MadCap Software

Accessibility 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

CONTENTS

CHAPTER 1 About Accessibility 5
Accessibility Standards and Guidelines 6
CHAPTER 2 Alternate Text and Screen Tips 9
Alt Text and Title Attributes (Screen Tips) 10
Images—Apply Alt Text and Screen Tip to All References 12
Applying Empty Alt Text 13
CHAPTER 3 Accessibility and Flare Outputs 15
HTML5 and WebHelp Outputs 16
PDF Output 18
Eclipse Help Output 19
CHAPTER 4 Accessibility Warnings 21
Accessibility Warnings—PDF, Eclipse Help, HTML Help, and WebHelp Outputs 22
Web Accessibility Warnings—Eclipse Help, HTML Help, HTML5, and WebHelp Outputs 23
Taking Action on Warnings 24
Ignoring Warnings 25
APPENDIX PDF Guides 27

CHAPTER 1

About Accessibility

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

Accessibility has to do with making sure that your documentation can be accessed by people with disabilities (i.e., individuals who have visual, hearing, and mobility impairments). Documentation is considered accessible when it is easier for those with disabilities to use and easier to read using assistive technologies such as Microsoft Windows Narrator, Connect Outloud, and Job Access With Speech (JAWS).

Different output formats in Flare support different accessibility features.

For more information not covered in this guide, see the online Help.

Accessibility Standards and Guidelines

Following are the two primary web accessibility standards:

- **Section 508** In 1973 the United States government passed the Rehabilitation Act. Part of this Act is Section 508, which was amended in 1998 and requires Federal agencies to follow standards in developing, maintaining, acquiring, or using electronic and information technology to make systems accessible to people with disabilities.

  See **http://www.section508.gov/index.cfm?fuseAction=stdsdoc**.

- **Web Content Accessibility Guidelines (WCAG)** These guidelines are created by the Web Accessibility Initiative (WAI), as part of the World Wide Web Consortium (W3C), in an effort to make the web accessible to people with disabilities.

  See **http://www.w3.org/WAI/intro/wcag.php**.

Section 508 and WCAG, although similar, are not the same thing. In the United States, all Federal agencies by law must comply with Section 508. On the other hand, WCAG is a set of recommendations for making web content accessible.

Following are standards and guidelines outlined by Section 508 and WCAG. As you can see, there are some similarities between the lists.

Section 508	WCAG
A text equivalent for every non-text element shall be provided (e.g., via "alt", "longdesc," or in element content). Equivalent alternatives for any multimedia presentation shall be synchronized with the presentation. Web pages shall be designed so that all information conveyed with color is also available without color, for example from context or markup. Documents shall be organized so they are readable without requiring an associated stylesheet. Redundant text links shall be provided for each active region of a server-side image map. Client-side image maps shall be provided instead of server-side image maps except where the regions cannot be defined with an available geometric shape. Row and column headers shall be identified for data tables. Markup shall be used to associate data cells and header cells for data tables that have two or more logical levels of row or column headers. Frames shall be titled with text that facilitates frame identification and navigation. Pages shall be designed to avoid causing the screen to flicker with a frequency greater than 2 Hz and lower than 55 Hz. A text-only page, with equivalent information or functionality, shall be provided to make a web site comply with the provisions of this part, when compliance cannot be accomplished in any other way. The content of the text-only page shall be updated whenever the primary page changes. When pages utilize scripting languages to display content, or to create interface elements, the information provided by the script shall be identified with functional text that can be read by assistive technology. When a web page requires that an applet, plug-in or other application be present on the client system to interpret page content, the page must provide a link to a plug-in or applet that complies with §1194.21 (a) through (l).	**Text Alternatives** Provide text alternatives for any non-text content so that it can be changed into other forms people need, such as large print, braille, speech, symbols or simpler language. **Time-based Media** Provide alternatives for time-based media. **Adaptable** Create content that can be presented in different ways (for example simpler layout) without losing information or structure. **Distinguishable** Make it easier for users to see and hear content including separating foreground from background. **Keyboard Accessible** Make all functionality available from a keyboard. **Enough Time** Provide users enough time to read and use content. **Seizures** Do not design content in a way that is known to cause seizures. **Navigable** Provide ways to help users navigate, find content, and determine where they are. **Readable** Make text content readable and understandable. **Predictable** Make web pages appear and operate in predictable ways. **Input Assistance** Help users avoid and correct mistakes. **Compatible** Maximize compatibility with current and future user agents, including assistive technologies.

Section 508	WCAG
When electronic forms are designed to be completed on-line, the form shall allow people using assistive technology to access the information, field elements, and functionality required for completion and submission of the form, including all directions and cues. A method shall be provided that permits users to skip repetitive navigation links. When a timed response is required, the user shall be alerted and given sufficient time to indicate more time is required.	

CHAPTER 2

Alternate Text and Screen Tips

You can make output accessible through the use of alternate text and screen tips.

This chapter discusses the following:

Alt Text and Title Attributes (Screen Tips) 10

Images—Apply Alt Text and Screen Tip to All References 12

Applying Empty Alt Text 13

Alt Text and Title Attributes (Screen Tips)

How do you make your output accessible? You can accomplish much of the work by making sure you have set alt and title attributes on your various elements—such as links, images, forms, and multimedia. These two attributes, although similar, are not identical.

An alt attribute is intended to provide *alternative* information for elements that cannot be seen (e.g., visually impaired individuals cannot see images in your output). The alt attribute text should briefly describe what the element is about.

On the other hand a title attribute is created when you add screen tip (or tool tip) text for an element. The title attribute text should provide *additional* information about the element.

Following are the different types of elements for which you can set alt and/or title attributes:

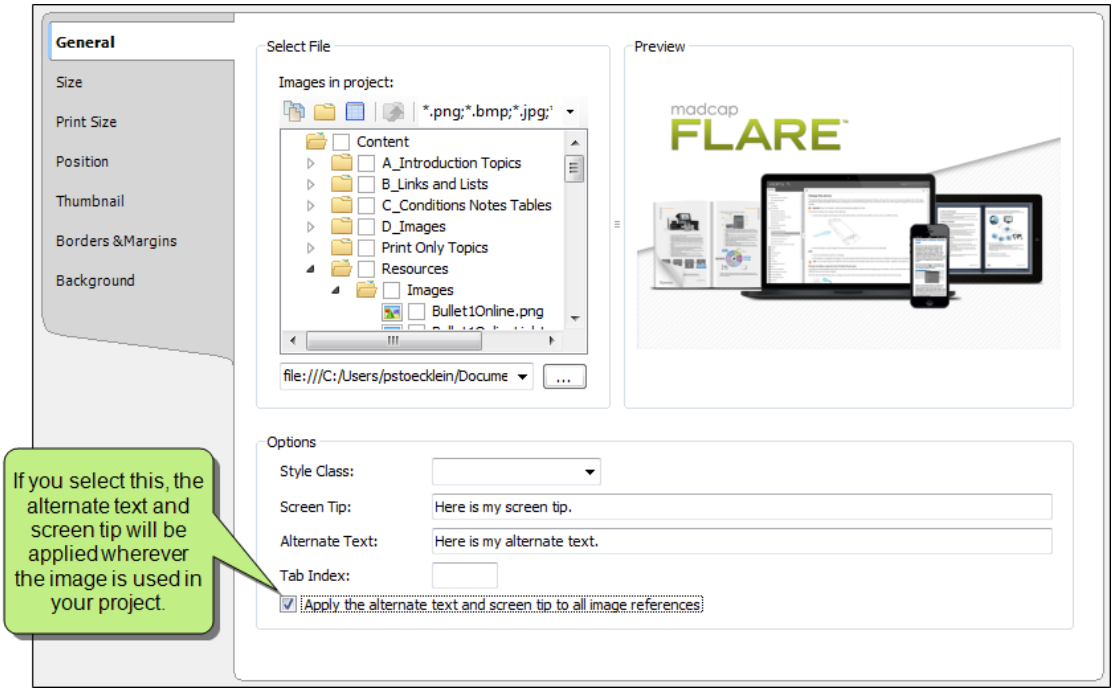
Element	Alt Attribute (Alternate Text)	Title Attribute (Screen Tip)
Audio (Links and Embedded)	✗	✓
Concept Links	✓	✗
Controls (e.g., Buttons, Labels, Input Boxes)	✓	✓
Cross-References*	✓	✓
Drop-Down Text	✓	✗
Equations	✓	✓
Expanding Text	✓	✗
Glossary Term Links	✓	✗
Image Hyperlinks	✓	✓
Image Maps	✓	✓

Element	Alt Attribute (Alternate Text)	Title Attribute (Screen Tip)
Images	✓	✓
Keyword Links	✓	✗
Movies (ILinks and Embedded)	✗	✓
QR Codes	✓	✓
Related Topics Links	✓	✗
Table of Contents (TOC) Icons	✓	✗
Text Hyperlinks*	✓	✓
Togglers	✓	✗
Toolbar Items	✓	✓
Topic Popups	✗	✓

* When you add a screen tip a <title> tag is created in the markup. If you insert cross-references or text hyperlinks, an <alt> (alternate text) tag is also added automatically with the same text.

Images—Apply Alt Text and Screen Tip to All References

When you insert or edit an image, you will see a check box that allows you to apply the same alternate text and screen tip for all instances of the image throughout your project.



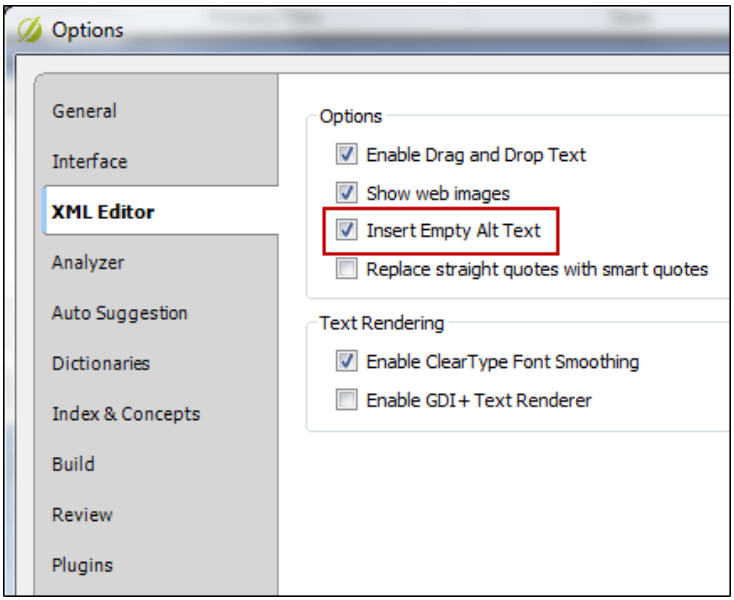
Applying Empty Alt Text

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**.

CHAPTER 3

Accessibility and Flare Outputs

You can work with accessibility in different ways with Flare's various output types.

This chapter discusses the following:

HTML5 and WebHelp Outputs 16
PDF Output 18
Eclipse Help Output 19

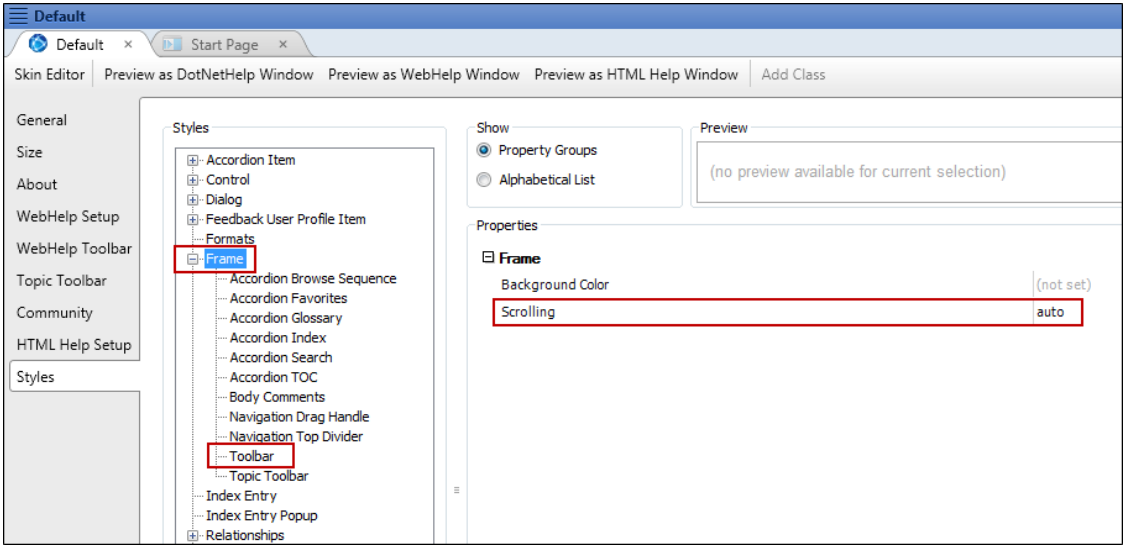
HTML5 and WebHelp Outputs

If you are generating any of the WebHelp output formats (WebHelp, HTML5, WebHelp AIR, WebHelp Mobile, WebHelp Plus), the runtime and navigation are Section 508 compliant. This helps to ensure your output's adherence to item 1194.22 of the Section 508 specification.

SCROLLING FOR TOOLBAR AND NAVIGATION FRAMES

The Section 508 standard states that "Documents shall be organized so they are readable without requiring an associated stylesheet." This means that some users need to view output with styles disabled in the browser. However, if styles are disabled, the frame containing the toolbar in WebHelp output may not be fully visible, and no scroll bar is provided because it was explicitly set not to display. Therefore, in order to be in compliance with Section 508, you can use an option in a Standard skin in Flare to set scrolling values for the toolbar and navigation frames in WebHelp and WebHelp Plus output.

This option can be found in the Skin Editor for Standard skins. In the **Styles** tab, you can place the setting on the **Frame** node or on the **Toolbar** node under it.



There are two options available:

- **auto** Scroll bars appear if needed. Select this option to ensure the output is compliant with Section 508.
- **no** Scroll bars are never shown. This is the default option.

Note: This *only* takes effect when stylesheets are disabled in the web browser. Please refer to your browser for instructions on how to disable styles.

PDF Output

Improved PDF accessibility helps ensure that PDFs generated by Flare are as accessible as possible to users with disabilities.

When you generate a PDF, the following help make the output more accessible:

- **Images—Alternate Text** HTML "alt" (alternate text) attributes are converted to the PDF equivalent during compilation so that screen readers have a way to describe the image. It is up to you to ensure that alt attributes are present on all images and descriptive enough to be useful to your end users.
- **Language for Content** XML "lang" attributes are converted to the PDF equivalent during compilation. This applies to both the project-level language (i.e., the language of the entire output document) and to element-level languages (i.e., languages applied to specific XHTML elements). This is an automatic process that occurs behind the scenes without user input.
- **PDF Tagging** You can select an option on the PDF Options tab of the Target Editor 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.

  Note: If you enable this feature, the size of the PDF output may increase somewhat. Compilation time should not be significantly affected.

- **Tables—Header Cells** Tables in PDF output are given a structure that designates which cells are header cells and which are content so that screen readers can translate the table data in a useful way.

Eclipse Help Output

Improved accessibility helps ensure that an Eclipse Help plug-in generated by Flare is as accessible as possible to users with disabilities.

The Eclipse platform provides its own accessibility features for the Eclipse Help Viewer. To learn more about accessibility features, visit the Workbench User Guide at **http://help.eclipse.org/**.

CHAPTER 4

Accessibility Warnings

There is a Warnings tab in the Target Editor for all outputs. These tabs contain check boxes that let you choose whether to be informed when certain issues occur while building output. Some of these warnings are related to accessibility for the following output types:

- HTML5 and WebHelp outputs
- Adobe PDF
- Eclipse Help
- Microsoft HTML Help

In other words, the accessibility warnings let you know ways that your output may not be as accessible as it could be.

This chapter discusses the following:

Accessibility Warnings—PDF, Eclipse Help, HTML Help, and WebHelp Outputs 22

Web Accessibility Warnings—Eclipse Help, HTML Help, HTML5, and WebHelp Outputs 23

Taking Action on Warnings 24

Ignoring Warnings 25

Accessibility Warnings—PDF, Eclipse Help, HTML Help, and WebHelp Outputs

The following warnings are available for Adobe PDF, Eclipse Help, Microsoft HTML Help, HTML5, and all of the WebHelp outputs:

- **Table has no header elements**
- **Table element does not contain a caption or summary**
- **QR code element does not contain alt text**
- **Equation element does not contain alt text**
- **Image element does not contain alt text**

Web Accessibility Warnings—Eclipse Help, HTML Help, HTML5, and WebHelp Outputs

The following additional web accessibility warnings are available for Eclipse Help, Microsoft HTML Help, and all of the WebHelp outputs:

- **Frame element has no title or name**
- **Input element has no associated label element**
- **Image button element does not contain alt text**
- **Server-side image map entry found**
- **Area element does not contain alt text**

Taking Action on Warnings

If you receive one or more of these types of compiler warnings, what should you do? That depends on the type of warning you receive:

- **Image Elements** If you receive this type of warning, you can fix the issue by editing the image and entering a description in the **Alternate Text** field.
- **Equation Elements** If you receive this type of warning, you can fix the issue by editing the equation and entering a description in the **Alternate Text** field.
- **QR Code Elements** If you receive this type of warning, you can fix the issue by editing the QR code and entering a description in the **Alternate Text** field.
- **Tables** If you receive this type of warning, you can fix the issue by editing the table properties and entering caption text (above the table) and/or adding a header row.
- **Form Elements** If you receive this type of warning, you can fix the issue by opening the markup for the document and adding a label to the form. To edit the markup, you can use the Internal Text Editor or Notepad. For more information about working with form elements (and labels within them), see **w3.org**.
- **Frame Elements** If you receive this type of warning, you can fix the issue by opening the markup for the document and adding a title to the frame element. To edit the markup, you can use the Internal Text Editor or Notepad. For more information about working with frame elements (and titles/names within them), see **w3.org**.

Ignoring Warnings

By default, all warnings (both those for accessibility and others) are enabled. However, can disable any warnings you want, telling Flare to ignore those 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.

APPENDIX

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