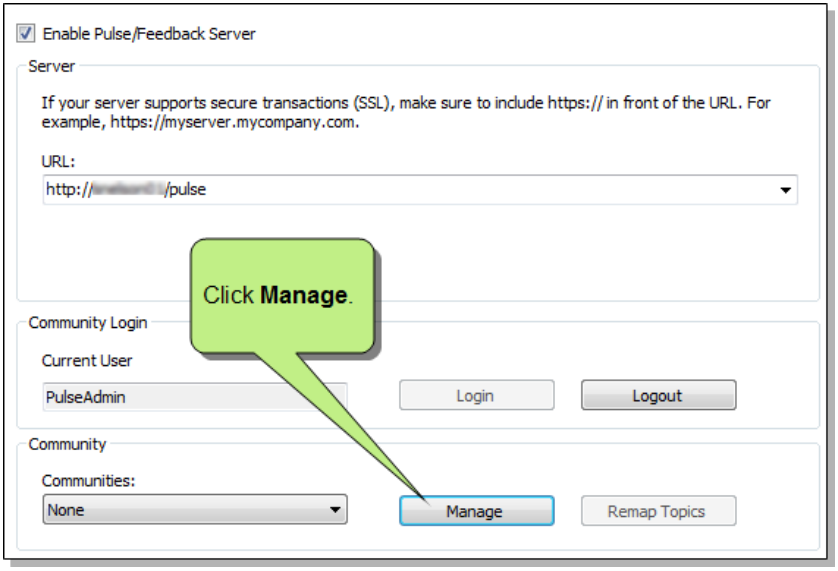
HOW TO ADD A NEW FLARE COMMUNITY

If this is the first time you are integrating Flare andPulse, add a new Flare community to your project's Flare target. This process automatically creates a new "system" on the Pulse server. In Pulse, a system hosts the following items:

- The activity streams that are used when moderating your topics with the Pulse dashboard
- The embedded discussion forms that registered Pulse users can access in the Flare output

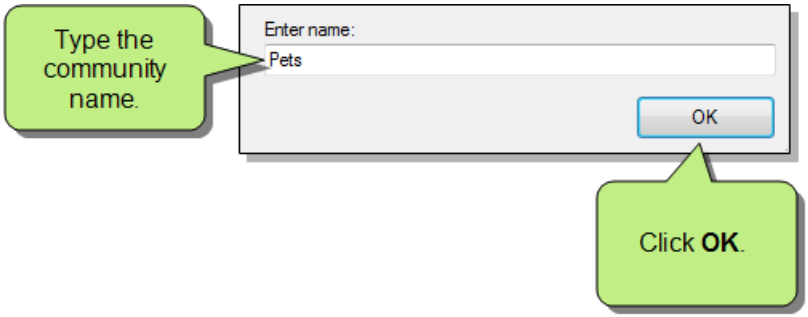
Important: To integrate Flare content with Pulse integration, you must always add new or use existing communities in Flare. While you have the ability to create systems using the Pulse dashboard, you do not have the ability integrate those systems with a Flare community.

1. In the **Community** tab, locate the **Communities** list and then click **Manage**.

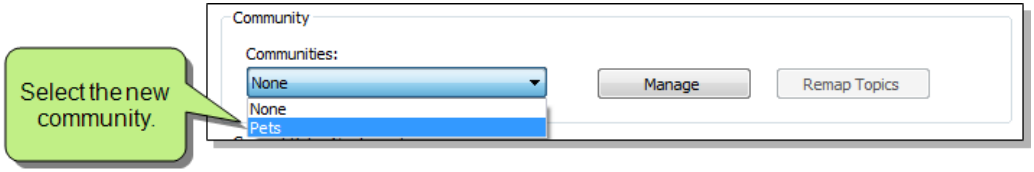


2. In the Manage Community dialog, click **Add**.

3. In the Enter Name dialog, type a name for your community. In this example, we are creating a community to use with a Flare target about Pets. Then click **OK**.



4. In the Manage Communities dialog, the new community appears. Click **OK** to close the dialog.
5. In the **Community** tab, select the community you just created from the **Communities** drop-down list.



6. Click **Refresh** to update the Current Pulse Configuration info. This provides you with basic information about the community, such as the status of the server, and the total number of users.

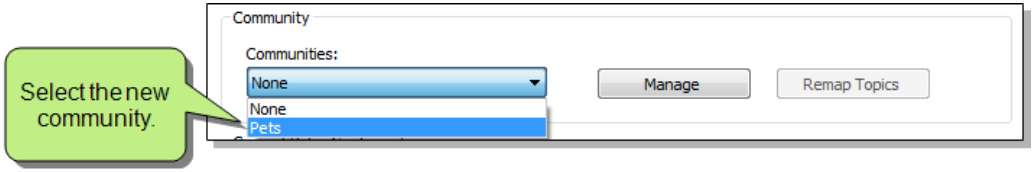
After the community is associated with the Flare target, the **Remap Topics** button becomes available so you can map your community topics to Pulse for the first time.

If needed, you can also have the ability to select an existing community, delete a community, and rename a community as follows:

HOW TO SELECT AN EXISTING FLARE COMMUNITY

If you need to publish any updates for your Flare output and do not wish to create a new community for your documentation release, select the community that you previously created for the target. This gives you the ability to ensure that any new, removed, and updated topics are mapped to the appropriate embedded discussion forms in the Pulse system.

1. In the **Community** tab, locate the **Community** area.
2. From the **Communities** list, select an existing community.



3. Click **Refresh** to update the Current Pulse Configuration info. This provides you with basic information about the community, such as the status of the server, and the total number of users.

Note: The Communities list in Flare initially only includes communities that have been added with the Flare user interface. If you create Pulse systems using the Pulse dashboard, those systems will be not be listed as Flare communities until you remap the topics.

Note: You can associate multiple Flare targets with a single community if desired. However, for best results with reporting, we recommend that you create a separate community for each target. This will also simplify any mapping/remapping tasks when you publish updates.

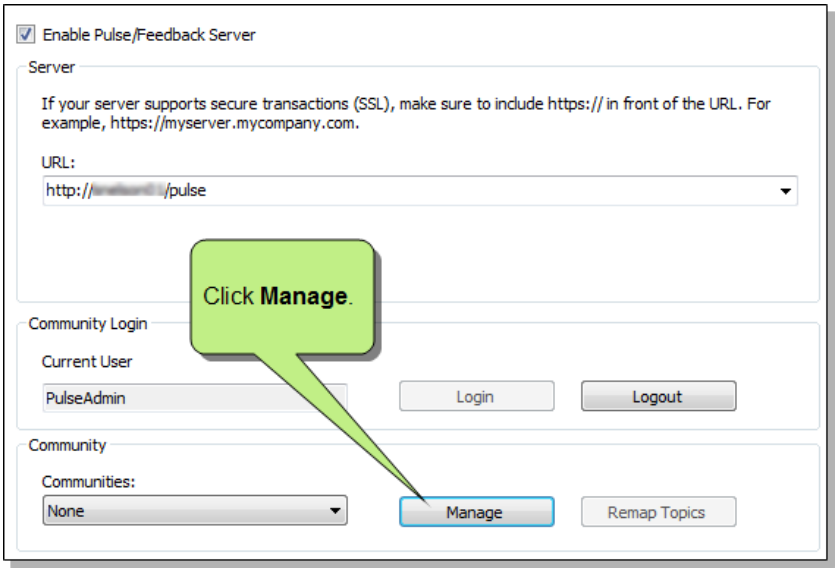
Note: If you want to verify that the Flare community you added is now a system in Pulse, log in to the Pulse dashboard using an account that is a member of the Administrators or Employees group. The select **Systems>Find Systems** from the menu. The community you created will appear in the list.

After the community is activated, the **Remap Topics** button becomes available so you can remap the community topics to Pulse.

HOW TO DELETE A FLARE COMMUNITY

If you want to delete a Flare community, use the steps below. This deletes both the Flare community and the Pulse system.

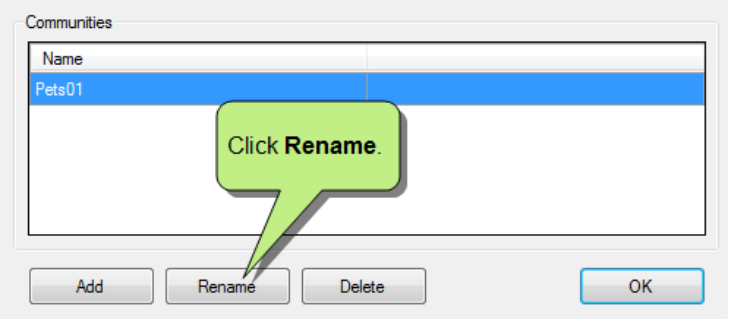
1. In the **Community** tab, locate the **Communities** list and click the **Manage**.



2. In the Manage Community dialog, select the community that you want to remove. Then click **Delete**.
3. When prompted to confirm the delete action, click **Yes**. This removes your selection from the Communities list.
4. Click **OK** to close the Manage Communities dialog.
5. Click **Remap Topics** and confirm the topics to remove.
6. Click to save your work.

HOW TO RENAME A FLARE COMMUNITY

1. In the **Community** tab, under **Current Pulse Configuration**, click the **Refresh** button.
2. Under the **Community** area, click the **Remap Topics** button ensure your topics have been mapped.
3. Under the **Community** area, click the **Manage** button.
4. In the Manage Community dialog, highlight the community that you want to rename. Then click **Rename**.



5. In the Enter Name dialog, type a new name for your community. Then click **OK**.

   This changes the name of the community in Flare, as well as the name of its corresponding Pulse system.

   Note: If an error message appears, try clicking the **Refresh** button and the remapping the topics. This typically resolves any issues.

6. Click **OK** to close the Manage Communities dialog.
7. Click to save your work.

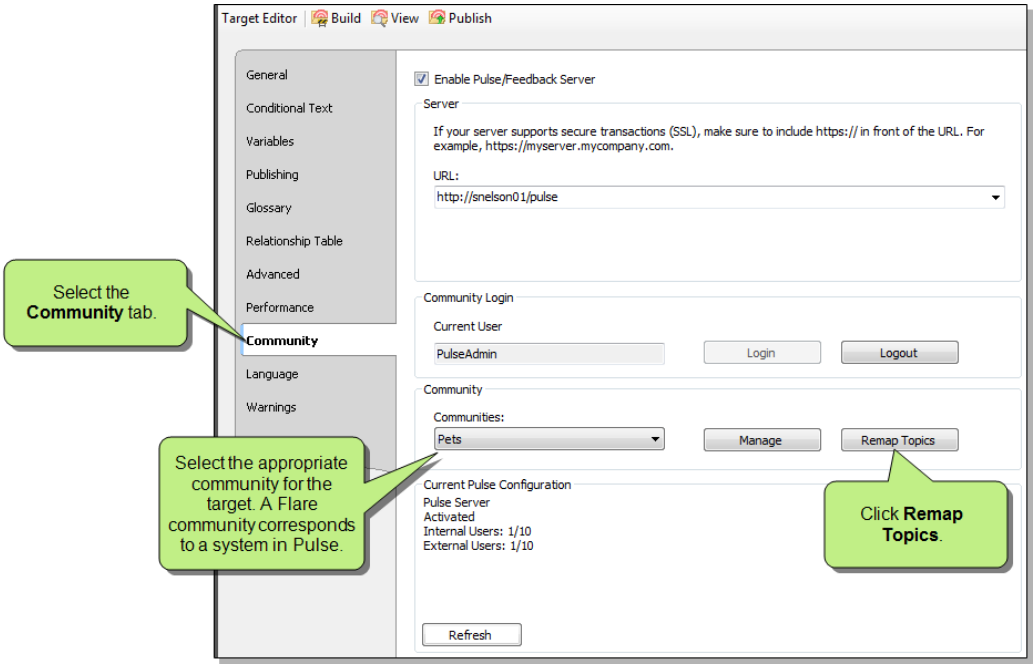
Mapping/Remapping a Flare Community to Pulse

In order to add the embedded discussion forms to the Flare output that you will publish, you must map/remap the community topics in Flare for Pulse.

HOW TO MAP A NEW FLARE COMMUNITY TO A PULSE SYSTEM

If this is the first time you are setting up a Flare community for a target, you will need to map the topics for your community to Pulse.

1. Make the desired changes to your Flare target. Always be sure to update the TOC before remapping the topics.
2. In the Target Editor, select the **Community** tab.
3. From the **Communities** drop-down list, select the appropriate community.
4. Click **Remap Topics**.

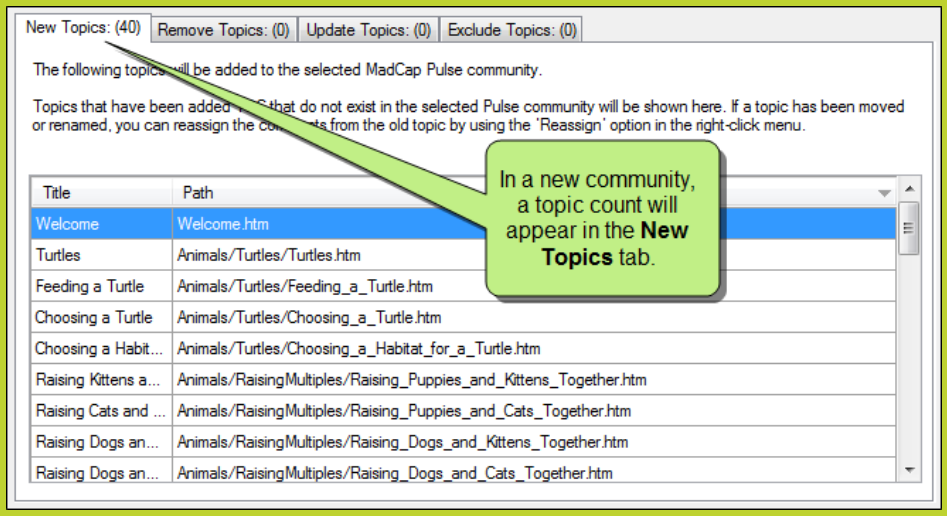


This opens the Remap Community Topics dialog.

5. In the Remap Community Topis dialog, make sure the **New Topics** tab is selected and review the topics in the list.

E X A M P L E

A topic count appears in the label of the **New Topics** tab. The tab lists the topic titles and paths to the topics. These items will be mapped to a corresponding stream in Pulse. In a new community, the counts in the other tabs will typically be zero (0).



6. When finished, click **OK**. The system automatically maps each community to a stream in Pulse. The community updated message appears.

7. Click to save your work.

HOW TO REMAP A FLARE COMMUNITY TO AN EXISTING PULSE SYSTEM

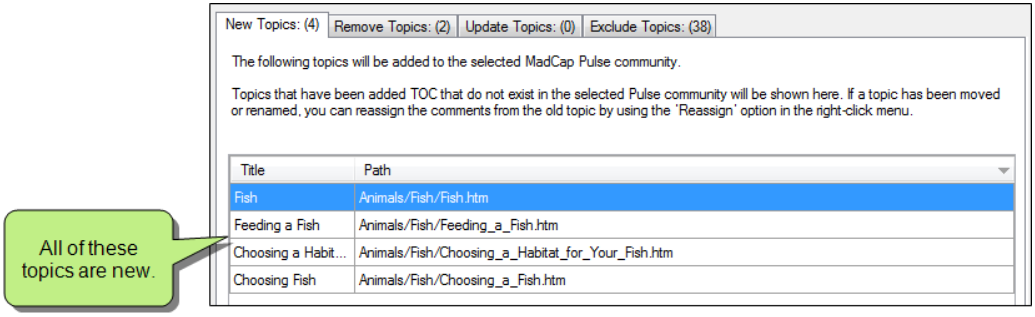
As you prepare to publish content updates to a Flare target, you can review the new, removed, and updated topics and then reassign any of the topics to the appropriate stream in the Pulse system. This stream corresponds to the embedded discussion form at the bottom of the Flare topic.

1. Make your content changes in Flare. Be sure to make all of the desired updates in the Flare TOC associated with your target.
2. Open the Flare target.
3. In the Target Editor, select the **Community** tab.
4. From the **Communities** drop-down list, select the appropriate community.
5. Click **Remap Topics**. This opens the Remap Community Topics dialog so you can review the updates:

   NEW TOPICS TAB

   The New Topics tab lists Flare updates that have an effect on the Pulse mapping definitions:

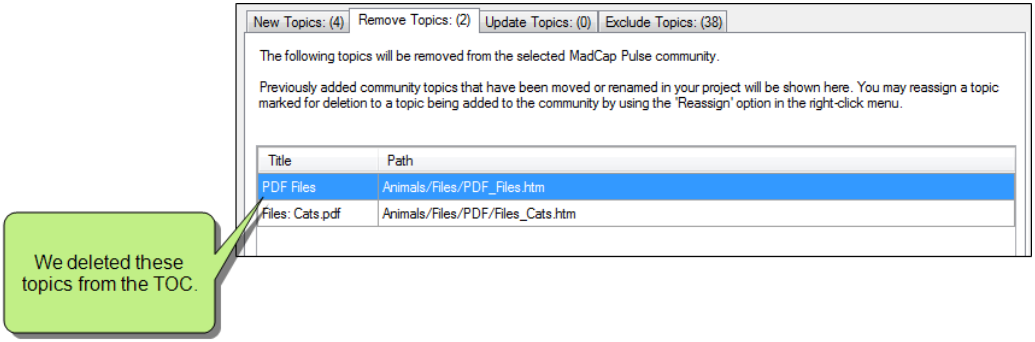
   - **New Topics** New topics that have been added to the Flare TOC associated with the target.
   - **Moved Topics** Existing topics that have been moved to a different folder or location in the Content Explorer.
   - **Renamed Topics (New Name)** Existing topics where the actual file name has changed. The new file name appears in this tab.



REMOVE TOPICS TAB

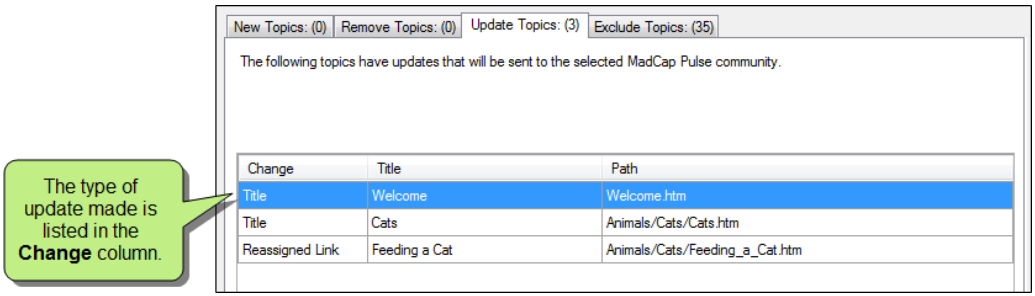
The Remove Topics tab lists Flare updates that have an effect on the Pulse mapping definitions:

- **Deleted Topics** Topics that have been removed from the Flare TOC associated with the target.
- **Moved Topics** Existing topics that have been moved to a different folder or location in the Content Explorer.
- **Renamed Topics (Former Name)** Existing topics where the file name has changed. The former file name appears in this tab.



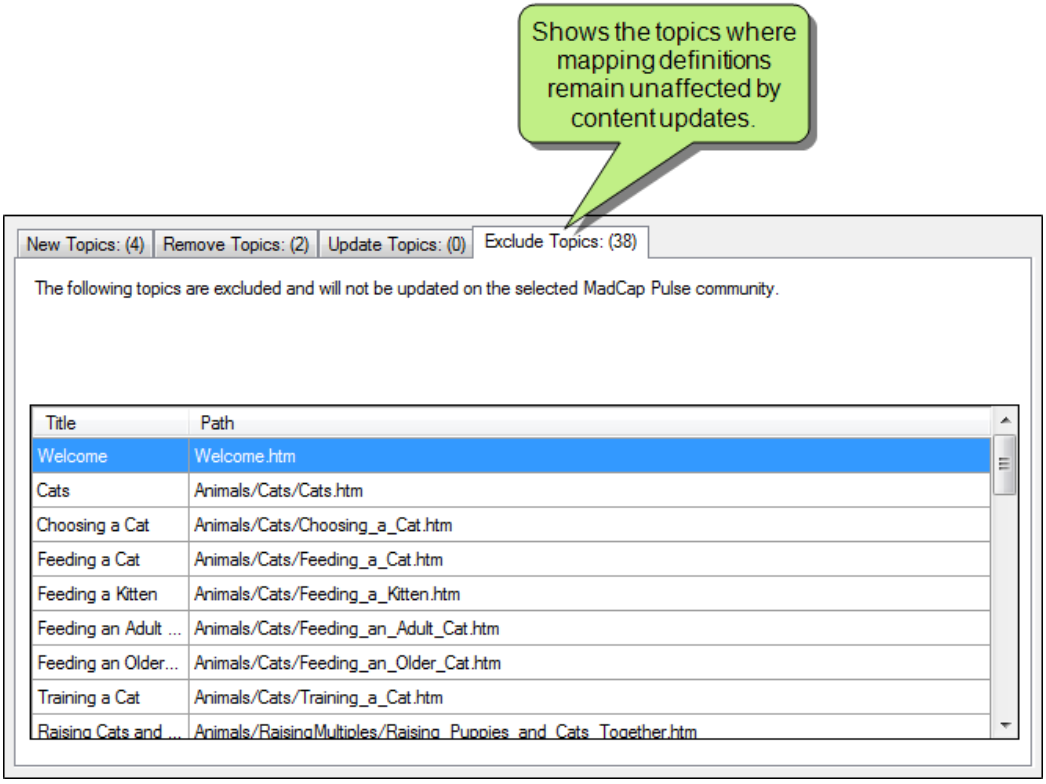
UPDATE TOPICS TAB

The Update Topics tab only lists community topic updates that specifically relate to Pulse mapping definitions. This includes topic updates such as title changes, reassigned links, and so on.



EXCLUDE TOPICS TAB

The Exclude Topics tab lists all topic files where Flare changes have no effect on the Pulse mapping definitions.



6. Review the topics in each tab and then reassign the links as desired. See below for instructions,
7. When finished, click **OK**. The community updated message appears.

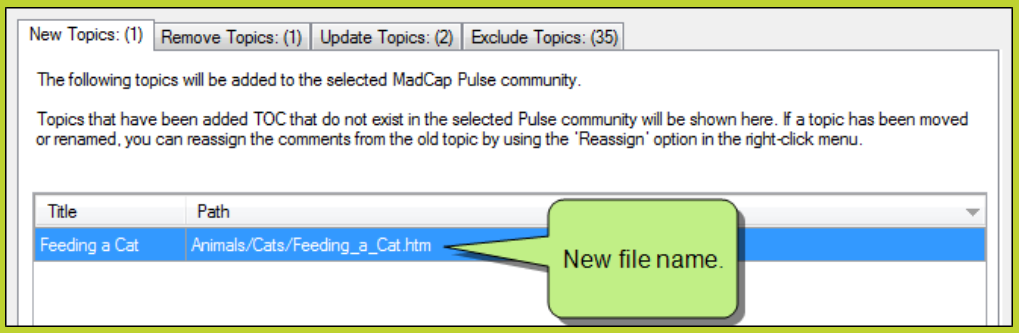
HOW TO REASSIGN A LINK

When you are preparing to publish content updates for aFlare target, there may occasionally be times when you want to reassign a link that is associated with a community topic. This lets you take the embedded discussion form associated with a topic and associate it with a different topic.

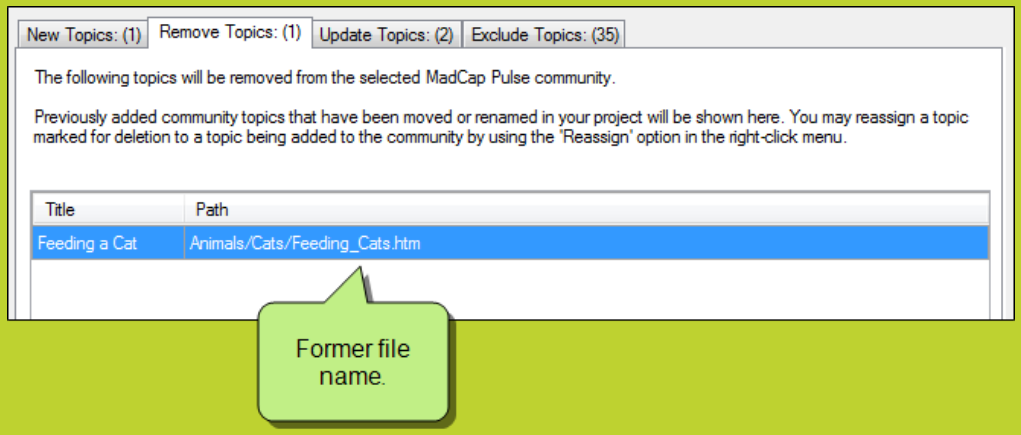
E X A M P L E

Let's say you have a topic with the file name, "Feeding_Cats.htm" and you decide to rename it, "Feeding_a_Cat.htm." When you make this change, the Remap Community Topics dialog will list this change in two places.

The New Topics tab will shown the new topic file name. If you click **OK** in the dialog, the system would add a new stream for the topic in the Pulse system:



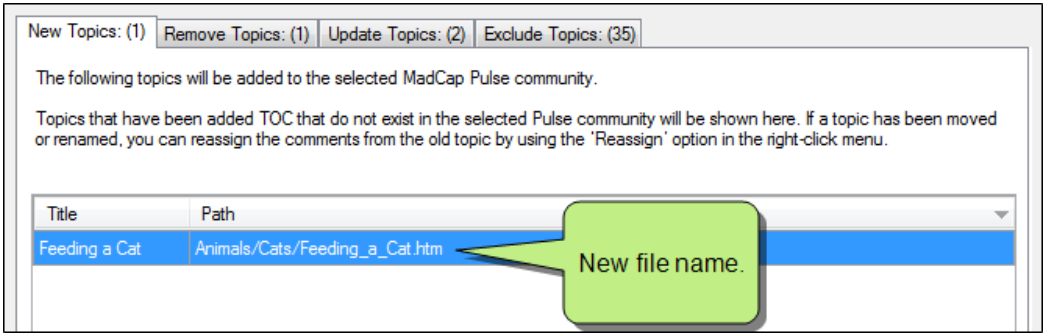
The Remove Topics tab will show the former file name. If you click **OK** in the dialog, the system would remove the corresponding stream for the topic in the Pulse system:



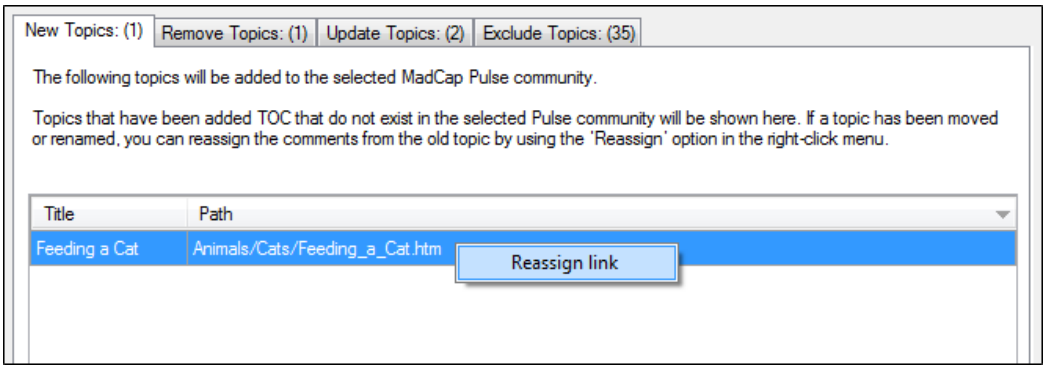
Instead of creating a new stream for the new topic named "Feeding_a_Cat.htm", let's say you want to keep the discussion from the existing stream associated with "Feeding_Cats.htm." You can reassign the link to tie the discussion with the old file name to the new file name.

HOW TO REASSIGN A LINK

1. Make your content changes in Flare. Be sure to make all of the desired updates in the Flare TOC associated with your target.
2. Open the Flare target.
3. In the Target Editor, select the **Community** tab.
4. Select the appropriate Flare community in the **Community** drop-down list.
5. Click **Remap Topics**. This opens the Remap Community Topics dialog so you can review the updates. In this example, we've renamed a topic file from "Feeding_Cats.htm" to "Feeding_a_Cat.htm."

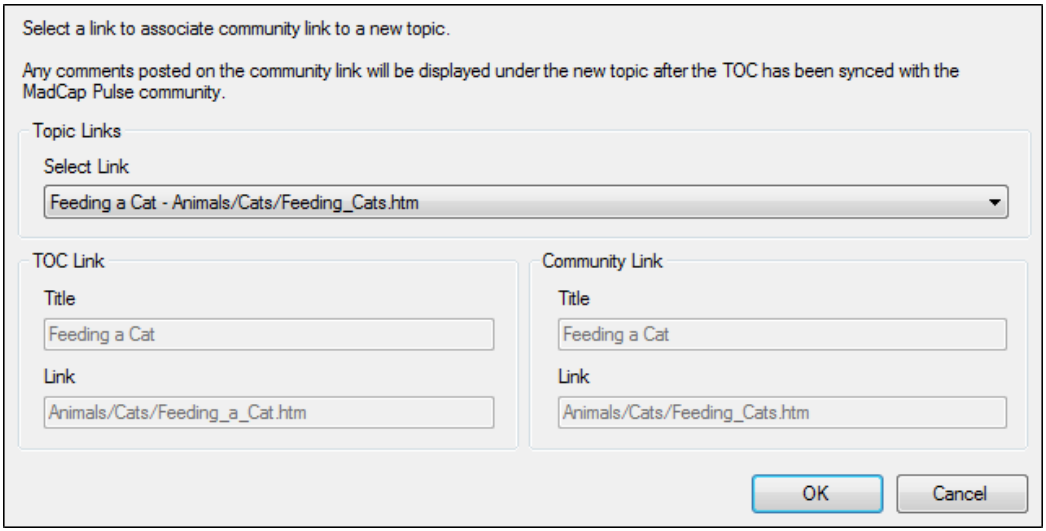


6. In the **New Topic** tab, right-click the new topic and select **Reassign Link** from the context menu.



This opens the Reassign Community Link dialog.

7. From the **Select Links** drop-down list, select the desired topic. This will map the embedded discussion form for the selected link with the topic. In this example, we will select the topic formerly named, "Feeding_Cats.htm."



8. When finished, click **OK**. The community updated message appears.

   When you republish your Flare target, the discussion originally linked to the former file is now linked to the new file.

HOW TO VERIFY THE COMMUNITY TOPICS IN THE PULSE SYSTEM

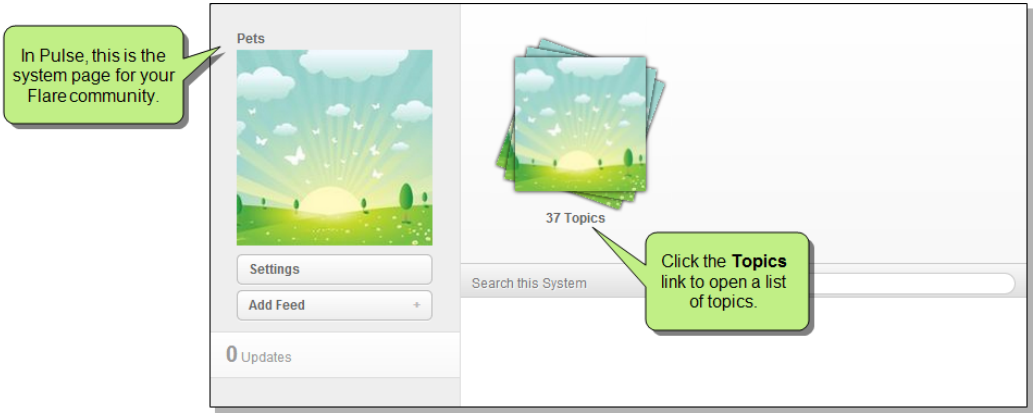
To see the community topics that have been created for your Flare output:

1. Log in to the Pulse dashboard with an appropriate user account.

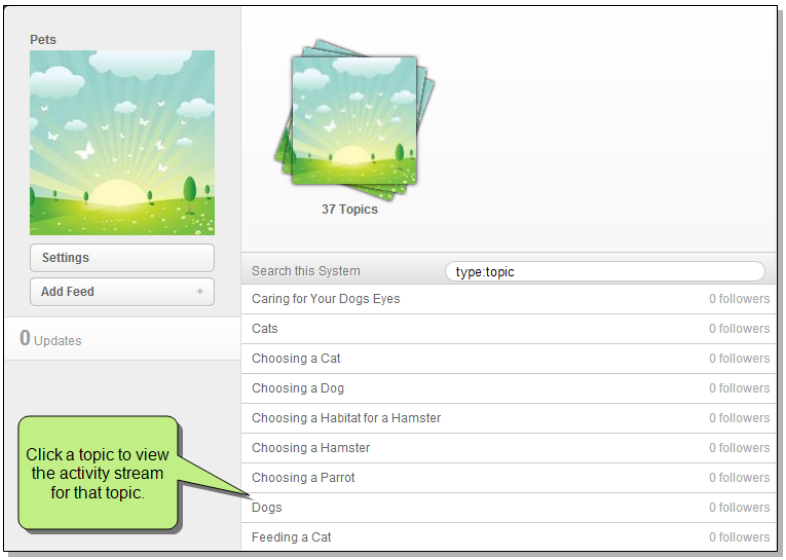
> **Note:** To access systems, the Pulse user account must be a member of the Administrators or Employees group.

2. In the menu bar, select **Systems>Find Systems**. The Find Systems page appears.
3. Under **New Systems**, click the Pulse system that corresponds to the appropriate Flare community.

4. In the community page, click the **Topics** link.

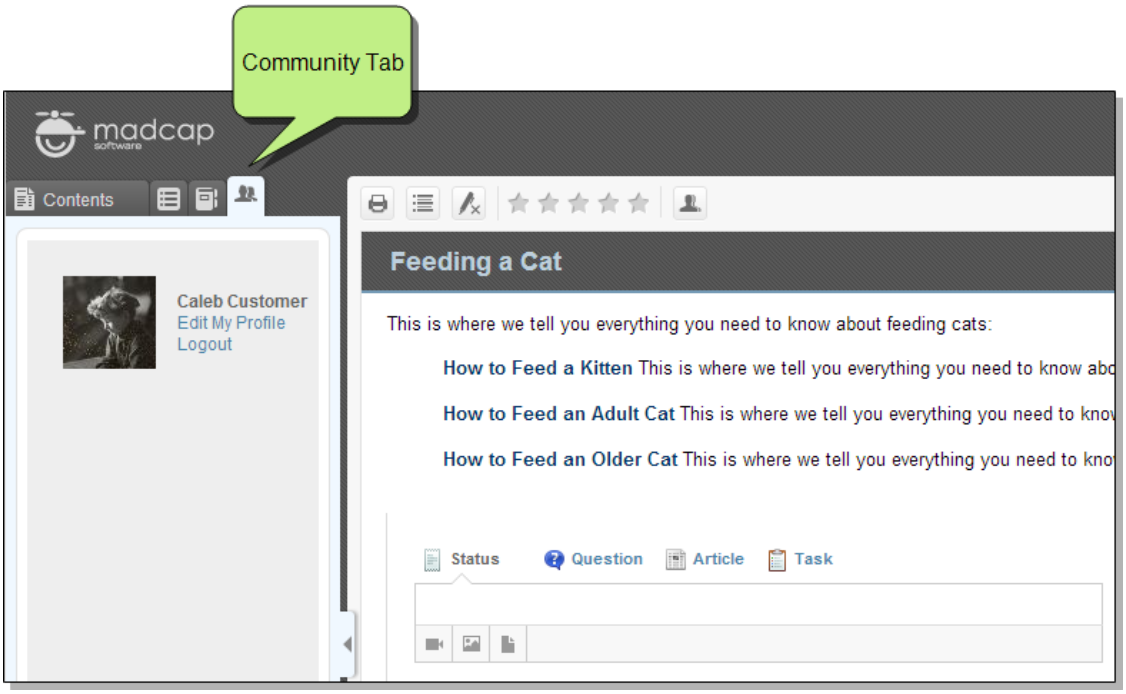


5. In the list of topics, click a topic. Any community activity for the topic will be reflected in the stream.



Enabling the Community Tab in the Output

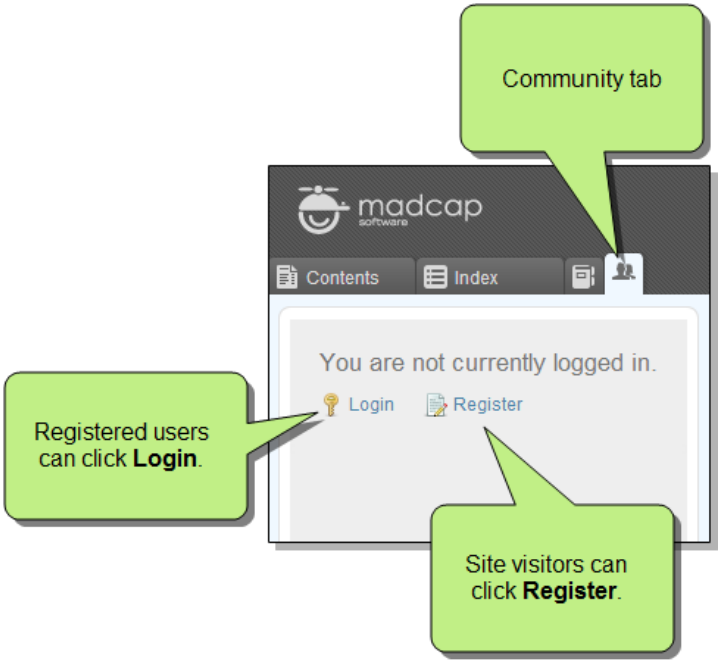
The final step in the integration process is to decide whether you want to include or exclude the Community tab with your published output. Below is an example of the Community tab:



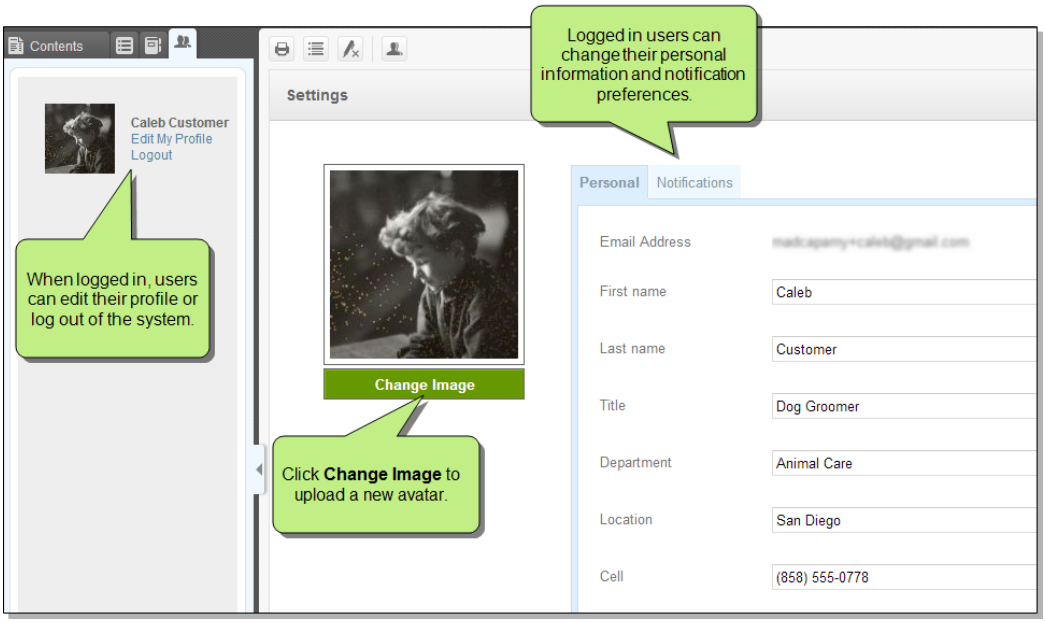
You can enable or disable this feature by editing the general features of the skin associated with the Flare target.

The benefits of including the Community tab in your output include:

- Gives site visitors the ability to register for a Pulse account. Also gives registered users the ability to log in to Pulse directly from the Flare output.



» Gives registered users the ability to edit their profile, which includes changing their avatar image, updating personal information, and modifying notification preferences.



Setting the Community Options for the Skin

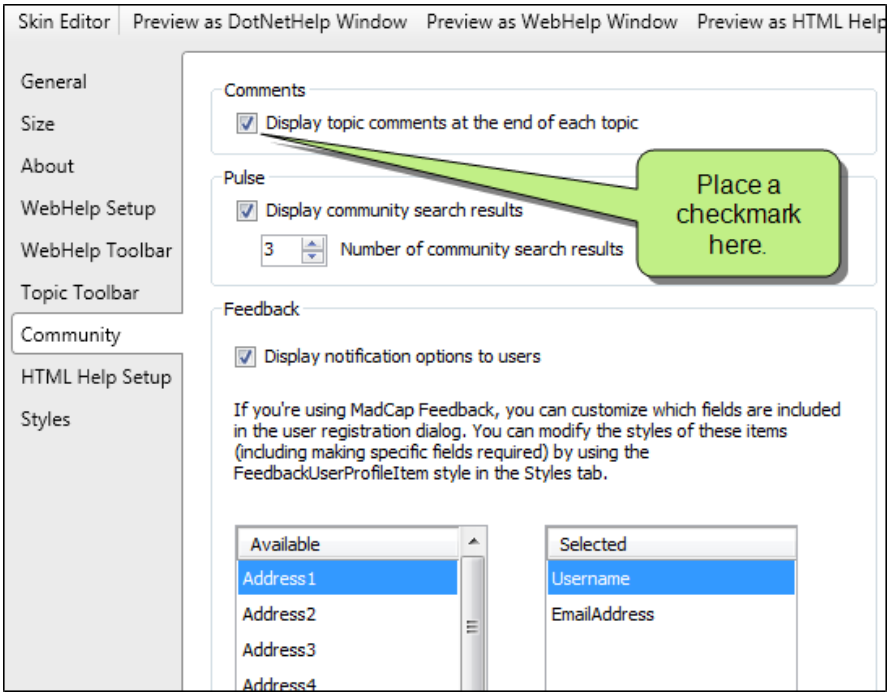
You can open a Standard or HTML5 skin and set community options.

HOW TO SET THE COMMUNITY OPTIONS FOR THE SKIN

1. In the Project Organizer, open a Standard or HTML5 skin.
2. In the Skin Editor, click the **Community** tab. You have these options:

   HOW TO DISPLAY TOPIC COMMENTS

   If you want registered Pulse users to have the ability to post comments in the embedded discussion forms that Pulse includes at the bottom of the Flare topics, place a check mark in the **Display topic comments at the end of each topic** check box. This check box is selected by default.

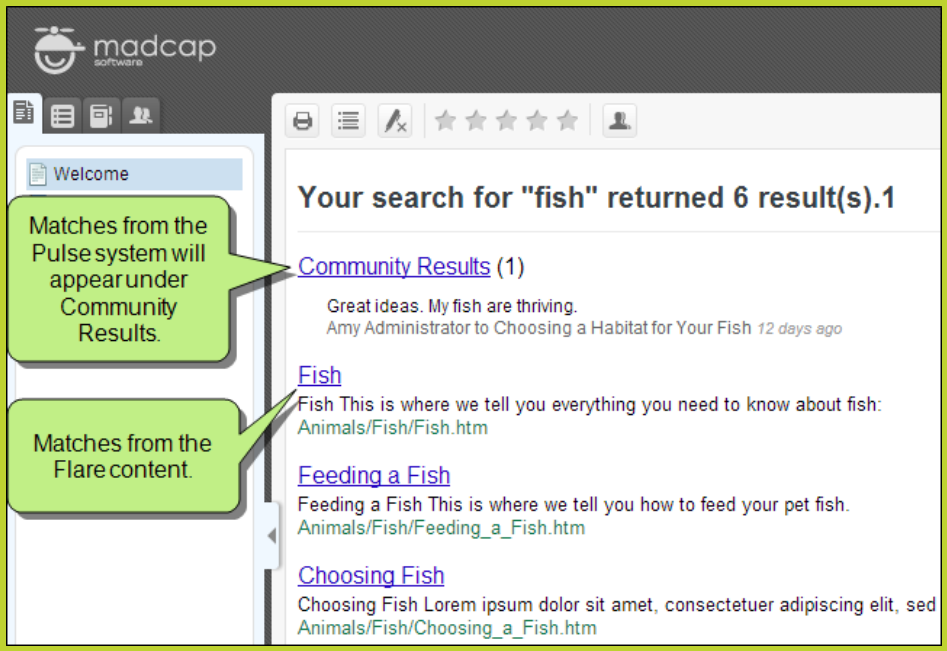


HOW TO DISPLAY COMMUNITY SEARCH RESULTS

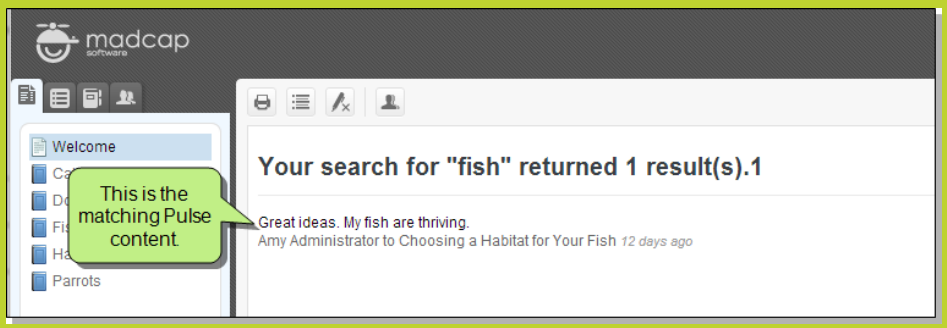
You can configure the Flare skin to give the Flare search engine the ability to index the Pulse system and return any community results when end users perform searches.

E X A M P L E S

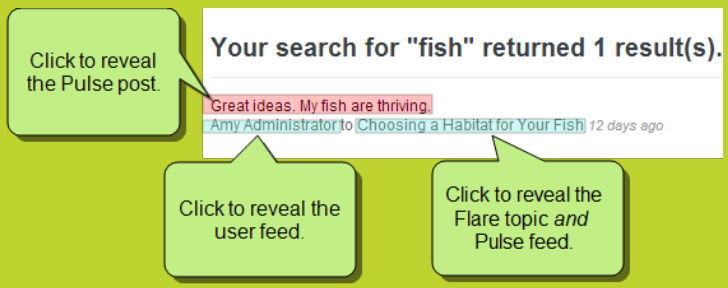
In HTML5 output, Pulse posts matching the search query are listed on the Search Results page under a **Community Results** link:



If a user clicks the **Community Results** link shown above, only the matching Pulse content appears in the results page:



Users can then hover the mouse cursor over the content to reveal a links that open either the matching Pulsepost, the matching Flare topic and the matchingPulse post, or the Pulse user feed.



This feature works similarly in WebHelp output.

HOW TO DISPLAY SEARCH RESULTS

a. Open a Standard or HTML5 skin.

b. Click the **Community** tab.

c. Place a check mark in the **Display community search results** box.

d. (Optional) In the **Number of community search results** field, you can enter the number of community search results shown by default when a search is made in the output. The default number in this field is 3.

> **Note:** The Feedback area in the Community tab of the Skin Editor is only relevant to Feedback Server Admin. The Feedback option is not supported by Pulse.

3. Click to save your work.

Building and Publishing the Flare Output

Once you set up the community features, you can build and publish your Flare output.

Setting Redacted Text on Targets

This feature is supported in Adobe PDF and Microsoft XPS output.

For each output (i.e., PDF or XPS document) that needs to include redacted content, you can open the target. Then you can use the Advanced tab to specify how the content marked as "redacted" should be treated in the output.

HOW TO SET REDACTED TEXT ON TARGETS

1. Open the appropriate PDF or XPS target.
2. In the Target Editor, select the **Advanced** tab.
3. In the **Redacted text** section, select one of the following.
   - **Black out** Displays black rectangles in places of content that has been redacted.
   - **Highlight** Displays content that has been redacted with a highlighted background.
   - **Display as normal text** Displays content that has been redacted in the same way that it displays other content.

   Why are there choices other than "black out"? Those options are available in case you need to generate another output that shows the redacted text.

   E X A M P L E

   Let's say you need to generate one PDF that blacks out the redacted text (e.g., the version being released to the public) and another PDF that shows the text rather than hiding it (e.g., the version being distributed to authorized individuals). In that case, you can open the target to be used for public release and set the redacted text option to "Black out." For the target to be used for authorized individuals, you can set the redacted text option to either "Highlight" or "Display as normal text."

4. Click to save your work.

Associating Relationship Tables with Targets

This feature is supported in all outputs.

If you have created relationship tables in your project, you need to tell Flare which tables to use for which targets.

HOW TO ASSOCIATE RELATIONSHIP TABLES WITH TARGETS

1. Open the target from the Project Organizer.
2. In the Target Editor, click the **Relationship Table** tab.
3. Click next to each relationship table that you want to use for that target so that it has a check mark.
4. Click to save your work.

Associating a Search Filter Set with a Target

This feature is supported in DotNet Help, HTML5, WebHelp, WebHelp AIR, and WebHelp Plus output.

After you create search filters, you need to associate the search filter set with the target that you want to build.

HOW TO ASSOCIATE A SEARCH FILTER SET WITH A TARGET

1. Open the target that you plan to build.
2. In the Target Editor, click the **Advanced** tab.
3. Click the drop-down arrow in the **Filter File** field, and select the search filter set that you want to associate with the target.
4. Click to save your work.

Enabling Partial-word Search

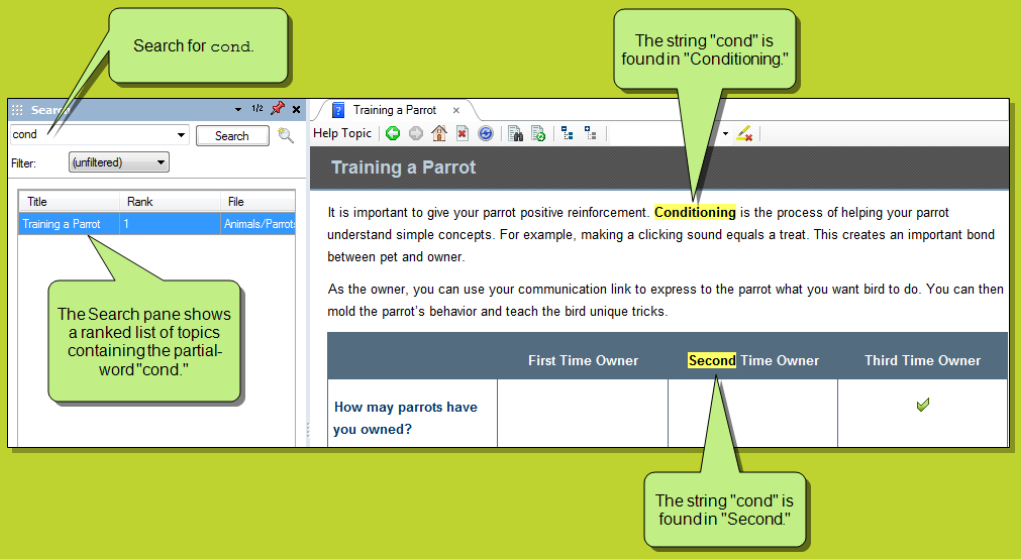
This feature is supported in DotNet Help, HTML5, WebHelp, WebHelp AIR, WebHelp Mobile,and WebHelp Plus output.

For some online outputs, you can provide end users with partial-word search capabilities. This lets users type part of a word or any string (including numbers), press **Enter** or click the **Search** button, and see search results matching those characters.

E X A M P L E

Partial search for: `cond`

Search finds matches in "conditioning" and "second."



This feature is not limited to words only. It works with any string (including numbers) that starts and ends with a space. For example, let's say that the IP address numbered "192.168.10.1" is included in one of your topics. To find the topic that includes the match, simply type `10.1` in the search field.

HOW TO ENABLE PARTIAL-WORD SEARCH

1. In the Project Organizer, open the target.
2. Select the **Performance** tab.
3. In the **Search Database** section, place a check mark in the **Enable Partial Word Searching** box.
4. In the **Minimum Word Size** field, enter the number of characters a user must type before seeing results. The minimum value is 3.

   Note: The higher the minimum word size that you enter, the smaller the search database.

5. Click to save your work.

Important: Enabling partial-word search in client-based outputs increases the size of the search database included with your target output. It also increases the amount of time it takes to build and publish the output.

Generating Sitemaps for SEO

This feature is supported in HTML5, WebHelp, WebHelp Mobile, and WebHelp Plus output.

You can generate a sitemap when compiling your output. This helps with search engine optimization (SEO), making it easier for search indexing services (i.e., spiders, crawlers, or bots) to find your output. Therefore, the entire output is indexed and search engine results are improved.

HOW TO GENERATE A SITEMAP FOR SEARCH ENGINE OPTIMIZATION

1. Open a target.
2. On the **Advanced** tab of the Target Editor, select **Generate Sitemap**.
3. In the **Web URL** field enter the path where the output is ultimately published. For example, our own output is published here: http://webhelp.madcapsoftware.com/flare10/.

   Note: Be sure to include the full path—including “http://”—when entering the web URL.

4. Click to save your work.

Associating Skins with Targets

This feature is supported in DotNet Help, Microsoft HTML Help, HTML5, WebHelp, WebHelp AIR, WebHelp Mobile, and WebHelp Plus output.

After you add a skin to your project and edit the settings, you need to associate it with a target. After you build the target, the output will be displayed in the skin.

HOW TO ASSOCIATE A SKIN WITH A TARGET

1. Open the target from the Project Organizer.
2. On the **General** tab of the Target Editor, click the drop-down arrow in the **Skin** field and select the skin that you want to associate with the target.
3. Click to save your work.

Setting All Skins to be Generated

This feature is supported in DotNet Help, Microsoft HTML Help, WebHelp, WebHelp AIR, and WebHelp Plus output.

An option on the Advanced tab of the Target Editor controls whether every skin in the project or just the selected skin in the target (which is set on the General tab) will be generated for the output. This option is enabled by default.

The reason to keep this option enabled is if you need the ability to make context-sensitive Help (CSH) calls to different skins. In this case, each of those skins must be generated so that they are available in the output for the CSH calls.

If you are not using multiple skins for CSH calls, you might want to disable this option. If you disable it, the size of the entire output will be minimized because unnecessary skins will be excluded.

HOW TO SET ALL SKINS TO BE GENERATED

1. Open a target used for online output.
2. On the **Advanced** tab of the Target Editor, select **Generate all skins** so that it has a check mark.
3. Click to save your work.

Note: For HTML5 targets, this option is not necessary because all skins are always automatically generated. This is due to the fact that compiling all skins does not affect the file size in HTML5 like it does in other output types.

Getting the Latest Version

After you bind a Flare project to a source control application, you can get the latest version of any of the source control files. When you do this, you are copying the most current files stored in the source control application to your local Flare project without necessarily checking out the files. This means that the "Read Only" designation will remain associated with the files until you check them out.

Following are steps for getting the latest version of all files in a project automatically, as well as steps for getting the latest version of files manually.

Automatic Get

You might use this option if you are working with a team of authors and want to make sure that you include the latest changes from other writers in the output (without having to manually get those files).

With this option:

- You will not be prompted before the "get" is performed.
- Flare will not get the latest copy of the files in the Targets folder, because that would conflict with the generation of the output.
- Conflicts with files will not cause local files to be overwritten. Therefore, if your local files are writable or already checked out, those files will be kept, rather than overwritten with the source control files.

HOW TO GET THE LATEST VERSION OF SOURCE CONTROL FILES AUTOMATICALLY

1. Open the target.
2. In the Target Editor, select the **General** tab.
3. Select the check box labeled **Automatically get latest version of all files before generating the target**.
4. Click to save your work.
5. Build the target.

Note: The "automatic get" feature is not supported if you are building output using the command line, as opposed to the Flare interface.

Manual Get

You can manually get the latest version of all of the files in the Flare project or specific files only.

HOW TO GET THE LATEST VERSION OF SOURCE CONTROL FILES MANUALLY

1. Do one of the following, depending on the part of the user interface you are using:
   - **Pending Check-Ins Window Pane** From the **Source Control** ribbon or **View** menu, open the Pending Check-Ins window pane. Select the files in the window pane that you want to check out, right-click, and select **Source Control>Get Latest Version** (for selected files) or **Source Control>Project>Get Latest Version All** (for all files in the project).
   - **Ribbon** Select the **Source Control** ribbon. In the **File** section select **Get Latest Version** (for selected files). Alternatively, in the **Project** section you can click **Get Latest Version All** (for all files in the project).
   - **Menu** Select **File>Source Control>Get Latest Version** (for selected files) or **File>Source Control>Project>Get Latest Version All** (for all files in the project).
   - **Standard Toolbar** Click the down arrow next to the **Source Control** button and select **Get Latest Version** (for selected files) or **Project>Get Latest Version All** (for all files in the project).

     To open the Standard toolbar from the menu view, you can select **View>Toolbars>Standard**.
2. If the local and server files are the same, a message tells you so. Click **OK**.

   If the local file is different form the file on the server, the Resolve Conflicts dialog opens. If you want to accept all of the differences between the local and server files, thus merging them, click **Auto Merge All**. If you want to review the differences in the files side by side and resolve each conflict, click **Resolve**.

Note: By default, when you open a project that is bound to source control, a message automatically asks if you wanted to get the latest version of files. However, you can disable this prompt in the Options dialog.

Specifying the Startup Topic in a Target

This feature is supported in Microsoft HTML Help, HTML5, WebHelp, WebHelp AIR, and WebHelp Plus output.

You can specify which topic is the first one that users see when they open the online output. If you have written a welcome or introduction topic, you will likely want this to be the "startup" topic displayed in your output.

HOW TO SPECIFY THE STARTUP TOPIC IN A TARGET

1. Open the target.
2. On the **General** tab of the Target Editor, click the browse button ... next to the **Startup Topic** field.
3. In the Link to Topic dialog, find and double-click the appropriate topic.
4. Click to save your work.

Applying Master Stylesheets

When you want to use styles in your content, the stylesheet needs to be made available for the content in question. In Flare, you can associate regular stylesheets with a single topic. However, you also have the option of using a regular stylesheet as a "master," applying it at either the project or target level, or both.

- **Master Stylesheets at the Project Level** You can associate a master stylesheet with an entire Flare project. The styles will therefore be available to all content in the project. You can set the project-level master stylesheet in the Project Properties dialog (select **Project>Project Properties**).
- **Master Stylesheets at the Target Level** You can associate a master stylesheet with an entire Flare target. The styles will therefore be used for the content in the target. If you have specified a master stylesheet at the project level and another at a target level, the stylesheet at the target will take precedence. If you are using different master stylesheets for different targets, the stylesheet associated with the primary target determines what you see in the XML Editor. You can set the target-level master stylesheet on the General tab of the Target Editor.

HOW TO APPLY A MASTER STYLESHEET TO A PROJECT

1. Do one of the following, depending on the part of the user interface you are using:
   - **Ribbon** Select the **Project** ribbon. In the **Properties** section select **Project Properties**.
   - **Menu** Select **Project>Project Properties**.
   - **Project Toolbar** Click Project Properties...

     To open the Project toolbar from the menu view, select **View>Toolbars>Project**.

   The Project Properties dialog opens.
2. Select the **Defaults** tab.
3. In the **Master Stylesheet** field, click the drop-down and select the stylesheet to be used as the master.
4. Click **OK**.
5. Click to save your work.

HOW TO APPLY A MASTER STYLESHEET TO A TARGET

1. From the Project Organizer open the target.
2. Select the **General** tab.
3. In the **Master Stylesheet** field, click the drop-down and select the stylesheet to be used as the master.
4. Click to save your work.

Note: If you apply a master stylesheet at either the target or project level, you will no longer be able to associate stylesheets at the topic level (unless you remove the master stylesheet associations at the project and/or target level). If you have previously applied multiple stylesheets to a single topic and then switch to using a master stylesheet, you should make sure that all of the styles you need from the various stylesheets are all included in your master stylesheet.

Associating a Medium with a Target

This feature is supported in all outputs.

After you decide on a medium for an output, you need to associate it with the target. After you build the target, the medium will be used to display the correct style settings in the output.

HOW TO ASSOCIATE A MEDIUM WITH A TARGET

1. Open the target.
2. In the Target Editor, click the **Advanced** tab.
3. Click the drop-down arrow in the **Medium** field, and select the stylesheet medium that you want to associate with the target.
4. Click to save your work.

Note: When you create a print-based target (Adobe FrameMaker, Adobe PDF, Microsoft Word, Microsoft XPS, XHTML), the "print" medium is automatically associated with that target (on the Advanced tab of the Target Editor). Of course, you can always select a different medium if you want.

Associating a Master Table of Contents with a Target

In most situations, you will have one TOC that you use for a particular output (target). In that case, you simply associate the appropriate TOC with the target. If you have multiple TOCs that you want to include in the same project or output target, the TOC that you associate with the project or target serves as the "master" TOC. In your master TOC, you have the option of creating links to the other TOC that you want to include in the output. If you do not select a TOC, Flare will use the first one in the project (if there is more than one). If you have specified a master TOC at the project level and another at a target level, the TOC at the target will take precedence.

HOW TO ASSOCIATE A MASTER TOC WITH A TARGET

1. Open the target from the Project Organizer.
2. On the **General** tab in the Target Editor, click the drop-down arrow in the **Master TOC** field, and select the TOC that you want to associate with the target.
3. Click to save your work.

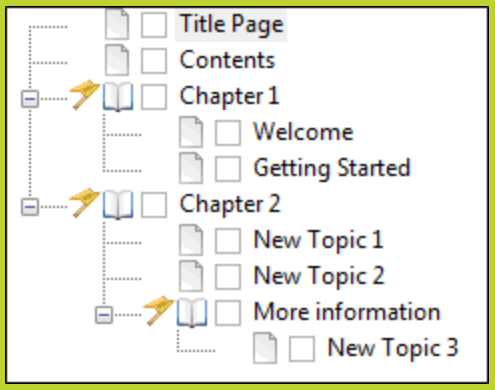
Using the TOC Depth for Heading Levels in Print Output

When creating print output, you can determine which topic headings will be displayed at which level in the generated TOC. For example, even though all of your topics may use a heading style of <h1>, you might want some of those topic headings to display at the first level in the print TOC (acting as an <h1>), others at the second level (acting as an <h2>), and still others at the third level (acting as an <h3>).

You can base the heading levels on the structure you create in the outline TOC. If a TOC entry is at the extreme left of the TOC Editor, it will display at the first level of the print TOC; if an entry is indented to the right one level, it will display at the second level; and so on. Not only will those headings be reflected in the generated TOC in the output, but the corresponding headings will be converted in the main body too (e.g., if a heading becomes an <h2> in the TOC, it will become an <h2> in the text as well).

E X A M P L E

Let's say the first paragraph in each topic that you create uses the <h1> style (except your title page and print TOC topic), and you structure the outline TOC like this:



If you do not use the depth feature, the print TOC will look like this in the output:

Welcome **2**

Getting Started **3**

NewTopic1 ..... **4**

NewTopic2 ..... **5**

NewTopic3 ..... **6**

But if you turn this feature on, the result will instead look like this in the output:

Welcome 2

Getting Started 3

NewTopic1 4

NewTopic2 5

NewTopic3 5

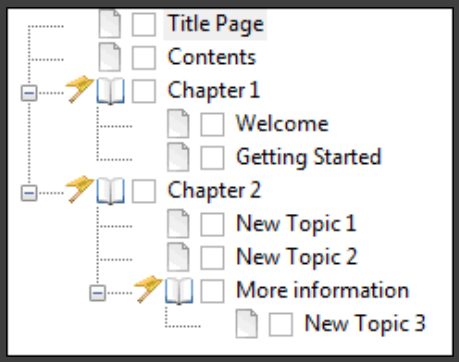
Notice that "NewTopic3" is indented to the third level in the print TOC (because that is its position in the outline TOC). The other topics are at the second level. The books ("Chapter 1" and "Chapter 2") are at the first level, but they are not linked to topics; therefore, they are not included in the output. If you want the unlinked books to be included in the output, you can create headings for them (see the online Help for steps).

HOW TO USE TOC DEPTH FOR HEADING LEVELS

1. Open the target.
2. In the Target Editor, select the **Advanced** tab.
3. In the **Generated TOC** section, select **Use TOC depth for heading levels**.
4. Click to save your work.

Note: If you create books in the outline TOC (whether they are linked to topics or not), they will affect the heading level in the print TOC. If you want unlinked books to be converted to headings in the output (both in the print TOC and in the main body), you can do so. See "Creating Headings for Unlinked Books in a Generated TOC" on the next page.

Note: What happens if you are using classes of heading styles and then select the "Use TOC depth for heading levels option"? For example, let's say you have classes of your <h1> and <h2> heading styles. Suppose you created h1.Special (with a blue font), and h2.Special (with a red font). Meanwhile the primary <h1> and <h2> styles both use a black font. In your topics, you've applied your heading style classes to some heading content. And in your outline TOC, you organized the structure like this:



Because you selected the option to use TOC depth for heading levels, some of the level-1 headings will become level-2 headings or even level-3 headings in the output. If you have headings where you used h1.Special but their level in the output was not affected (i.e., they remain as level-1 headings), they will keep the properties from the style class. In other words, they will stay blue. But if the structure of the outline TOC causes them to become level-2 headings in the output, they will take on the properties of the h2.Special tag—because the class has the same name ("Special") as its <h1> counterpart. Therefore, those headings will have a red font. However, let's say that your level-2 class was named something different (e.g., h2.Red). In that case, h1.Special headings that are pushed to become level-2 headings will instead use properties from the main <h2> tag. In other words, they will have a black font—not blue like h1.Special and not red like h2.Red.

Creating Headings for Unlinked Books in a Generated TOC

This feature is supported in EPUB and in all print outputs (Adobe FrameMaker, Adobe PDF, Microsoft Word, Microsoft XPS, XHTML).

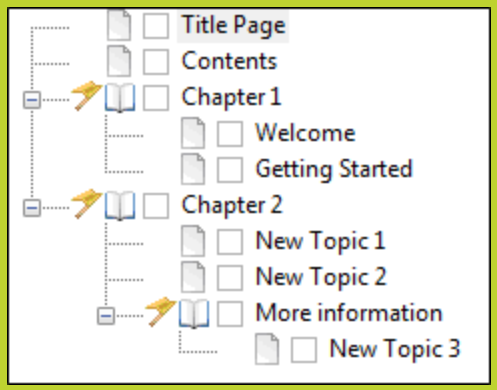
When creating print output, you can determine which topic headings will be displayed at which level in the generated TOC. For example, even though all of your topics may use a heading style of <h1>, you might want some of those topic headings to display at the first level in the print TOC (acting as an <h1>), others at the second level (acting as an <h2>), and still others at the third level (acting as an <h3>).

You can base the heading levels on the structure you create in the outline TOC. If a TOC entry is at the extreme left of the TOC Editor, it will display at the first level of the print TOC; if an entry is indented to the right one level, it will display at the second level; and so on. Not only will those headings be reflected in the generated TOC in the output, but the corresponding headings will be converted in the main body too (e.g., if a heading becomes an <h2> in the TOC, it will become an <h2> in the text as well). See "Using the TOC Depth for Heading Levels in Print Output" on page 189.

If you create books in the outline TOC (whether they are linked to topics or not), they will affect the heading level in the print TOC. If you want unlinked books to be converted to headings in the output (both in the print TOC and in the main body), you can do so.

E X A M P L E

Let's say the first paragraph in each topic that you create uses the <h1> style (except your title page and print TOC topic), and you structure the outline TOC like this:



If you use the TOC depth feature, the print TOC will look like this in the output:

Welcome 2

Getting Started 3

NewTopic1 4

NewTopic2 5

NewTopic3 5

Notice that "NewTopic3" is indented to the third level in the print TOC (because that is its position in the outline TOC). The other topics are at the second level. The books ("Chapter 1" and "Chapter 2") are at the first level, but they are not linked to topics; therefore, they are not included in the output.

If you use the feature to include the unlinked books, the print TOC will look like this in the output:

Chapter 1 2

Welcome 2

Getting Started 3

Chapter 2 4

NewTopic1 4

NewTopic2 5

More Information 6

NewTopic3 6

HOW TO CREATE HEADINGS FOR UNLINKED BOOKS IN A GENERATED TOC

1. Open the target.
2. In the Target Editor, select the **Advanced** tab.
3. In the **Generated TOC** section, select **Inject headings for unlinked books in TOC**.
4. Click to save your work.

Removing Images from Headings in a Generated TOC

This feature is supported in all print outputs (Adobe FrameMaker, Adobe PDF, Microsoft Word, Microsoft XPS, and XHTML).

If you have inserted images into the headings in your topics, you can make sure those images do not appear in the generated TOC of the printed output.

This option works somewhat differently for Word output than it does for the other outputs.

WORD

- If the option is disabled, images appear in both content headings and the print TOC.
- If the option is enabled, images are removed from both content headings and the print TOC.

PDF, XHTML, XPS, FRAMEMAKER

- If the option is disabled, images appear in content headings only.
- If the option is enabled, images are removed completely.

HOW TO REMOVE IMAGES FROM HEADINGS IN A GENERATED TOC

1. Open the target.
2. In the Target Editor, select the **Advanced** tab.
3. In the **Generated TOC** section, select **Remove images from paragraphs with heading levels**.
4. Click to save your work.

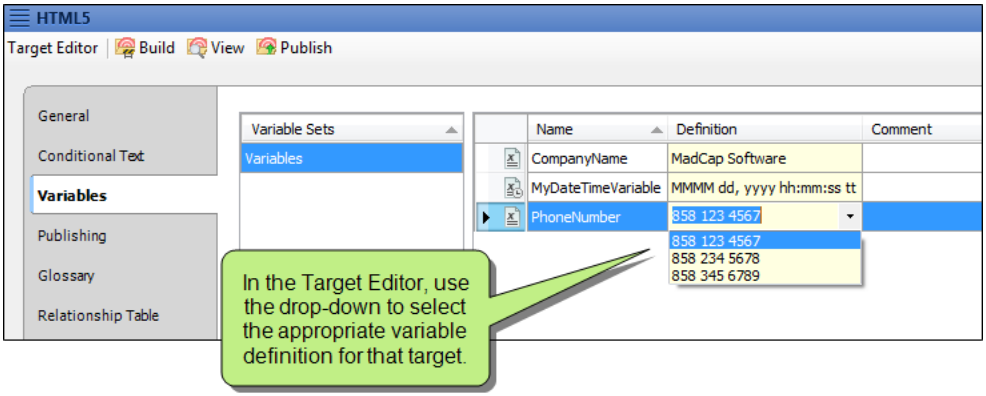
Note: Keep in mind that, with this feature, the images are also removed from the heading in the content of the document. Therefore, if you want to have the images in the content, but not in the print TOC, you should not use this feature. Instead, you must remove the images from the print TOC manually after generation (if you compile an editable output, such as Word or FrameMaker).

Overriding Variable Definitions in Targets

The variables that you create and define in the Variable Set Editor are available to your entire project. However, if you want the definition for a variable to be different in a particular target, you can override the project-level definition for that target in the Target Editor.

HOW TO OVERRIDE A VARIABLE DEFINITION IN A TARGET

1. Make sure the Project Organizer is open.
2. Double-click the **Targets** folder. The available targets are shown.
3. Double-click the target for which you want to override a variable definition. The Target Editor opens to the right, with tabs that you can use to customize that specific target.
4. Click the **Variables** tab.The Variable tab shows all of the variable sets in your project and their associated variables.
5. Select the appropriate variable set.
6. Do one of the following, depending on whether the variable has only one or multiple definitions in the Variable Set Editor. For information on adding multiple definitions, see the online Help.
   - **Single Definition** Click in the **Definition** cell for the variable that you want to override and press **F2** on your keyboard. Then type a different definition and press **Enter** on your keyboard.
   - **Multiple Definitions** In the **Definition** cell, click the down arrow and select an alternative definition.



7. Click to save your work. The alternative variable definition will be used in the output for this target. However, for other targets in your project, the original variable definition will be used.

Note: If you preview a topic, the variable definition associated with the primary target is displayed.

Note: If you click the **Edit Set** button at the bottom of the tab, the Variable Set Editor opens. When you use that editor, you are editing variables for all targets, not those for a specific target.

Specifying WebHelp AIR Options

If you are generating WebHelp AIR output, there are several options that you can set in order to affect the output.

HOW TO SPECIFY WEBHELP AIR OPTIONS

1. Open a target that is using the WebHelp AIR format.

   If necessary, you can change the output type for a target. See "Changing the Output Type for a Target" on page 123.

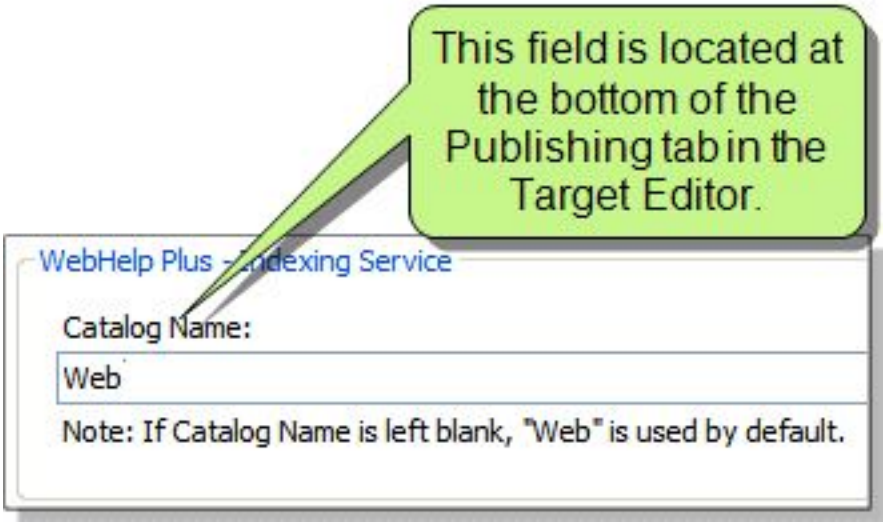
2. Select the **Adobe AIR Options** tab.
3. Complete the fields in the dialog.
   - **Program Name** The WebHelp AIR format creates a file with an .air extension. When this file is accessed, an executable file with an .exe extension is created and placed on the user's hard drive. By default, the executable file will be named after the target that you are generating. However, if you want the output to use a different file name, you can enter it in this field.
   - **ID** You can enter an identifier for the application, which uniquely identifies your application along with the publisher ID (which AIR derives from the certificate used to sign the application package). You cannot use spaces in this field.
   - **Copyright** You can enter a copyright in this field.
   - **Digital Certificate** If your company has a digital certificate, you can click the browse button next to this field and select it. By using a digital certificate, the installation will not display "unknown" for the name of the publisher when the user installs the application (i.e., opens the online output).
   - **Certificate Password** Enter the password for the digital certificate.
   - **Program Title** You can enter a title for the online output. When a user installs the application to open the online output, this title will be displayed in one of the installation windows. It is also displayed when you uninstall the application.
   - **Version** You can use this field to enter a version number for your online output.
   - **Description** You can enter a description for your online output. When a user installs the application to open the online output, this description will be displayed in one of the installation windows, under the title (if you specify one).
   - **Install Folder** By default, when a user installs the application to view your online output, the executable file is placed in their c:\Program Files directory, within a subfolder named after the target. If you want the EXE file to be placed in a subfolder with a different name, you can enter it in this field.
   - **Program Menu Folder** You can enter a folder name, which dictates where the program menu option displays in the Start menu. If you do not enter a folder name, the program option is displayed at the root level of the Start menu.
4. Click to save your work.

Setting Up a WebHelp Plus Target

The following steps show you how to set up your WebHelp Plus target for server-based output.

HOW TO SET UP THE WEBHELP PLUS TARGET

1. In Flare, open the WebHelp Plus target. The Target Editor opens.
2. Select the **Publishing** tab.
3. In the **Indexing Service Catalog Name** field, type the catalog that you are using for the output. In most cases, this will be `Web`, which is the default value. However, if you or someone in your company (e.g., network administrator) creates a custom catalog, you need to enter that name in the field.



What is a catalog? Microsoft Indexing Service stores all of its index information in catalogs. A catalog comprises index information and stored properties for a particular group of file system directories. When Indexing Service is installed with Windows XP, it automatically builds a catalog, called the System catalog, listing contents of all permanently attached disk drives. The System catalog contains an index for all documents except certain system and temporary files. If Internet Information Services (IIS) is installed, the Indexing Service also creates a web catalog, which contains an index of IIS, the default virtual server of the World Wide Web.

Note: This step is necessary only for Windows XP and Windows Server 2003.

4. Click to save your work.
5. Generate the target.
6. Publish the output to any location on the web server. If you want to use the default location that was created after you installed IIS, you can publish the output to C:\Inetpub\wwwroot. If you do this, you can use the "Default Web Site" folder that you will encounter when performing the next set of steps (i.e., configuring IIS on the production server).

   If you are working on your local machine for testing purposes, you do not need to publish the output. You can simply view your WebHelp Plus output. When you view WebHelp Plus output on your local computer, Flare creates a special folder called "MCPreview" within your "C:\Inetpub\wwwroot" folder, and it places a copy of your WebHelp Plus output files in it. This enables you to test the advanced features of WebHelp Plus on your local machine. (For Windows Vista users testing locally, the physical location is C:\MadCap\[Name of Project]\[Name of Main Entry File].htm.)

Enabling Mirror Margins in Word Output

Let's say you create a page layout with mirror margins on pages (e.g., the odd pages are set up to have a left margin of 2 inches and a right margin of 1 inch, whereas the even pages are set up to have a left margin of 1 inch and a right margin of 2 inches). In such a case, Microsoft Word is unable to display the correct margins without a little help. Therefore, you need to enable mirror margins by selecting a single check box in the Target Editor. When you generate Word output, you will then see the appropriate margins on each page.

HOW TO ENABLE MIRROR MARGINS IN WORD OUTPUT

1. Open the target being used for Word output.
2. Select the **Advanced** tab.
3. Click the check box next to **Generate "Mirror Margins" for MS Word Output**.
4. Click to save your work.

Specifying XPS Output Via Microsoft Word

Microsoft's XML Paper Specification (XPS) is a document format with a markup language that is a subset of XAML for Windows Presentation Foundation. XPS is an alternative to Adobe's Portable Document Format (PDF).

You can generate XPS output from your project directly. Make sure you have the latest version of Microsoft .NET Framework installed on your computer. This is a free download from **microsoft.com**.

Alternatively, you can generate XPS output while simultaneously building Word 2007 output (by installing a free add-in download from Microsoft).

The following describes how to use the second option, sending output to XPS via Microsoft Word. If you want to send output to XPS directly, simply install the latest version of Microsoft .NET Framework and build an XPS target.

Requirements

In order to send output to XPS via Microsoft Word, you need to have the following setup.

- You must have Microsoft Word 2007 installed. Previous versions of Word do not support this functionality.
- You must download and install the following free add-ins from Microsoft.
  - **To View and Generate XPS** Go to **http://www.microsoft.com/whdc/xps/viewxps.mspx**

    Note: This first add-in is not necessary for Microsoft Vista users.

  - **To Save as XPS** Go to **http://www.microsoft.com/downloads/details.aspx?familyid=4D951911-3E7E-4AE6-B059-A2E79ED87041&displaylang=en**

HOW TO SPECIFY XPS OUTPUT FOR MICROSOFT WORD

1. Open a Word target.
2. In the Target Editor, do one of the following.

   ##### *USE THE OUTPUT FILE FIELD ON THE GENERAL TAB*

   a. Select the **General** tab.

   b. In the **Output File** field, enter the name that you want to use for the main output file and add the .xps extension at the end of it. For example, if you want your main output file to be called "MyNewManual," you would enter the following in this field: `MyNewManual.xps`

   OR

   ##### *USE THE CHECK BOX ON THE ADVANCED TAB*

   a. Select the **Advanced** tab.

   b. Click the check box next to **Generate XPS for MS Word Ouput**.

   > **Note:** If you use this second check box option, it will overrule any file extension specified for the output file name on the General tab.

3. Click to save your work.

Ignoring Warnings When Building Output

This feature is supported in all outputs.

You can tell Flare to ignore certain warnings so that they do not display in the Build Progress dialog when generating output. This can be done globally for all targets in all projects. You can also override those global settings for a specific target in a project.

By default, Flare will use the global settings that are specified in the Options dialog. So if you want all of your projects and targets to have the same settings for ignoring warnings, it's best to just use the Options dialog. However, if you want the settings to be unique for a particular target, you can override the global settings by using the Target Editor.

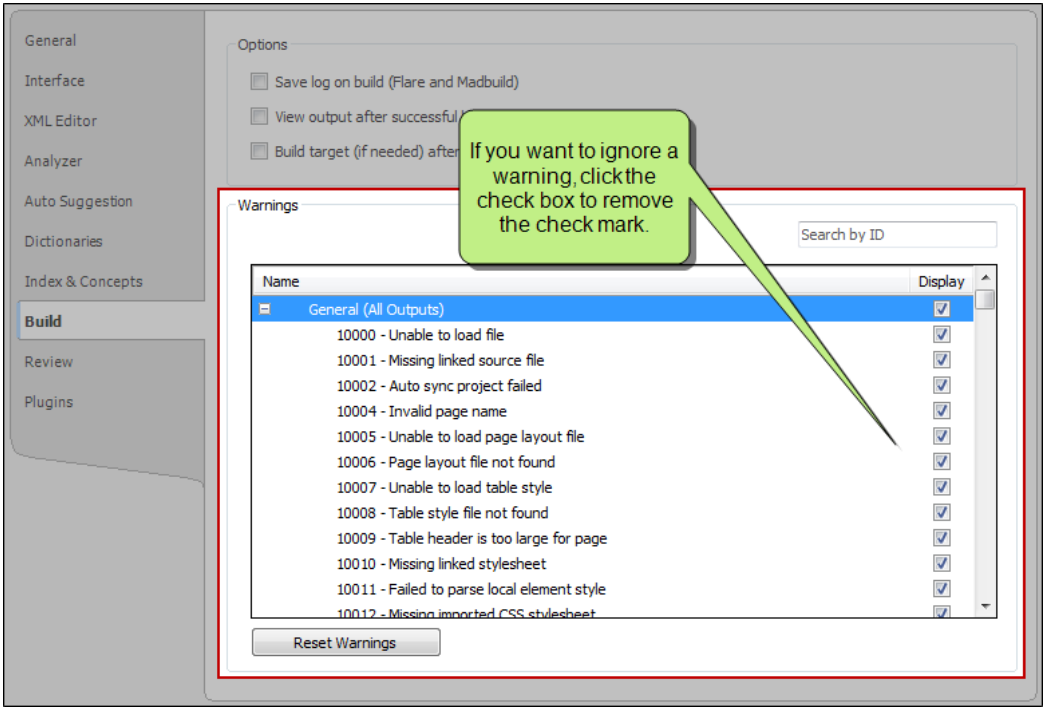
HOW TO IGNORE WARNINGS GLOBALLY

1. Do one of the following, depending on the part of the user interface you are using:
   - **Ribbon** Select **File>Options**.

     You can use the Options dialog to switch between ribbons and the classic tool strip layout. For more information see the online Help.

     Keep in mind that the smaller the application window becomes, the more the options in a ribbon shrink. Therefore, you might only see a small icon instead of text, or you might see only a section name displayed with a down arrow to access the options in it. You can hover over small icons to see tooltips that describe them. You can also enlarge the application window or click one of the section drop-downs in the ribbon to locate a hidden feature.
   - **Menu** Select **Tools>Options**.

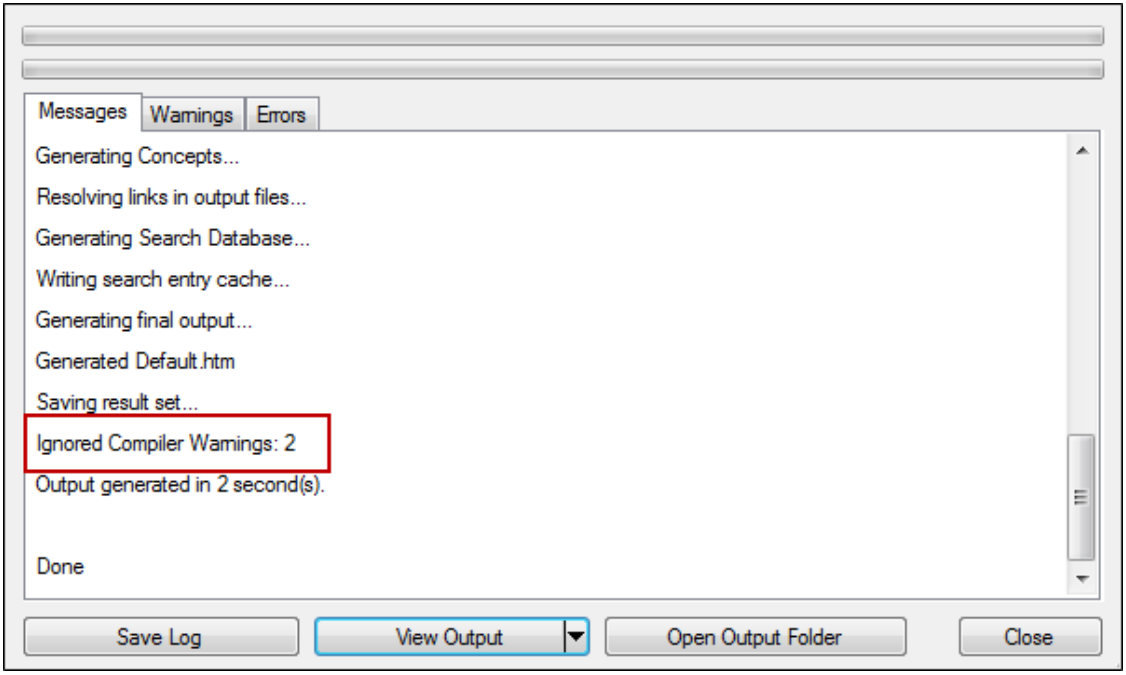
   The Options dialog opens.
2. Select the **Build** tab.
3. In the **Warnings** section of the tab, you can remove check marks next to any warning that you want Flare to ignore. By default, all check boxes are initially selected.



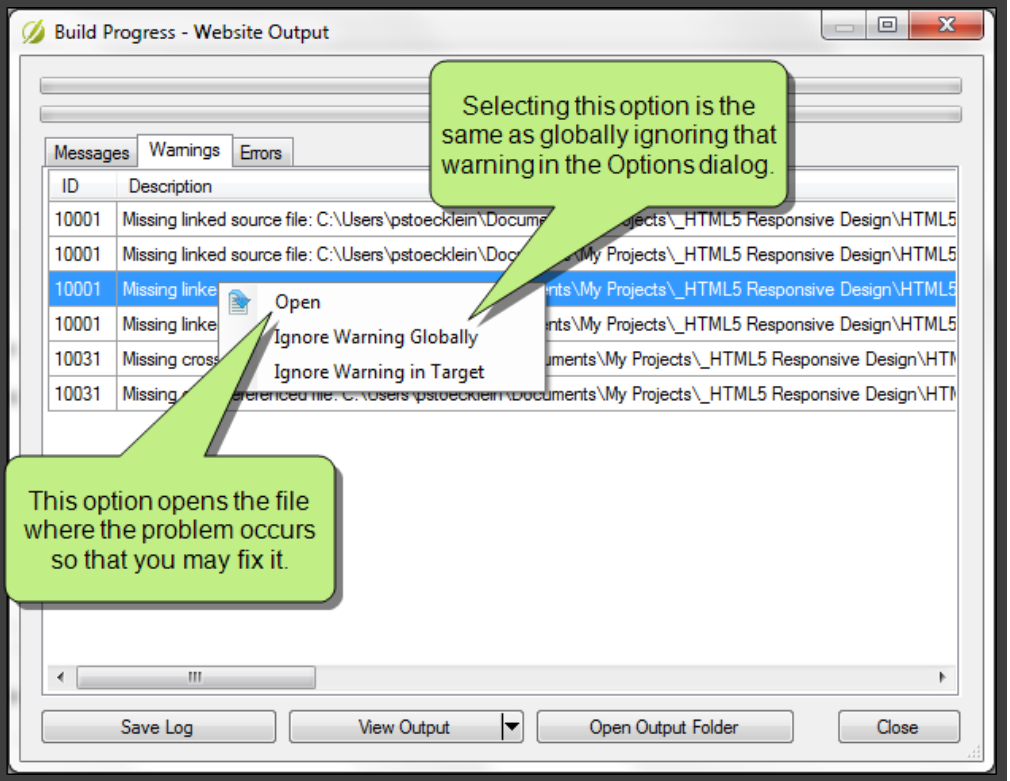
If you want to return to the default settings where all of the check boxes are selected, you can click **Reset Warnings** at the bottom of the tab.

4. Click **OK**.

The Messages tab of the Build Progress dialog indicates how many warnings were ignored during the build.

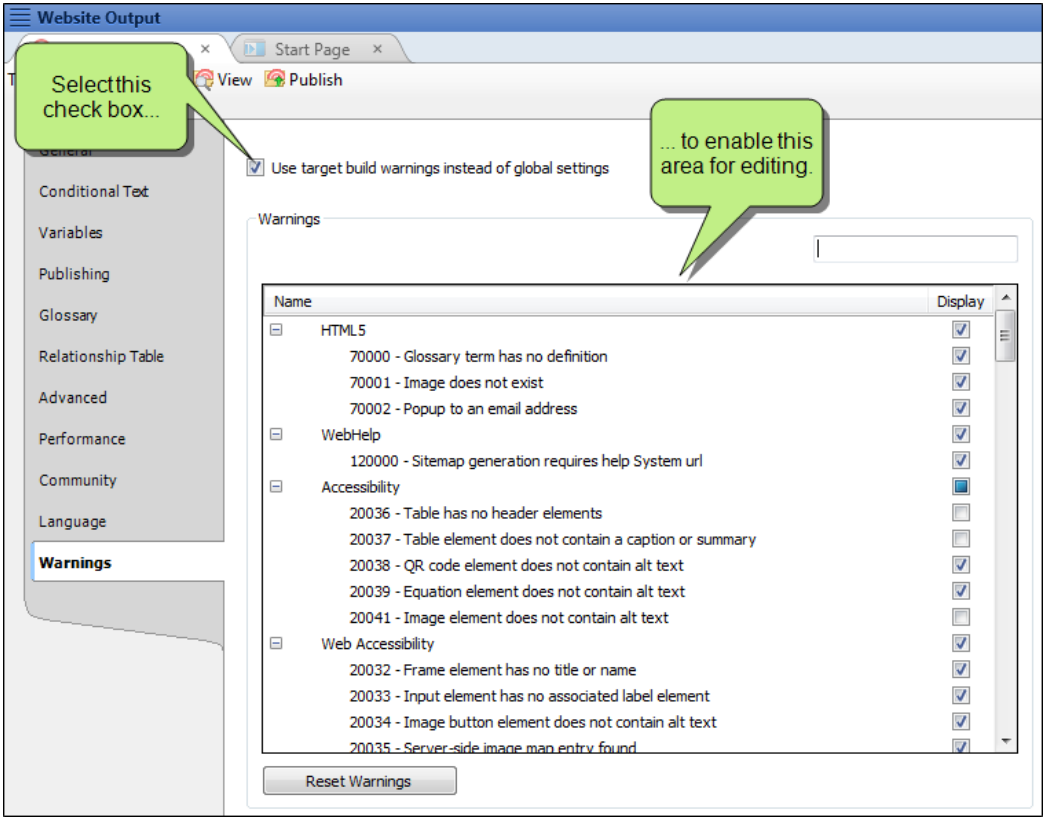

Messages
Warnings
Errors
Generating Concepts...
Resolving links in output files...
Generating Search Database...
Writing search entry cache...
Generating final output...
Generated Default.htm
Saving result set...
Ignored Compiler Warnings: 2
Output generated in 2 second(s).
Done
Save Log
View Output
Open Output Folder
Close

Note: In the Build Progress dialog after a target has been generated, you can select the **Warnings** tab, right-click on any warning message, and select **Ignore Warning Globally**. Doing this removes the check mark for that warning from the Options dialog. (You can also select an option to open the file where the problem occurs.)



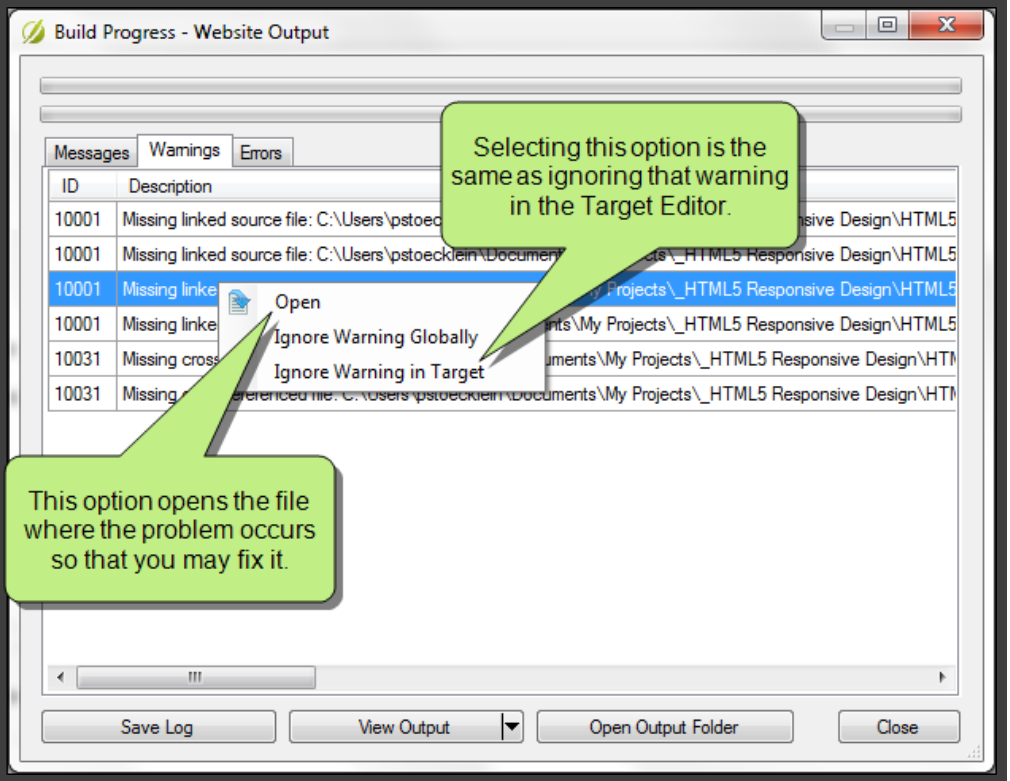
HOW TO IGNORE WARNINGS FOR A TARGET

1. Open a target.
2. Select the **Warnings** tab.
3. Click **Use target build warnings instead of global settings**. This enables the area below with all of the check boxes.
4. You can remove check marks next to any warning that you want Flare to ignore for that target. By default, all check boxes are initially selected.



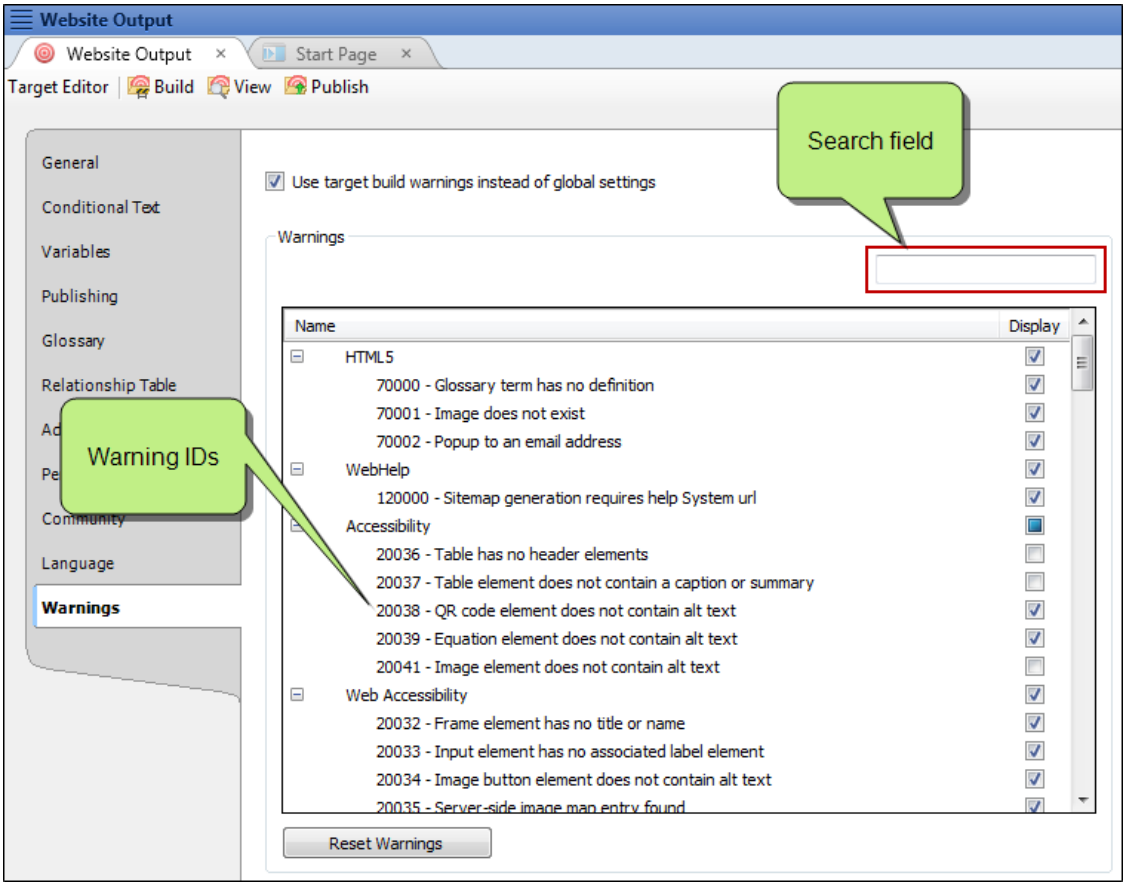
5. Click to save your work.

Note: In the Build Progress dialog after a target has been generated, you can select the **Warnings** tab, right-click on any warning message, and select **Ignore Warning in Target**. Doing this removes the check mark for that warning from the Target Editor. (You can also select an option to open the file where the problem occurs.)



SEARCH BY ID

Each type of warning has a numerical ID to the left of it. You can use the search field to quickly jump to a particular warning.



Warning: It is recommended that you be careful with which compiler warnings you choose to ignore. Only ignore warnings that you are certain you do not need to receive.

Note: You can only ignore warnings; you cannot ignore errors, which prevent output from being generated.

Note: If you open a target that contains old accessibility warning settings from previous versions of Flare, that target file will become "dirty" (i.e., an asterisk is shown next to the file name in the Target Editor) and you will see a message indicating that Flare is updating the file to use the new report warning system. You can then simply save the target file with the new settings. This does not affect your target in any other way.

CHAPTER 9

Building Output

After you have completed the first part of the development process (i.e., creating and designing content, developing targets), you are ready to build the final output. Of course, you can build the output at any point during the development process, but if you make additional changes to content, targets, or the look and feel, you will need to build the output again to make sure the changes are included in the files that you deliver to your end users.

This chapter discusses the following:

Ways to Build Output 214
Build Progress Dialogs 215
Where Output Files are Stored 216
Building the Primary Target 217
Building a Single Target 219
Building Output Using a Batch Target 220

Ways to Build Output

Building the final output involves generating one or more targets in your project, usually with just the click of a button or two. Following are the different ways you can build your final output.

- **Primary Target** Do this if you are only concerned about building the primary target for your project. See "Building the Primary Target" on page 217.
- **Single Target** Do this if you want to build a target that is not designated as your primary target. See "Building a Single Target" on page 219.
- **Batch Target** Do this if you want to build and/or publish one or multiple targets in a batch from the user interface, perhaps scheduled to run at a specific time. See "Building Output Using a Batch Target" on page 220.
- **Command Line/Batch** Do this if you want to build targets from your operating system's command line. Using this method, you do not have to open Flare at all. In addition, this method allows you to build a single target or all targets in your Flare project in one batch. The best way to use this feature is to create a batch file with the necessary commands in it. Then you can use a scheduling tool (such as the "Scheduled Tasks" utility in Windows) to run the batch file automatically whenever you want. This is similar to the "batch target" feature. However, the command line feature works outside of the Flare interface, is a bit more manual, and does not support as many processes. For steps see the online Help.
- **Build Then View Automatically** If you select an option in the user interface to view output from a target, Flare shows that output if it is up to date. But if there is no output or it is out of date, Flare gives you a message, asking if you want to build the output. This feature lets you avoid seeing this message for all targets in all projects. You can select an option to avoid seeing this message for all targets in all projects. This can be done in the Options dialog (**File>Options** or **Tools>Options**) by selecting the **Build** tab and choosing **Build target (if needed) after selecting 'View'**.

Warning: If your computer's anti-virus (AV) software scans the Output folder during the build process and you experience one or more "Build Failed" errors, the AV software may be preventing the build process from accessing the output files. To avoid this, you can temporarily turn off or disable your AV scan. You may also be able to exclude the Output folder, add Flare to an exceptions list, or postpone/reschedule scans to a different time. See your AV vendor for information.

Build Progress Dialogs

When you build targets, a Build Progress dialog shows the progress, including warnings or errors that may occur. This dialog contains the following tabs:

- **Messages Tab** This shows compiler messages and warning text. It is essentially a running log of what is happening as the output is generated.
- **Warnings Tab** This shows warnings reported by the compiler. Warnings do not prevent the output from being generated. Instead they might be issues that you need to fix (e.g., missing files) or suggestions for improving your output (e.g., accessibility settings). You can right-click on a warning and tell Flare to ignore it in future builds.
- **Errors Tab** This shows errors, which are critical problems that stop the project from building output.

When a target finishes building, you can do the following from this dialog:

- **Open Files With Warnings** In the Warnings tab, you can see information about each issue. If you double-click a row, the file that is shown in the File column is opened so you can deal with the issue.
- **Ignore Warnings** You can tell Flare to ignore certain warnings so that they do not display in the Build Progress dialog when generating output. You can do this in the Build Progress dialog by right-clicking on any warning message and selecting **Ignore Warning Globally** or **Ignore Warning in Target**. You can also ignore warnings from the Options dialog or the Target Editor. See "Ignoring Warnings When Building Output" on page 204.

  > **Warning:** It is recommended that you be careful with which compiler warnings you choose to ignore. Only ignore warnings that you are certain you do not need to receive.

- **Save Log** This saves the log file after an output has been generated so that you can refer to it later. The log file is stored in the Reports folder of the Project Organizer.

  You also have the option to do this automatically at a global level for all targets in all projects. This can be done in the Options dialog (**File>Options** or **Tools>Options**) by selecting the **Build** tab and choosing **Save log on build**.
- **View Output** This opens the generated output. You can also click the down arrow next to this button and select a specific browser or device to display the output. If you generated output that is enabled with responsive design, the Build Progress dialog includes view options not only for seeing output on a full browser output, but also on a tablet or smart phone. See "Viewing Output" on page 233.
- **Open Output Folder** This opens the project Output folder in Windows so that you can see the output subfolders and files.

Where Output Files are Stored

When you build a target, Flare creates output files and places them in a folder named after the target, which is stored in a subfolder of your project called "Output." For example, let's say your project is stored here: C:\MyProject. In that case, after you generate output, the files would be stored here:

C:\MyProject\Output\MyName\TargetName

Depending on the output type associated with the target, the generated output might consist of many files.

Building the Primary Target

Use the following steps if you want to quickly build the primary target for your project.

What do we mean by this? It has to do with the final output of your project. When all is said and done with your project, you want your output to look and behave in a certain way. But not all authors have the same audience or work for the same company with the same needs and the same computer networking setup. Therefore, Flare lets you choose the best type of output for your needs.

Now, in your project, you can use one or more of Flare's output types. You can even make copies of one output type and make changes to it. Each *instance* of an output type is called a "target."

This concept of targets may seem a bit confusing if you are new to Flare. But don't worry. It will become much more clear as you become more familiar with Flare.

So what is the big deal with the primary target? It is simply about convenience. Although you can use more than one target for your project—you will most likely use just one target most of the time. So Flare lets you designate one target as your primary target and gives you shortcuts to perform certain tasks with it, including building the output from the primary target.

HOW TO BUILD THE PRIMARY TARGET

1. Do one of the following, depending on the part of the user interface you are using:
   - **Ribbon** Select the **Project** ribbon. In the **Content** section select the face of the **Build Primary** button.

     You can use the Options dialog to switch between ribbons and the classic tool strip layout. For more information see the online Help.

     Keep in mind that the smaller the application window becomes, the more the options in a ribbon shrink. Therefore, you might only see a small icon instead of text, or you might see only a section name displayed with a down arrow to access the options in it. You can hover over small icons to see tooltips that describe them. You can also enlarge the application window or click one of the section drop-downs in the ribbon to locate a hidden feature.
   - **Menu** Select **Build>Build Primary**.
   - **Project Toolbar** Click the face of Build Primary.

     To open the Project toolbar from the menu view, select **View>Toolbars>Project**.
   - **Local Toolbar** In the local toolbar of the Target Editor click Build.
   - **Keyboard Shortcut** Press **F6**.

2. After the output files finish generating, a message asks if you want to view the output. Click either **Yes** or **No**. If you select Yes, the generated output opens.
3. (Optional) If any errors or warnings occur during the process, you can save the report file by clicking **Save Log** in the Build Progress dialog. You can then open the error report file from the Reports folder in the Project Organizer. You can also ignore warnings that you do not want to see anymore (see "Ignoring Warnings When Building Output" on page 204).
4. (Optional) From the Build Progress dialog, you can view the output by clicking **View Output**.
5. (Optional) If you want to open the Windows folder holding the output files that were generated, click **Open Output Folder** in the Build Progress dialog.

Warning: If your computer's anti-virus (AV) software scans the Output folder during the build process and you experience one or more "Build Failed" errors, the AV software may be preventing the build process from accessing the output files. To avoid this, you can temporarily turn off or disable your AV scan. You may also be able to exclude the Output folder, add Flare to an exceptions list, or postpone/reschedule scans to a different time. See your AV vendor for information.

Building a Single Target

Use the following steps if you want to build a single target in your project that is not designated as your primary target.

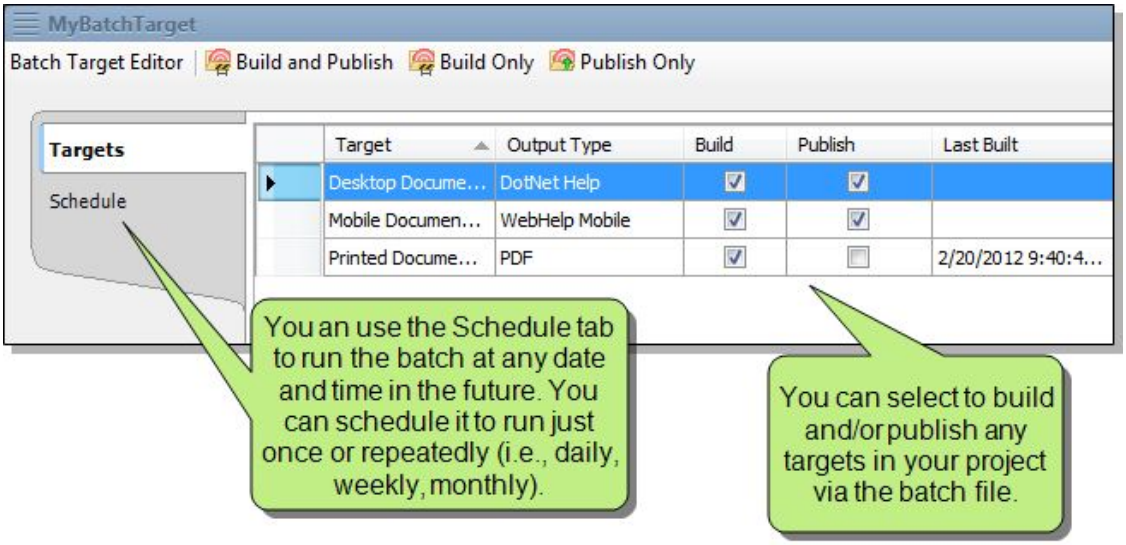
HOW TO BUILD A TARGET

1. Do one of the following, depending on the part of the user interface you are using:
   - **Ribbon** Select the **Project** ribbon. In the **Build** section select **Build Primary>Build [name of target]**.
   - **Project Toolbar** Click the down arrow next to Build Primary and from the menu select the name of the target.

     To open the Project toolbar from the menu view, select **View>Toolbars>Project**.
   - **Local Toolbar** In the local toolbar of the Target Editor click Build.
   - **Right-Click** In the Project Organizer, open the Targets folder, right-click on a target, and from the context menu select **Build [name of target]**.
2. After the output files finish generating, a message asks if you want to view the output. Click either **Yes** or **No**. If you select Yes, the generated output opens.
3. (Optional) If any errors or warnings occur during the process, you can save the report file by clicking **Save Log** in the Build Progress dialog. You can then open the error report file from the Reports folder in the Project Organizer. You can also ignore warnings that you do not want to see anymore by right-clicking on them in the Warnings tab.
4. (Optional) From the Build Progress dialog, you can view the output by clicking **View Output**.
5. (Optional) If you want to open the Windows folder holding the output files that were generated, click **Open Output Folder** in the Build Progress dialog.

Building Output Using a Batch Target

Use the following steps if you want to generate and/or publish one or multiple targets in a batch from the user interface, perhaps scheduled to run at a specific time.



E X A M P L E

Let's say you have a project with 17 targets and you need to build and publish 12 of those targets at the beginning of each day. Rather than manually building and generating each of these targets, you can use a batch target. Within this batch target, you can select the "Build" and "Publish" options next to each of the 12 targets. Furthermore, you can use the Schedule tab in the batch target to specify that the batch should be run each night at 2 a.m. When you arrive at the office each morning, the outputs for those 12 targets will already be generated and placed in the appropriate publishing destination folders.

Batch Generate UI/Publish Versus Command Line

The batch target in the Flare interface is similar to the "command line" feature. However, the command line feature works outside of the Flare interface, is a bit more manual, and does not support as many processes.

Tasks Associated with Using a Batch Target

Following are the basic tasks associated with using the batch target feature to build and/or publish output.

- "Create a Batch Target" on the next page
- "(Optional) Schedule Build or Publish Processes" on page 224
- "(Optional) Manually Start Build or Publish Processes" on page 227

Create a Batch Target

First, you must create a batch target. The batch target is a simple file that points to other targets and stores information such as whether to build or publish targets, as well as scheduling commands. After creating the file, you can specify its settings in the Batch Target Editor. A batch target file has an .flbat extension and is stored in the Project Organizer under the Targets folder.

HOW TO CREATE A BATCH TARGET

1. Do one of the following, depending on the part of the user interface you are using:
   - **Ribbon** Select the **Project** ribbon. In the **Content** section select **New>Batch Target**.
   - **Menu** Select **Project>Add Batch Target**.
   - **Right-Click** In the Project Organizer, right-click on the **Targets** folder and from the context menu select **Add Batch Target**.

   The Add File dialog opens.
2. In the **File Type** field at the top, make sure **Batch Target** is selected.
3. In the **Source** area select one of the following:
   - **New from template** This lets you choose either the factory template file or one of your own customized template files as a starting point. The new file will take on all of the settings contained in the template. If you want to use the factory template provided by Flare, expand the **Factory Templates** folder and click on a template file. If you want to use your own customized template file, expand the appropriate folder and click on a file. For more information about templates, see the online Help.

     > **Note:** In some dialogs and wizards you can click the **Manage Templates** button if you want to open the Template Manager. This lets you manage any of your template files (e.g., add new templates, enter descriptions for templates). For more information see the online Help.

   - **New from existing** This lets you choose an existing file of the same type as a starting point for your new file. As with template files, your new file will take on all of the settings contained in the file you select. To use this option, click the browse button ..., use the Open File dialog to find a file, and double-click it.
4. (Optional) If you want to place the file into a subfolder that you previously created in the Project Organizer, in the **Folder** field click ... and select the subfolder. Otherwise, keep the default location.
5. In the **File Name** field, type a new name for the batch target.
6. Click **Add**. The target is added to the Targets folder in the Project Organizer.
7. A message asks if you want to create a scheduled task. Click **Yes**. The Batch Target Editor opens to the right.

8. The Targets tab in the editor displays all of the other targets already contained in your project. Click the **Build** and/or **Publish** check boxes next to the targets that you want to be affected when the batch runs.

> **Note:** In order for targets to be published when the batch runs, you must also create and associate a publishing destination with the target(s).

9. Click to save your work.

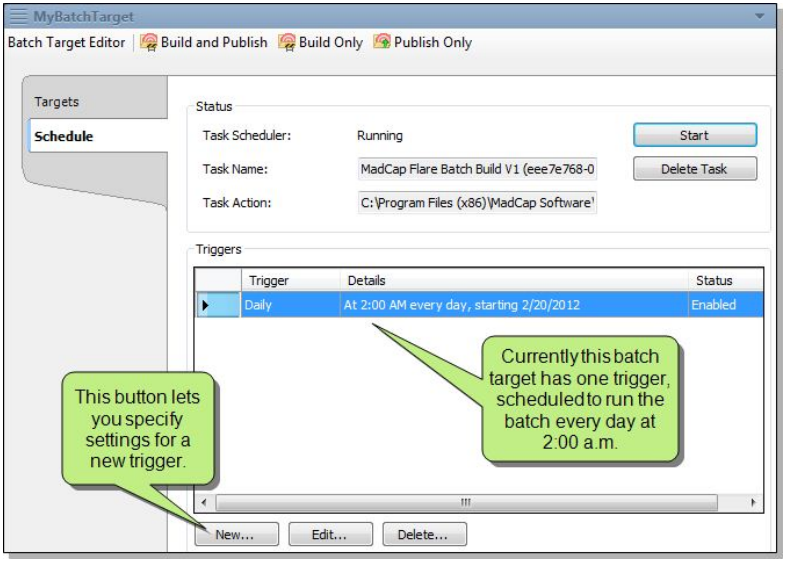
(Optional) Schedule Build or Publish Processes

After you create a batch target in the Flare user interface, you can start that batch whenever you need to (i.e., tell it to start building and/or publishing the related targets). However, you also have the option of creating scheduled tasks. You might do this if you want your targets to be generated or published automatically overnight. Scheduled builds are created using the Windows Task Scheduler. However, the user interface in Flare makes it easier for you to create scheduled tasks without leaving the application. If you use this scheduling feature, you do not need to have Flare open at the time the batch runs.

When a scheduled task runs, the command prompt window opens on your computer and minimizes while the batch runs. This window closes automatically when the batch process is finished. If any errors or warnings occur during the process, a report is automatically saved so you can review the messages. You can then open the error report file from the Reports folder in the Project Organizer.

HOW TO SCHEDULE BUILD OR PUBLISH PROCESSES

1. From the **Targets** folder in the Project Organizer, open the batch target file.
2. If a message indicates that you must create a scheduled task, click **Yes**.
3. In the Batch Target Editor, click the **Schedule** tab.
4. At the bottom of the tab, click **New**.



The New Trigger dialog opens.

5. In the **Settings** area, select how often you want the trigger in the batch to be run. You can also click in the **Start** fields to change the beginning date and/or time:
   - If you select **One Time**, click in the **Start** fields to change the beginning date and/or time.
   - If you select **Daily**, a field displays so that you can specify a certain number of days for the process to recur.
   - If you select **Weekly**, a field displays so that you can specify a certain number of weeks for the process to recur. In addition, check boxes are available so that you can select certain days of the week for the process to run.
   - If you select **Monthly**, additional fields are displayed so that you can select certain months for the process to run, even on specific days during particular months.
6. (Optional) In the **Advanced Settings** area, you can set any of the following.
   - **Repeat task every** You can specify if you want the trigger for the batch to run periodically after a certain number of minutes or hours.
   - **For a duration of** You can specify how long you want the trigger for the batch to be repeated.
   - **Expire** You can specify whether the trigger for the batch should stop running after a certain date.
   - **Enabled** By default the trigger for the batch will be enabled. However, you can disable the trigger if necessary. The trigger will remain in the batch file (even though it will not run while being disabled). You can re-enable it in the future if you want.
7. Click **OK**.

Tip: Because scheduled tasks in batch targets use Windows Task Scheduler, the settings in that utility are applied. By default, scheduled tasks will run only if you are logged on. However, you can change this setting in Windows Task Scheduler so that the batch runs whether you are logged on or not. To accomplish this, first open Windows Task Scheduler. The steps may be different depending on the operating system. For example, in Windows 7 click the **Start** button and select **All Programs>Accessories>System Tools>Task Scheduler**. Then select **Task Scheduler Library** to see a list of your scheduled tasks. In the list select the appropriate task and click **Properties**. Click **Run whether user is logged on or not** (in Windows 7 this is on the General tab). Click **OK**. You will be prompted to enter your Windows login password. When you are finished working for the day, log off your computer (instead of shutting down completely). The task will run as scheduled.

Note: If you create a scheduled task in a batch target, a .job file is automatically created in Windows Task Scheduler. If you delete the batch target from within Flare, the .job file is automatically removed from Windows

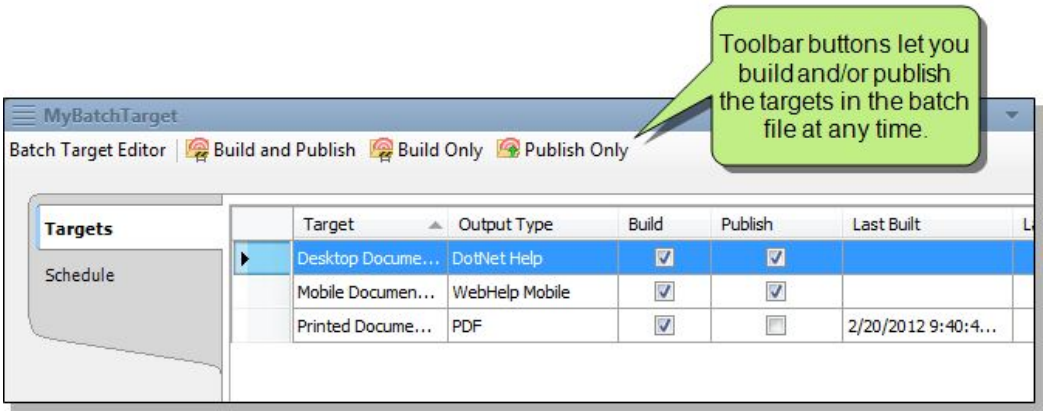
Task Scheduler. However, if you delete the project or batch target from Windows (outside of Flare), the .job file remains in Windows Task Scheduler. Therefore, you will need to remove the .job file manually from there.

(Optional) Manually Start Build or Publish Processes

Although you can schedule a batch target to run at a specific time, you can always open the batch target and manually start the batch.

HOW TO MANUALLY START BUILD OR PUBLISH PROCESSES

1. From the **Targets** folder in the Project Organizer, open the batch target file.
2. In the local toolbar of the Batch Target Editor, click any of the following.
   - Build and Publish Builds and publishes the selected targets (if they have check marks in the Build and Publish boxes in the batch target).
   - Build Only Builds the selected targets (if they have check marks in the Build box in the batch target).
   - Publish Only Publishes the selected targets (if they have check marks in the Publish boxes in the batch target).



   The Batch Progress dialog shows each target and its build status.

3. (Optional) If any errors or warnings occur during the process, you can save the report file by clicking **Save Log**. You can then open the error report file from the Reports folder in the Project Organizer. You can also ignore warnings that you do not want to see anymore (see "Ignoring Warnings When Building Output" on page 204).
4. (Optional) You can view the output of each built target—by selecting the target row in the Batch Progress dialog and clicking **View Output**.

5. (Optional) If you want to open the Windows folder holding the output files that were generated, select a target in the grid at the top of the dialog, then click **Open Output Folder**.
6. When the process finishes, click **Close** in the Build Progress dialog.

Warning: If your computer's anti-virus (AV) software scans the Output folder during the build process and you experience one or more "Build Failed" errors, the AV software may be preventing the build process from accessing the output files. To avoid this, you can temporarily turn off or disable your AV scan. You may also be able to exclude the Output folder, add Flare to an exceptions list, or postpone/reschedule scans to a different time. See your AV vendor for information.

Note: When you publish output using a batch target, changed files are replaced and stale files are always removed.

Note: You cannot use scheduled tasks for batch targets if you are working in trial mode.

CHAPTER 10

Publishing Output to Destinations

After you create a publishing destination and associate it with a target, you are ready to publish your output to the destination.

You can publish output in the following ways.

- **Primary Target** You can quickly publish your primary target.
- **Single Target** You can publish a specific single target.
- **Batch Target** You can publish (and generate) multiple targets by using a batch target. You can even create a scheduled task so that the batch runs at a specific time (e.g., nightly at 3 am). For steps on using this method, see "Building Output Using a Batch Target" on page 220.

HOW TO PUBLISH YOUR PRIMARY TARGET TO A DESTINATION

1. If your output is not up to date, rebuild it. See "Building Output" on page 213.
2. Do one of the following, depending on the part of the user interface you are using:
   - **Ribbon** Select the **Project** ribbon. In the **Content** section select the face of the **Publish Primary** button.
   - **Menu** Select **Build>Publish Primary**.
   - **Project toolbar** Click the face of Publish Primary.

     To open the Project toolbar from the menu view, select **View>Toolbars>Project**.
   - **Local Toolbar** In the local toolbar of the Target Editor click Publish.
   - **Keyboard Shortcut** Press **CTRL+F6**.
3. If necessary, change any of the options in the Publish Target dialog.
   - **Publish** Add or remove the check mark next to any destinations (depending on whether you want to publish to that destination or not).
   - **Upload Log File** Select this option if you want Flare to publish the log file with the rest of your output files. You can disable this option so that the log file is not uploaded. However, keep in mind that if you do this, you cannot use either of the following options.
     - **Upload Only Changed Files** Select this option if you want Flare to republish only the files that have changed. This can save significant time.
     - **Remove Stale Files** Select this option if you want Flare to identify files that were previously published but are no longer needed. Flare will then remove such files from the destination.

     For single-file outputs (such as Adobe PDF or Microsoft HTML Help), you might decide to disable publication of the log file. But for multi-file outputs (such as HTML5, WebHelp, DotNet Help, and others), you might prefer to use a destination file with this option enabled.
4. Click **Start Publishing**.
5. If a user name and password are required and you have not already provided them in the Destination Editor, enter them in the dialog that opens.
6. After the files are successfully published, a dialog opens, asking if you want to view the log (which lists the files that have been uploaded). Click **Yes** if you want to see it, and close the log when you are finished.
7. In the Publish Target dialog, click **Close**.

HOW TO PUBLISH A SINGLE TARGET TO A DESTINATION

1. If your output is not up to date, rebuild it. See "Building Output" on page 213.
2. Do one of the following, depending on the part of the user interface you are using:
   - **Ribbon** Select the **Project** ribbon. In the **Build** section select **Publish Primary>Publish [name of target]**.
   - **Project toolbar** Click the down arrow next to Publish Primary and from the menu select the name of the target.

     To open the Project toolbar from the menu view, select **View>Toolbars>Project**.
   - **Local Toolbar** In the local toolbar of the Target Editor click Publish.
   - **Right-Click** In the Project Organizer, open the Targets folder, right-click on a target, and from the context menu select **Publish [name of target]**.
3. If necessary, change any of the options in the Publish Target dialog.
   - **Publish** Add or remove the check mark next to any destinations (depending on whether you want to publish to that destination or not).
   - **Upload Log File** Select this option if you want Flare to publish the log file with the rest of your output files. You can disable this option so that the log file is not uploaded. However, keep in mind that if you do this, you cannot use either of the following options.
     - **Upload Only Changed Files** Select this option if you want Flare to republish only the files that have changed. This can save significant time.
     - **Remove Stale Files** Select this option if you want Flare to identify files that were previously published but are no longer needed. Flare will then remove such files from the destination.

     For single-file outputs (such as Adobe PDF or Microsoft HTML Help), you might decide to disable publication of the log file. But for multi-file outputs (such as HTML5, WebHelp, DotNet Help, and others), you might prefer to use a destination file with this option enabled.
4. Click **Start Publishing**.
5. If a user name and password are required and you have not already provided them in the Destination Editor, enter them in the dialog that opens.
6. After the files are successfully published, a dialog opens, asking if you want to view the log (which lists the files that have been uploaded). Click **Yes** if you want to see it, and close the log when you are finished.
7. In the Publish Target dialog, click **Close**.

CHAPTER 11

Viewing Output

You can view the generated output for any of the targets in your project in the following ways:

- **Primary Target** View output for the primary target.
- **Specific Target** View output for a specific target in your project.
- **After Generating** View output for a specific target immediately after generating output; depending on the type of target, you can choose a specific browser or device (such as an EPUB reader) to view the output.
- **Automatic** View outputs automatically after generating targets.
- **Build Then View Automatically** If you select an option in the user interface to view output from a target, Flare shows that output if it is up to date. But if there is no output or it is out of date, Flare gives you a message, asking if you want to build the output. This feature lets you avoid seeing this message for all targets in all projects.

HOW TO VIEW OUTPUT FOR THE PRIMARY TARGET

1. Do one of the following, depending on the part of the user interface you are using:
   - **Ribbon** Select the **Project** ribbon. In the **Content** section select the face of the **View Primary** button.
   - **Menu** Select **Build>View Primary**.
   - **Project Toolbar** Click the face of View Primary.

     To open the Project toolbar from the menu view, select **View>Toolbars>Project**.
   - **Local Toolbar** In the local toolbar of the Target Editor click View.
   - **Keyboard Shortcut** Press **SHIFT+F6**.
2. If the output for the target has not yet been generated or is out of date, a message lets you know and asks if you want to generate the output. Click **Yes**. The generated output opens.

HOW TO VIEW OUTPUT FOR A SPECIFIC TARGET

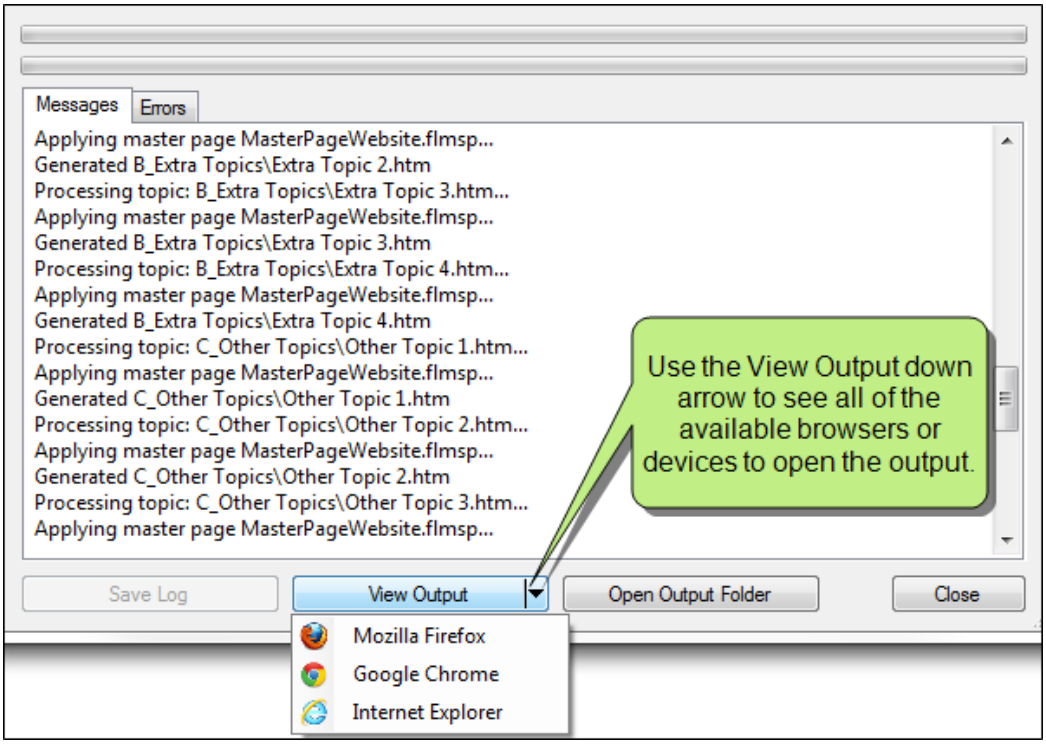
1. Do one of the following, depending on the part of the user interface you are using:
   - **Ribbon** Select the **Project** ribbon. In the **Build** section select **View Primary>View [name of target]**.
   - **Project Toolbar** Click the down arrow next to [View Primary] and from the menu select the name of the target.

     To open the Project toolbar from the menu view, select **View>Toolbars>Project**.
   - **Local Toolbar** In the local toolbar of the Target Editor click [View].
   - **Right-Click** In the Project Organizer, open the Targets folder, right-click on a target, and from the context menu select **View [name of target]**.
2. If the output for the target has not yet been generated or is out of date, a message lets you know and asks if you want to generate the output. Click **Yes**. The generated output opens.

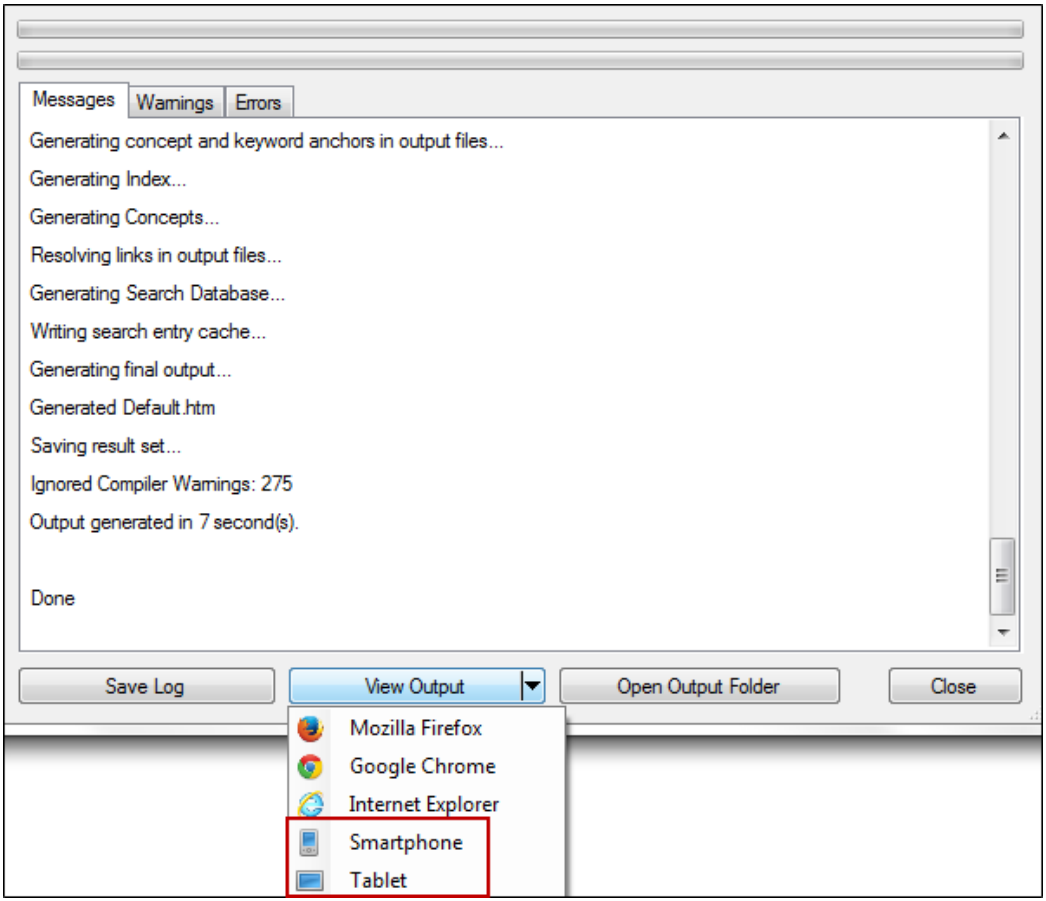
HOW TO VIEW OUTPUT IMMEDIATELY AFTER BUILDING IT

1. Generate output for the target.
2. When the build process finishes, a message asks if you want to view the output (if no errors or warnings are found). If you want to see it in the default browser or device, click **Yes**.

   If instead, you want to see it in a specific browser or device, click **No**. Then in the Build Progress dialog, click the down arrow next to the **View Output** button and select the browser or device.

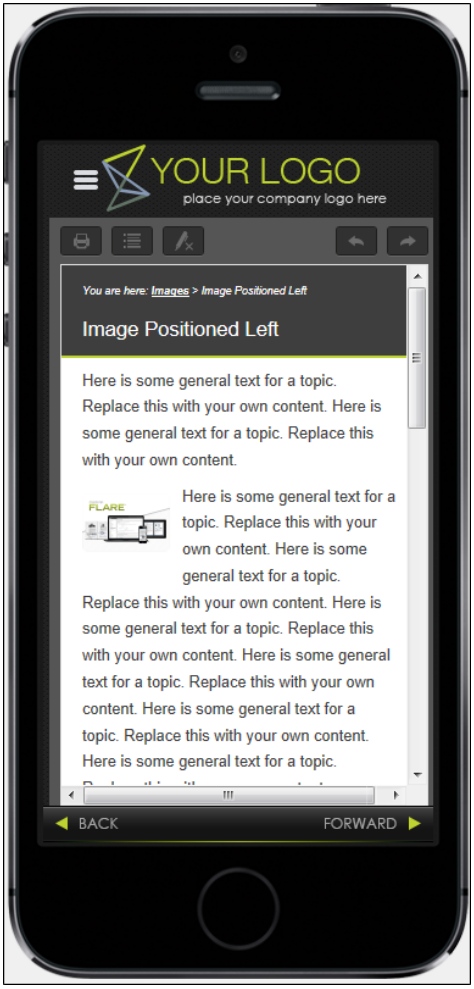


If you generated output that is enabled with responsive design, the Build Progress dialog includes view options not only for seeing output on a full browser output, but also on a tablet or smart phone.



If you select the tablet option, the view displays like this:

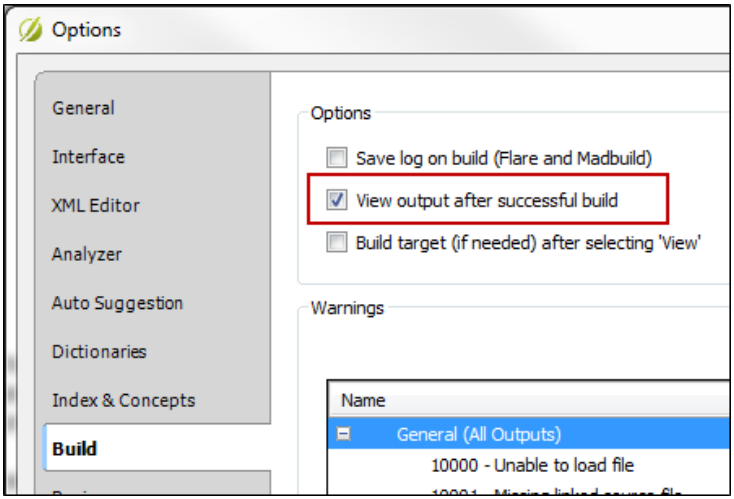
If you select the smart phone option, the view displays like this:



HOW TO VIEW OUTPUTS AUTOMATICALLY

1. Do one of the following, depending on the part of the user interface you are using:
   - **Ribbon** Select **File>Options**.
   - **Menu** Select **Tools>Options**.

   The Options dialog opens.
2. Select the **Build** tab.
3. Select **View output after successful build**.



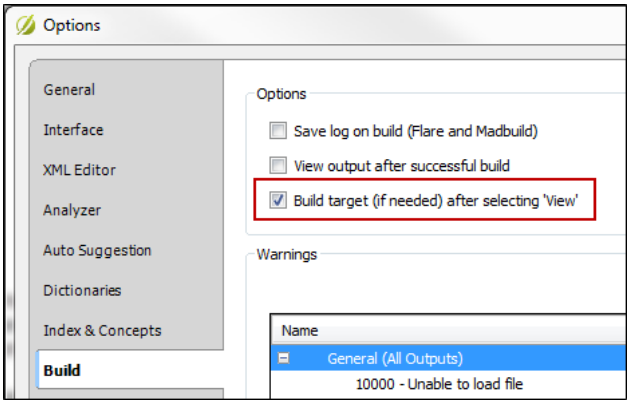
4. Click **OK**.

After you generate a target, the output will automatically open, without asking you first.

HOW TO BUILD OUTPUTS AUTOMATICALLY IF NEEDED WHEN SELECTING VIEW

1. Do one of the following, depending on the part of the user interface you are using:
   - **Ribbon** Select **File>Options**.
   - **Menu** Select **Tools>Options**.

   The Options dialog opens.
2. Select the **Build** tab.
3. Select **Build target (if needed) after selecting 'View'**.



4. Click **OK**.

APPENDIX A

PDF Guides

The following PDF guides are available for download from the online Help.

Accessibility Guide
Analyzer Guide
Autonumbers Guide
Condition Tags Guide
Context-sensitive Help Guide
DotNet Help Guide
Eclipse Help Guide
Getting Started Guide
Global Project Linking Guide
HTML Help Guide
HTML5 Guide
Images Guide
Importing Guide
Index Guide
Key Features Guide
Language Support Guide
Movies Guide
Navigation Links Guide
Print-based Output Guide
Project Creation Guide
Pulse Guide
QR Codes Guide
Reports Guide
Reviews & Contributions Guide
Search Guide
SharePoint Guide
Shortcuts Guide
Skins Guide

Snippets Guide

Source Control Guide

Styles Guide

Tables Guide

Tables of Contents Guide

Targets Guide

Templates Guide

Topics Guide

Touring the Workspace Guide

Transition From FrameMaker Guide

Variables Guide

WebHelp Outputs Guide

What's New Guide

APPENDIX B

Output Type Comparison Tables

Following are tables that show the various output types available, with the distinguishing features of each.

✓ Supported

✓ Supported with limitations

✗ Not supported

Note: If an item does not have a check mark, that does not necessarily mean the feature is unavailable in the output. Rather, it means that the feature cannot be added in Flare. For example, search cannot be added in Flare for EPUB output, but some ebook readers may have search built in.

General

Following are general features supported in each output type:

	ONLINE								PRINT					OTHER
	DotNet Help	Eclipse Help	Microsoft HTML Help	HTML	WebHelp	WebHelp Mobile	WebHelp Plus	AIR	FM	PDF	WORD	XHTML	XPS	EPUB
.NET Integration	✓	✗	✗	✗	✗	✗	✗	✗	✗	✗	✗	✗	✗	✗
Main Entry File Extension	.mchelp	.xml or .jar	.chm	.htm	.htm	.htm	.htm	.air	.fm or .book	.pdf	.xml (default)	.htm	.xps	.epub or .mobi
Output Window or Application	MadCap Help Viewer	Eclipse Help Viewer	Microsoft HTML Help Viewer	Browser Window	Browser Window	Mobile Web Browser	Browser Window	Application Window	Adobe FrameMaker	Browser Window or Adobe PDF Reader	Microsoft Word	Browser Window	Browser Window	Ebook Reader or Web
Responsive Output	✗	✗	✗	✓	✗	✗	✗	✗	✗	✗	✗	✗	✗	✗
Single Output File Possible	✗	✓	✓	✗	✗	✗	✗	✓	✗	✓	✓	✗	✓	✓

	ONLINE								PRINT					OTHER
	DotNet Help	Eclipse Help	Microsoft HTML Help	HTML	WebHelp	WebHelp Mobile	WebHelp Plus	AIR	FM	PDF	WORD	XHTML	XPS	EPUB
Special Requirements	MadCap DotNet Help Viewer	Eclipse Standard 4.3.1 (Kepler) or higher Java (V7 Update 45) or later	Internet Explorer	✗	✗	✗	IIS ASP.NET MS Indexing Service	Java Runtime Environment (JRE) Adobe AIR	✗	✗	✗	✗	✗	Reader

Accessibility

Following are accessibility features (e.g., Section 508, WCAG) supported in each output type:

	ONLINE								PRINT					OTHER
	DotNet Help	Eclipse Help	Microsoft HTML Help	HTML	WebHelp	WebHelp Mobile	WebHelp Plus	AIR	FM	PDF	WORD	XHTML	XPS	EPUB
Accessibility Supported	✓	✓	✓	✓	✓	✓	✓	✓	✗	✓	✗	✗	✗	✓
Compiler Warnings	✗	✓	✓	✓	✓	✓	✓	✓	✗	✓	✗	✗	✗	✗
Empty Alt Text for Images	✓	✓	✓	✓	✓	✓	✓	✓	✗	✓	✗	✗	✗	✗
Scrolling for Toolbars and Navigation Frames—Enable	✗	✗	✗	✗	✓	✗	✓	✗	✗	✗	✗	✗	✗	✗

Context-Sensitive Help

Following are context-sensitive Help (CSH) features supported in each output type:

	ONLINE								PRINT					OTHER
	DotNet Help	Eclipse Help	Microsoft HTML Help	HTML	WebHelp	WebHelp Mobile	WebHelp Plus	AIR	FM	PDF	WORD	XHTML	XPS	EPUB
CSH Supported	✓	✓	✓	✓	✓	✓	✓	✗	✗	✗	✗	✗	✗	✗
Embedded CSH Supported	✓	✗	✗	✗	✗	✗	✗	✗	✗	✗	✗	✗	✗	✗

Feedback Statistics and Reporting Features

Following are Feedback statistics and reporting features supported in each output type:

	ONLINE								PRINT					OTHER
	DotNet Help	Eclipse Help	Microsoft HTML Help	HTML 5	WebHelp	WebHelp Mobile	WebHelp Plus	AIR	FM	PDF	WORD	XHTML	XPS	EPUB
Feedback Supported	✓	✗	✓	✓	✓	✗	✓	✗	✗	✗	✗	✗	✗	✗
Search Results	✓	✗	✗	✓	✓	✗	✓	✗	✗	✗	✗	✗	✗	✗

Generated Content

Following are generated content features supported in each output type:

	ONLINE								PRINT					OTHER
	DotNet Help	Eclipse Help	Microsoft HTML Help	HTML	WebHelp	WebHelp Mobile	WebHelp Plus	AIR	FM	PDF	WORD	XHTML	XPS	EPUB
Autonumbers	✗	✗	✗	✗	✗	✗	✗	✗	✓	✓	✓	✓	✓	✓
Breadcrumbs	✓	✗	✓	✓	✓	✓	✓	✓	✗	✗	✗	✗	✗	✗
Browse Sequences	✓	✗	✓	✓	✓	✓	✓	✓	✗	✗	✗	✗	✗	✗
Concept Links	✓	✓	✓	✓	✓	✓	✓	✓	✗	✗	✗	✗	✗	✗
Keyword Links	✓	✓	✓	✓	✓	✓	✓	✓	✗	✗	✗	✗	✗	✗
List of Concepts	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
List of Elements	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓

	ONLINE								PRINT					OTHER
	DotNet Help	Eclipse Help	Microsoft HTML Help	HTML	WebHelp	WebHelp Mobile	WebHelp Plus	AIR	FM	PDF	WORD	XHTML	XPS	EPUB
List of Endnotes	✗	✗	✗	✗	✗	✗	✗	✗	✓	✓	✓	✓	✓	✓
Lists (Numbered and Bulleted)	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
Page Numbers	✗	✗	✗	✗	✗	✗	✗	✗	✓	✓	✓	✓	✓	✗
Related Topics Links	✓	✓	✓	✓	✓	✓	✓	✓	✗	✗	✗	✗	✗	✗
Relationship Links	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
Scripts	✓	✓	✓	✓	✓	✓	✓	✓	✗	✗	✗	✗	✗	✗
Shortcut Controls	✗	✗	✓	✗	✗	✗	✗	✗	✗	✗	✗	✗	✗	✗
Topic Toolbars	✓	✗	✓	✓	✓	✓	✓	✓	✗	✗	✗	✗	✗	✗

Glossaries

Following are glossary features supported in each output type:

	ONLINE								PRINT					OTHER
	DotNet Help	Eclipse Help	Microsoft HTML Help	HTML	WebHelp	WebHelp Mobile	WebHelp Plus	AIR	FM	PDF	WORD	XHTML	XPS	EPUB
Glossaries—Created via Glossary File and Proxy	✗	✓	✗	✗	✗	✗	✗	✗	✓	✓	✓	✓	✓	✓
Glossaries—Created via Glossary File and Skin Setting	✓	✗	✓	✓	✓	✓	✓	✓	✗	✗	✗	✗	✗	✗
Auto-generate	✗	✗	✗	✗	✗	✗	✗	✗	✓	✓	✓	✓	✓	✓

Horizontal Rules

Following are horizontal rule features supported in each output type:

	ONLINE								PRINT					OTHER
	DotNet Help	Eclipse Help	Microsoft HTML Help	HTML	WebHelp	WebHelp Mobile	WebHelp Plus	AIR	FM	PDF	WORD	XHTML	XPS	EPUB
Horizontal Rules Supported	✓	✓	✓	✓	✓	✓	✓	✓	✗	✓	✗	✓	✓	✓

Images

Following are image features supported in each output type:

	ONLINE								PRINT					OTHER
	DotNet Help	Eclipse Help	Microsoft HTML Help	HTML	WebHelp	WebHelp Mobile	WebHelp Plus	AIR	FM	PDF	WORD	XHTML	XPS	EPUB
Images Supported	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
Image Hyperlinks	✓	✓	✓	✓	✓	✓	✓	✓	✗	✗	✗	✗	✗	✓
Image Maps	✓	✓	✓	✓	✓	✓	✓	✓	✗	✗	✗	✗	✗	✗
Pre-Compile Resized	✓	✓	✓	✓	✓	✓	✓	✓	✓	✗	✓	✗	✗	✓
Thumbnail Images	✓	✓	✓	✓	✓	✓	✓	✓	✗	✗	✗	✗	✗	✗
Web-Safe—Convert from Non—Web-safe	✓	✓	✓	✓	✓	✓	✓	✓	✗	✗	✗	✗	✗	✗

Indexes

Following are index features supported in each output type:

	ONLINE								PRINT					OTHER
	DotNet Help	Eclipse Help	Microsoft HTML Help	HTML	WebHelp	WebHelp Mobile	WebHelp Plus	AIR	FM	PDF	WORD	XHTML	XPS	EPUB
Indexes—Created Using Keywords and Proxy	✗	✓	✗	✗	✗	✗	✗	✗	✓	✓	✓	✓	✓	✓
Indexes—Created Using Keywords and Skin Setting	✓	✗	✓	✓	✓	✓	✓	✓	✗	✗	✗	✗	✗	✗
Auto-generate	✗	✗	✗	✗	✗	✗	✗	✗	✓	✓	✓	✓	✓	✓
Binary Indexes	✗	✗	✓	✗	✗	✗	✗	✗	✗	✗	✗	✗	✗	✗
Bookmarks in Index Entries	✗	✗	✓	✗	✗	✗	✗	✗	✗	✗	✗	✗	✗	✗
Index Links	✓	✗	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
Search—Can Exclude Index Entries from Search	✓	✗	✗	✓	✓	✓	✓	✓	✗	✗	✗	✗	✗	✗

Language Support

Following are language support features supported in each output type:

	ONLINE								PRINT					OTHER
	DotNet Help	Eclipse Help	Microsoft HTML Help	HTML	WebHelp	WebHelp Mobile	WebHelp Plus	AIR	FM	PDF	WORD	XHTML	XPS	EPUB
Author and Generate Output in Left-to-Right (LTR) and Right-to-Left (RTL) Languages	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
Invert Image Callouts to RTL	✗	✓	✓	✓	✓	✓	✓	✓	✗	✓	✓	✓	✓	✓
Invert Page Layouts to RTL	✗	✗	✗	✗	✗	✗	✗	✗	✗	✓	✓	✓	✓	✗
Invert Styles to RTL	✗	✓	✓	✓	✓	✓	✓	✓	✗	✓	✓	✓	✓	✓
Output Interface—Display in LTR and RTL	✗	✓	✗	✓ Regular skin (UI Text tab)	✓ Language skin	✓ Language skin	✓ Language skin	✓ Language skin	✗	✗	✗	✗	✗	✗

	ONLINE								PRINT					OTHER
	DotNet Help	Eclipse Help	Microsoft HTML Help	HTML	WebHelp	WebHelp Mobile	WebHelp Plus	AIR	FM	PDF	WORD	XHTML	XPS	EPUB
Output Interface—Display English, French, Japanese, or German	✓ Possible with Help Viewer (end user selects language)	✗	✗	✗	✗	✗	✗	✗	✗	✗	✗	✗	✗	✗
Language Skins	✓	✗	✓	✗	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
Separate UI Text tab in Skin Editor for Localization	✗	✗	✗	✓	✗	✗	✗	✗	✗	✗	✗	✗	✗	✗

Master Pages and Page Layouts

Following are master page and page layout features supported in each output type:

	ONLINE								PRINT					OTHER
	DotNet Help	Eclipse Help	Microsoft HTML Help	HTML 5	WebHelp	WebHelp Mobile	WebHelp Plus	AIR	FM	PDF	WORD	XHTML	XPS	EPUB
Master Pages Supported	✓	✓	✓	✓	✓	✓	✓	✓	✓	✗	✓	✗	✗	✗
Page Layouts Supported	✗	✓	✗	✗	✗	✗	✗	✗	✓	✓	✓	✓	✓	✗

Merging Output

Following are merging output features supported in each output type:

	ONLINE								PRINT					OTHER
	DotNet Help	Eclipse Help	Microsoft HTML Help	HTML	WebHelp	WebHelp Mobile	WebHelp Plus	AIR	FM	PDF	WORD	XHTML	XPS	EPUB
Merge Output Supported	✓	✓	✓	✓	✓	✓	✓	✓	✗	✗	✗	✗	✗	✗
Merge Output at Runtime	✗	✗	✗	✓	✗	✗	✓	✗	✗	✗	✗	✗	✗	✗

Miscellaneous Features

Following are miscellaneous features supported in each output type:

	ONLINE								PRINT					OTHER
	DotNet Help	Eclipse Help	Microsoft HTML Help	HTML	WebHelp	WebHelp Mobile	WebHelp Plus	AIR	FM	PDF	WORD	XHTML	XPS	EPUB
Equations	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
Footnotes	✓ Converted to popup	✓	✓ Converted to popup	✓ Converted to popup	✓ Converted to popup	✓ Converted to link	✓ Converted to popup	✓ Converted to popup	✓	✓	✓	✓	✓	✓
Object Positioning	✓	✓	✓	✓	✓	✓	✓	✓	✗	✓	✗	✓	✓	✓
QR Codes	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
Preview Topics in Output Type	✓	✓	✓	✓	✓	✓	✓	✓	✗	✓	✗	✓	✓	✓
Redacted Text	✗	✗	✗	✗	✗	✗	✗	✗	✗	✓	✗	✓	✓	✗

	ONLINE								PRINT					OTHER
	DotNet Help	Eclipse Help	Microsoft HTML Help	HTML	WebHelp	WebHelp Mobile	WebHelp Plus	AIR	FM	PDF	WORD	XHTML	XPS	EPUB
Slideshows	✓	✗	✓	✓	✓	✓	✓	✓	✗	✗	✗	✗	✗	✗
Snippets	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
Tables	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
Text Boxes	✓	✓	✓	✓	✓	✓	✓	✓	✗	✓	✗	✓	✓	✗

Multimedia

Following are multimedia features supported in each output type:

	ONLINE								PRINT					OTHER
	DotNet Help	Eclipse Help	Microsoft HTML Help	HTML	WebHelp	WebHelp Mobile	WebHelp Plus	AIR	FM	PDF	WORD	XHTML	XPS	EPUB
Audio Supported	✓	✓	✓	✓	✓	✓ If mobile browser supports it	✓	✓	✗	✗	✗	✗	✗	✓ Depends on reader
Movies Supported	✓	✓ Does not support MP4 or MPG	✓	✓	✓	✓ If mobile browser supports it	✓	✓	✗	✗	✗	✗	✗	✓ Depends on reader
Movies—HTML5 Movie Format Supported (Depending on Browser)	✗	✓	✗	✓	✓	✓	✓	✓	✗	✗	✗	✗	✗	✗

Navigation Links

Following are navigation link features supported in each output type:

	ONLINE								PRINT					OTHER
	DotNet Help	Eclipse Help	Microsoft HTML Help	HTML	WebHelp	WebHelp Mobile	WebHelp Plus	AIR	FM	PDF	WORD	XHTML	XPS	EPUB
Cross-References Supported	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
Cross-References—Context-Sensitive	✗	✗	✗	✗	✗	✗	✗	✗	✓	✓	✓	✓	✓	✓
Drop-down Text	✓	✓	✓	✓	✓	✓	✓	✓	✗	✗	✗	✗	✗	✗
Expanding Text	✓	✓	✓	✓	✓	✓	✓	✓	✗	✗	✗	✗	✗	✗
Text Hyperlinks	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
Text Popups	✓	✓	✓	✓	✓	✓	✓	✓	✗	✗	✗	✗	✗	✗
Togglers	✓	✓	✓	✓	✓	✓	✓	✓	✗	✗	✗	✗	✗	✗
Topic Popups	✓	✓	✓	✓	✓	✓	✓	✓	✗	✗	✗	✗	✗	✗

Pulse (Socially Enabled Output)

Following are Pulse features supported in each output type:

	ONLINE								PRINT					OTHER
	DotNet Help	Eclipse Help	Microsoft HTML Help	HTML	WebHelp	WebHelp Mobile	WebHelp Plus	AIR	FM	PDF	WORD	XHTML	XPS	EPUB
Pulse Integration	✗	✗	✗	✓	✓	✗	✓	✗	✗	✗	✗	✗	✗	✗

Search

Following are search features supported in each online output type.

General Support

The table below summarizes search support in each output type:

	ONLINE								PRINT					OTHER
	DotNet Help	Eclipse Help	Microsoft HTML Help	HTML	WebHelp	WebHelp Mobile	WebHelp Plus	AIR	FM	PDF	WORD	XHTML	XPS	EPUB
Search Supported Via Flare Integration	✓	✗	✓	✓	✓	✓	✓	✓	✗	✗	✗	✗	✗	✗

End User Search Features

The table below summarizes the key search features and capabilities for each output type. With HTML5 and WebHelp outputs, you can distribute either client- or server-based outputs. This is why the table below lists them twice—under client- and server-based output.

End user search features are supported by the Eclipse Help Viewer and search operations are provided by a plug-in that you develop using the Eclipse platform (see **org.eclipse.help.ui.searchEngine**). The Flare search engine does not handle Eclipse Help search operations.

	CLIENT-BASED OUTPUT						SERVER-BASED OUTPUT	
	DotNet Help	Microsoft HTML Help	WebHelp	HTML 5	AIR	WebHelp Mobile	WebHelp Plus	HTML 5
Additional Requirements	Help Viewer SQL Server Compact (for asterisk (*) wildcard search and boolean operators)	Internet Explorer			Java Runtime Environment (JRE) Adobe AIR		IIS ASP.NET Microsoft Indexing Service or Windows Search	IIS ASP.NET Microsoft Indexing Service or Windows Search
Glossary Search—Find Matching Terms	✓	✓	✗	✓	✗	✗	✗	✓
Index Search—Find Matching Terms	✓	✓	✓	✓	✓	✗	✓	✓

	CLIENT-BASED OUTPUT						SERVER-BASED OUTPUT	
	DotNet Help	Microsoft HTML Help	WebHelp	HTML 5	AIR	WebHelp Mobile	WebHelp Plus	HTML 5
Search Favorites—Save Queries to Favorites List	✓	✗	✓	✗	✓	✗	✓	✗
Search Query—Asterisk (*) Wildcard Character	✓ Requires SQL Server Compact.	✓	✓	✓	✗	✗	✗	✗
Search Query—Boolean Operators	✓ Requires SQL Server Compact.	✓ Select operator from built-in list.	✓	✓	✓	✓	✗	✗
Search Query—Full-text Search (not case sensitive)	✓	✓	✓	✓	✓	✓	✓	✓
Search Query—Enclose Terms in Quotes (" ")	✓	✓	✓	✓	✓	✓	✓	✓
Search Results—Narrow Search Scope Using Search Filter Sets	✓	✗	✓	✓	✓	✗	✓	✓

	CLIENT-BASED OUTPUT						SERVER-BASED OUTPUT	
	DotNet Help	Microsoft HTML Help	WebHelp	HTML 5	AIR	WebHelp Mobile	WebHelp Plus	HTML 5
Search Results—Results Listed in Ranked Order and Show Rank Number	✓	✓ Results are listed in order. Rank number is not visible.	✓	✓ Results are listed in order. Rank number is not visible.	✓	✓ Results are listed in order. Rank number is not visible.	✓	✓ Results are listed in order. Rank number is not visible.
Search Results—Search Hits Highlighted in Topics	✓	✓	✓	✓	✓	✓	✓	✓

Content Developer Search Features

This table summarizes the key search features and capabilities that are available to you, the content developer, for each output type. With HTML5 and WebHelp outputs, you can distribute either client- or server-based outputs. This is why the table below lists them twice—under client- and server-based output.

	CLIENT-BASED OUTPUT						SERVER-BASED OUTPUT	
	DotNet Help	Microsoft HtML Help	WebHelp	HTML5	AIR	WebHelp Mobile	WebHelp Plus	HTML5
Enable Search Feature	✓ Search cannot be disabled.	✓ Standard skin	✓ Standard skin	✓ HTML5 skin	✓ Standard skin	✓ Mobile skin	✓ Standard skin	✓ HTML5 skin
Search Highlighting—Set Highlight Color	✓	✓	✓	✓	✓	✓	✓	✓
Search Performance—Chunk Large Search Database Files	✓	✗	✓	✓	✓	✓	✓	✓
Search Performance—Enable Partial-word Searching and Set Minimum Word Size	✓	✗	✓	✓	✓	✓	✓	✓

	CLIENT-BASED OUTPUT						SERVER-BASED OUTPUT	
	DotNet Help	Microsoft HTML Help	WebHelp	HTML 5	AIR	WebHelp Mobile	WebHelp Plus	HTML 5
Search Performance—Enable Stop Words in Search	✓	✓ Stop words always enabled.	✓	✓	✓	✓	✓	✓
Search Performance—Exclude Index Entries from Search	✓	✓ Index entries always excluded.	✓	✓	✓	✓	✓	✗
Search Performance—Exclude Non-words from Search	✓	✓ Non-words always excluded.	✓	✓	✓	✓	✓	✓
Search Performance—Include SQL Search Database	✓ Requires SQL Server Compact.	✗	✗	✗	✗	✗	✗	✗

	CLIENT-BASED OUTPUT						SERVER-BASED OUTPUT	
	DotNet Help	Microsoft HTML Help	WebHelp	HTML 5	AIR	WebHelp Mobile	WebHelp Plus	HTML 5
Search Performance—Pre-merge Search Database	✓	✗	✓	✓	✓	✓	✓	✓
Search Performance—Set N-Gram Size for Chinese, Japanese, and Korean Projects	✓	✗	✓	✓	✓	✓	✓	✓
Search Toolbar—Include and Customize a Quick Search Field and/or Select Search Button	✓	✗	✓	✗	✓	✗	✓	✗

Skins

Following are skin features supported in each output type:

	ONLINE								PRINT					OTHER
	DotNet Help	Eclipse Help	Microsoft HTML Help	HTML	WebHelp	WebHelp Mobile	WebHelp Plus	AIR	FM	PDF	WORD	XHTML	XPS	EPUB
Skin Type	Standard	✗	Standard	HTML5	Standard	Mobile	Standard	Standard	✗	✗	✗	✗	✗	✗
About Box	✗	✗	✓	✗	✓	✓	✓	✓	✗	✗	✗	✗	✗	✗
Accordion Titles—Exclude	✗	✗	✗	✗	✓	✗	✓	✓	✗	✗	✗	✗	✗	✗
Browser Settings	✗	✗	✗	✓	✓	✗	✓	✗	✗	✗	✗	✗	✗	✗
Caption for Output Window	✗	✗ org.eclipse.help.base plug-in	✓	✓	✓	✓	✓	✓	✗	✗	✗	✗	✗	✗

	ONLINE								PRINT					OTHER
	DotNet Help	Eclipse Help	Microsoft HTML Help	HTML	WebHelp	WebHelp Mobile	WebHelp Plus	AIR	FM	PDF	WORD	XHTML	XPS	EPUB
Elements (e.g., tabs, accordions)—Specify Default Element	✗	✗	✓	✓	✓	✗	✓	✓	✗	✗	✗	✗	✗	✗
Elements (e.g., tabs, accordions)—Specify Which to Include	✓	✗	✓	✓	✓	✓	✓	✓	✗	✗	✗	✗	✗	✗
Feedback Comments, Email Notifications, User Profile	✓	✗	✓	✓	✓	✗	✓	✗	✗	✗	✗	✗	✗	✗
Generate All	✓	✗	✓	✗	✓	✗	✓	✓	✗	✗	✗	✗	✗	✗
Language Skins	✓	✗	✓	✗	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓

	ONLINE								PRINT					OTHER
	DotNet Help	Eclipse Help	Microsoft HTML Help	HTML	WebHelp	WebHelp Mobile	WebHelp Plus	AIR	FM	PDF	WORD	XHTML	XPS	EPUB
Navigation Links in Stand-alone Topics	✗	✗	✗	✓	✓	✗	✓	✓	✗	✗	✗	✗	✗	✗
Navigation Pane Settings	✗	✗	✓	✓	✓	✗	✓	✓	✗	✗	✗	✗	✗	✗
Preview Skin for Output Type	✓	✗	✓	✓	✓	✓	✓	✓	✗	✗	✗	✗	✗	✗
Separate UI Text Tab in Skin Editor for Localization	✗	✗	✗	✓	✗	✗	✗	✗	✗	✗	✗	✗	✗	✗

	ONLINE								PRINT					OTHER
	DotNet Help	Eclipse Help	Microsoft HTML Help	HTML	WebHelp	WebHelp Mobile	WebHelp Plus	AIR	FM	PDF	WORD	XHTML	XPS	EPUB
Styles	✓ Only some styles (e.g., Feedback, toolbar) are supported.	✗	✓ Only some styles (e.g., Feedback, toolbar) are supported.	✓	✓	✓	✓	✓	✗	✗	✗	✗	✗	✗
TOC Entry—Select Skin For	✗	✗	✓	✗	✗	✗	✗	✗	✗	✗	✗	✗	✗	✗
Topic Tool-bar—Custom Settings	✓	✗	✓	✓ Topic tool-bar settings controlled in proxy or stylesheet, not in skin	✓	✓ Limited settings	✓	✓	✗	✗	✗	✗	✗	✗

	ONLINE								PRINT					OTHER
	DotNet Help	Eclipse Help	Microsoft HTML Help	HTML	WebHelp	WebHelp Mobile	WebHelp Plus	AIR	FM	PDF	WORD	XHTML	XPS	EPUB
Web Toolbar—Add via Skin	✗	✗	✓	✗	✗	✗	✗	✗	✗	✗	✗	✗	✗	✗
Web Toolbar—Custom Set-tings	✗	✗	✓	✓	✓	✗	✓	✓	✗	✗	✗	✗	✗	✗
Window—Size/Position	✓	✗	✓	✓	✓	✗	✓	✓	✗	✗	✗	✗	✗	✗
Window—User-Defined Size/Position	✗	✗	✓	✗	✗	✗	✗	✗	✗	✗	✗	✗	✗	✗

Tables of Contents and Mini-TOCs

Following are table of contents (TOC) and mini-TOC features supported in each output type:

	ONLINE								PRINT					OTHER
	DotNet Help	Eclipse Help	Microsoft HTML Help	HTML	WebHelp	WebHelp Mobile	WebHelp Plus	AIR	FM	PDF	WORD	XHTML	XPS	EPUB
TOCs—Created via Headings and Proxy	✗	✓	✗	✗	✗	✗	✗	✗	✓	✓	✓	✓	✓	✓ Some ebook readers have built-in TOC
TOCs—Created via TOC File and Skin Setting	✓	✗	✓	✓	✓	✓	✓	✓	✗	✗	✗	✗	✗	✗
Auto-generate	✗	✗	✗	✗	✗	✗	✗	✗	✓	✓	✓	✓	✓	✓
Binary TOCs	✗	✗	✓	✗	✗	✗	✗	✗	✗	✗	✗	✗	✗	✗
Mark as New	✗	✓	✓	✓	✓	✓	✓	✓	✗	✗	✗	✓	✗	✗
Mini-TOCs	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓

	ONLINE								PRINT					OTHER
	DotNet Help	Eclipse Help	Microsoft HTML Help	HTML	WebHelp	WebHelp Mobile	WebHelp Plus	AIR	FM	PDF	WORD	XHTML	XPS	EPUB
Skin—Select for TOC Entry	✗	✗	✓	✗	✗	✗	✗	✗	✗	✗	✗	✗	✗	✗
Synchronize With Topics	✓	✗	✓	✓	✓	✗	✓	✓	✗	✗	✗	✗	✗	✗

Target Settings

Following are target features supported in each output type:

	ONLINE								PRINT					OTHER
	DotNet Help	Eclipse Help	Microsoft HTML Help	HTML	WebHelp	WebHelp Mobile	WebHelp Plus	AIR	FM	PDF	WORD	XHTML	XPS	EPUB
Auto-generate Glossary, Index, TOC	✗	✗	✗	✗	✗	✗	✗	✗	✓	✓	✓	✓	✓	✓
Characters and Spaces—Replace With Underscores	✓	✓	✓	✓	✓	✓	✓	✓	✗	✗	✗	✗	✗	✗
CMYK	✗	✗	✗	✗	✗	✗	✗	✗	✗	✓	✗	✗	✗	✗
Content Folder—Omit from Output	✗	✓	✓	✓	✓	✗	✓	✓	✗	✗	✗	✗	✗	✗
Crop and Registration Marks—Include in Output	✗	✗	✗	✗	✗	✗	✗	✗	✗	✓	✗	✗	✗	✗

	ONLINE								PRINT					OTHER
	DotNet Help	Eclipse Help	Microsoft HTML Help	HTML	WebHelp	WebHelp Mobile	WebHelp Plus	AIR	FM	PDF	WORD	XHTML	XPS	EPUB
DOCTYPE Declaration	✓	✗	✓ Recommend disabling this option for non-English content	✗	✓	✓ Standard mobile DOCTYPE always used	✓	✓	✗	✗	✗	✗	✗	✗
Empty Pages—Inject Automatically	✗	✗	✗	✗	✗	✗	✗	✗	✗	✓	✗	✗	✓	✗
File Extensions—Custom	✓	✓	✗	✓	✓	✓	✓	✓	✗	✗	✗	✗	✗	✗
Mark of the Web	✗	✓	✗	✓	✓	✗	✓	✗	✗	✗	✗	✗	✗	✗
Meta Tags—Custom	✓	✓	✓	✓	✓	✓	✓	✓	✗	✗	✗	✗	✗	✗
Startup Topic	✓	✗	✓	✓	✓	✗	✓	✓	✗	✗	✗	✗	✗	✗

	ONLINE								PRINT					OTHER
	DotNet Help	Eclipse Help	Microsoft HTML Help	HTML	WebHelp	WebHelp Mobile	WebHelp Plus	AIR	FM	PDF	WORD	XHTML	XPS	EPUB
Stylesheet Medium	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
Warnings—Ignore	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓

Variables

Following are variable features supported in each output type:

	ONLINE								PRINT					OTHER
	DotNet Help	Eclipse Help	Microsoft HTML Help	HTML	WebHelp	WebHelp Mobile	WebHelp Plus	AIR	FM	PDF	WORD	XHTML	XPS	EPUB
Custom Variables	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
Heading Variables	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✗
Running HF Variables	✗	✗	✗	✗	✗	✗	✗	✗	✓	✗	✓	✗	✗	✗
System Variables	✓	✓	✓	✓	✓	✓	✓	✓	✗	✓	✓	✓	✓	✓