



Advanced Techniques
for Creating
Accessible
Adobe PDF Files

©2001 Adobe Systems Incorporated. All rights reserved.

Advanced Techniques for Creating Accessible Adobe® PDF Files

This manual, as well as the software described in it, is furnished under license and may be used or copied only in accordance with the terms of such license. The content of this manual is furnished for informational use only, is subject to change without notice, and should not be construed as a commitment by Adobe Systems Incorporated. Adobe Systems Incorporated assumes no responsibility or liability for any errors or inaccuracies that may appear in this book.

Except as permitted by such license, no part of this publication may be reproduced, stored in a retrieval system, or transmitted, in any form or by any means, electronic, mechanical, recording, or otherwise, without the prior written permission of Adobe Systems Incorporated.

Please remember that existing artwork or images that you may want to include in your project may be protected under copyright law. The unauthorized incorporation of such material into your new work could be a violation of the rights of the copyright owner. Please be sure to obtain any permission required from the copyright owner.

Any references to company names in sample templates are for demonstration purposes only and are not intended to refer to any actual organization.

Adobe, the Adobe logo, Acrobat, Acrobat Reader, Acrobat Capture, and PageMaker, are either registered trademarks or trademarks of Adobe Systems Incorporated in the United States and/or other countries. Microsoft and Windows are either registered trademarks or trademarks of Microsoft Corporation in the United States and/or other countries. All other trademarks are the property of their respective owners.

Contains an implementation of the LZW algorithm licensed under U.S Patent 4,558,302.

Adobe Systems Incorporated, 345 Park Avenue, San Jose, California 95110, USA.

Notice to U.S. government end users. The software and documentation are “commercial items,” as that term is defined at 48 C.F.R. §2.101, consisting of “commercial computer software” and “commercial computer software documentation,” as such terms are used in 48 C.F.R. §12.212 or 48 C.F.R. §227.7202, as applicable. Consistent with 48 C.F.R. §12.212 or 48 C.F.R. §§227.7202-1 through 227.7202-4, as applicable, the commercial computer software and commercial computer software documentation are being licensed to U.S. government end users (A) only as commercial items and (B) with only those rights as are granted to all other end users pursuant to the terms and conditions set forth in the Adobe standard commercial agreement for this software. Unpublished rights reserved under the copyright laws of the United States. Adobe Systems Incorporated, 345 Park Avenue, San Jose, CA 95110-2704, USA. For U.S. Government End Users, Adobe agrees to comply with all applicable equal opportunity laws including, if appropriate, the provisions of Executive Order 11246, as amended, Section 402 of the Vietnam Era Veterans Readjustment Assistance Act of 1974 (38USC 4212), and Section 503 of the Rehabilitation Act of 1973, as amended, and the regulations at 41 CFR Parts 60-1 through 60-60, 60-250, and 60-741. The affirmative action clause and regulations contained in the preceding sentence shall be incorporated by reference.

Contents

Overview

About Adobe PDF	1
Authoring for Accessibility	2
About Adobe Acrobat Software	2
About this Guide	4

Tagged Adobe PDF, Read Order, and Reflow

Creating Tagged Adobe PDF	6
About Element Tags	7
About Artifacts	9
About Element Properties	10

Checking the Read Order of Documents

Viewing the Logical Structure Tree	13
Reflowing a document	15

Modifying the Structure of a Document

Rearranging elements in the structure tree	17
Fixing improperly marked artifacts	18
Tagging manually to fix problem areas	19
Adding and fixing links	21
Creating and tagging form fields	25
Changing structure element types	27
Working with figures and alternate text	27

Design Issues and Accessible Adobe PDF

Working with tables	29
Working with fonts	30
Working with complex vector graphics	31
Working with complex layouts	31
Working with layers	31

About Tag Adobe PDF Agent

Tag Adobe PDF Agent tips	33
--------------------------	----

Appendix: Adobe Standard Elements

About Adobe Standard Element Tag Types	35
--	----

Overview

Welcome to Adobe® Systems Incorporated's *Advanced Techniques for Creating Accessible Adobe PDF Files*, a step-by-step guide that covers more advanced techniques for optimizing Adobe Portable Document Format (PDF) files so that they can be made accessible to users with disabilities such as blindness or low vision.

This volume is the second in a series. The first volume is titled *How to Create Accessible Adobe PDF files*, and covers the basics of using Adobe Acrobat® 5.0 software and Adobe Acrobat Capture® 3.0 software to optimize your documents for distribution as Adobe PDF files that can be read easily with the aid of Windows®-based assistive technology such as screen readers. In this volume you will learn how to modify the structure and read order of a document using the Acrobat 5.0 Tags palette.

About Adobe PDF

Adobe Portable Document Format (PDF) is standard for electronic document distribution worldwide. Adobe PDF is a universal file format that preserves the exact look and feel of any source document, including all of the fonts, formatting, colors, and graphics, regardless of the application and platform used to create it. Adobe PDF files can be opened reliably across a broad range of hardware and software with the free Adobe Acrobat Reader® software.

Adobe PDF is a standard on the World Wide Web and is also used to distribute electronic documents over corporate networks, via e-mail, and on CD-ROM. Many corporations, educational institutions, and government agencies use Adobe PDF to distribute documents to the public or to groups within their organizations. Adobe PDF is also gaining tremendous popularity as the basis for new e-business and e-government processes. Its support for electronic forms, digital signatures, password security, and electronic mark-ups make it the ideal platform for converting yesterday's paper-based business processes to fully interactive digital processes.

Authoring for Accessibility

Whether you're publishing a document in HTML, Adobe PDF, or some other format, creating accessible documents requires much more than simply representing the original document accurately. Sighted people can look at a printed page and easily discern the difference between titles, subtitles, columns of text, headers, footers, and so on. Visual clues, such as location of the text on the page, bold text, and large font sizes help them determine the structure of a document so they can read and navigate it easily.

Unfortunately, assistive technologies such as screen readers can't depend on these visual clues. They must instead rely on the underlying computer-based information to provide that same structure. As a result, making documents accessible depends on two things:

- Authoring the original documents so that they contain not just content (such as the text in the document) but also information about the structure of the content (such as how the text flows within the page and from page to page).
- Using publishing tools such as Adobe Acrobat software and Adobe PDF technology that can retain and encode both the content and the structure so that it can be interpreted by assistive technology.

In order to do this:

- Authors need to be aware of the importance of writing with the intent of creating accessible documents, and how to accomplish that.
- Authoring and publishing tools can be enhanced to help authors create accessible documents.

These same requirements apply to any type of publishing environment, regardless of the file format or application. If you want to make it possible for people with screen readers to navigate documents correctly, the underlying structural information must be present.

About Adobe Acrobat Software

Adobe Acrobat 5.0 software provides a number of capabilities that improve access for disabled users. In particular, it:

- Creates tagged Adobe PDF files when converting Microsoft® Office 2000 files to Adobe PDF, making it easier for people who use screen reader software to navigate a document in the proper reading order.
- Converts new and existing PDF files to tagged Adobe PDF files with the optional Make Accessible plug-in.
- Includes tools to help troubleshoot and optimize Adobe PDF files for accessibility, including the accessibility checker, the Tags palette, and more.
- Supports the Microsoft Active Accessibility (MSAA) Application Programming Interface (API) for the Windows platform for integration with assistive technology products, including screen readers from vendors such as GW Micro (www.gwmicro.com) and Freedom Scientific (www.freedomscientific.com).
- Provides usability enhancements, including enhanced keyboard shortcuts, support for high-contrast viewing, and the ability to zoom in and reflow text on the screen.
- Improves onscreen viewing by providing the ability to zoom in on text, and the ability to reflow the text on the screen for users with impaired vision.
- Supports export of text from a tagged Adobe PDF file to Rich Text Format (RTF), or to XML, HTML, or TXT via a plug-in for Macintosh® and Windows. Resulting files can be repurposed with a variety of applications including word processors and non-MSAA-compliant screen readers.
- Applies optical character recognition (OCR) to scanned pages via the Paper Capture plug-in (for Acrobat 5.0 for Windows) or the Paper Capture Online service (for Macintosh and Windows). Resulting files contain computer-generated text, which is required for use with assistive technology.

Adobe Acrobat Capture 3.0, meanwhile, is a powerful, flexible tool that lets you convert paper-based documents into accessible Adobe PDF files. This software:

- Retains the exact look and feel of the original document while embedding a layer of searchable, accessible text using Optical Character Recognition (OCR) software.
- Lets you review and fix suspect text quickly and easily with the QuickFix tool.
- Eliminates random text by letting you pre-select areas of the page to treat as text or graphics using the Zone Tool.
- Adds hyperlinks and bookmarks automatically so users can easily navigate tables of contents, indexes, and more.

- Creates tagged Adobe PDF files when converting paper documents through use of the optional Tag Adobe PDF agent.

Important: *The capabilities described in this guide require the Make Accessible plug-in for Acrobat 5.0 and the Tag Adobe PDF agent for Acrobat Capture 3.0. See “Using the Make Accessible Plug-in” on page 20 and “Working with Scanned and Paper Documents” on page 13 of How to Create Accessible Adobe PDF Files for information on how to obtain these optional tools.*

About this Guide

In this guide, you'll find advanced procedures and techniques you need to optimize the structure of accessible Adobe PDF files. Of course, your specific workflow will depend on the type of document you are starting with. This guide will show you how to do the following:

- Check documents for accessibility.
- Navigate the logical structure tree of a document in the Tags palette in Acrobat 5.0.
- View and modify the structure of a tagged Adobe PDF document.
- Manually create structure elements and order them using the Tags palette.
- Create and order tags for special elements such as links and form fields.
- Work around problems that can arise using fonts, tables, and complex layouts.

This guide assumes you are familiar with the different applications described. For more detail on each application and feature, please refer to the user guide for that product.

Tagged Adobe PDF, Read Order, and Reflow

Adobe is committed to providing solutions that improve accessibility to information contained in Adobe PDF files by enhancing support for assistive technology such as screen readers through the MSAA API for the Windows platform.

Tagged Adobe PDF is a version of PDF that provides structure and order information to allow PDF documents to be read by screen readers.

The major requirements for accessible documents covered in this volume are:

Logical reading order For a screen reader to effectively read information on a page, clues must be provided to help it determine in what order the text on the page is meant to be read. This order can be ambiguous in the case of multi-column text, side bars, pulled quotes inserted in the middle of or between columns, tables, and the like. By tagging and identifying the various blocks of text and other page elements that make up the layout, tagged Adobe PDF provides a means of supplying these clues, and defining the intended read order of the page.

Alternate text descriptions for images Graphic representations such as illustrations, drawings, charts and graphs cannot be interpreted by a screen reader. By supplying description text for images, a screen reader can describe to a visually impaired user the information the image is meant to impart.

Document language Although current screen readers are designed to read documents based on one language dictionary, it is anticipated that future technologies will be able to access multiple language dictionaries. For this reason, the specification of the document language is considered a requirement for providing accessible documents online. Acrobat 5.0's Accessibility Checker will log an error if the document language is not specified.

About reflow

Identifying and assigning the intended read order of layout elements in a document is also useful for Adobe PDF documents that are intended to be read on more than one size of display device, making it possible to reflow the document content onto a much smaller PDA or other handheld device. Tagged Adobe PDF uses the same kind of structural information that provides accessibility to allow documents to be reflowed.

It is important to note, however, that the reflow order, or Page Content (PC) Order, is based on the order in which page elements are to be displayed on a computer screen, and may not follow the logical order, or read order, in every respect.

For this reason, while reflowing a document is suggested as one way of checking a document for proper structure, if a problem area is identified, it is essential that the logical structure of the document be modified, and not the PC order only. The logical structure can be modified only by changing the order of elements in the logical structure tree displayed in the Tags palette of Acrobat 5.0.

While Acrobat 5.0 also provides a TouchUp Order tool to fix reflow problems (problems effecting the order in which page elements are displayed), this tool does not effect the read order and should not be used to try to correct read order problems.

Creating Tagged Adobe PDF

The ideal way to create tagged Adobe PDF is to author the original document using an authoring application that creates tagged Adobe PDF. Currently, these applications include Adobe PageMaker® 7.0 and Microsoft Office 2000 applications on the Windows platform. Adobe intends to support tagged PDF creation in other authoring applications such as FrameMaker® and InDesign™.

If you do not have access to these applications, you can still create tagged Adobe PDF by using the Make Accessible plug-in, or for high-volume scanning operations, Capture 3.0 and Tag Adobe PDF agent.

See *How to Create Accessible Adobe PDF Files* for more information on using these methods to create tagged Adobe PDF.

In this guide, you will also learn how to create a tagged Adobe PDF file manually. This method should be used only as a last resort and only for the simplest of documents. Manually tagged Adobe PDF files are not reflowable.

About Element Tags

Tagged Adobe PDF files, whether created from an authoring application that creates tagged Adobe PDF, the Make Accessible Plug-in, or the Web Capture feature of Acrobat 5.0, contain structural information regarding the ordering of page or layout elements. In most cases, the order of the various elements will be correct. But in some cases, it may be necessary to modify the order of these elements to improve read order.

The structure of a document is represented in Acrobat 5.0's Tags palette by a logical structure tree comprised of a series of hierarchically arranged nodes. Each node or line in the logical structure tree represents either a page element, such as a block of text or an image, or a higher level element, such as a section or article, that contains these page elements. The type of element each node represents is indicated by a tag, or element name. In most cases, where the element represents some text content of the page, a portion of the document content will be displayed beneath the element name to further aid in identifying the element.

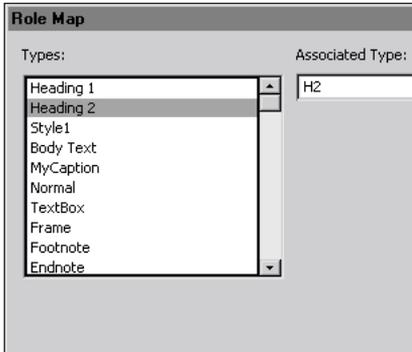
Names of Elements and Role Mapping

The tag name of elements in the logical structure tree are determined by the authoring application, and may vary from the standard element tag descriptors recognized by Acrobat 5.0. PDF provides a mechanism to map non-standard element types to Adobe standard tags.

A list and description of Adobe Standard Element Types is provided in Appendix A of this guide for reference.

When creating new tags in Acrobat, users are strongly encouraged to select a tag type from the available set of Adobe standard element types. When creating structure elements in authoring applications, it is recommended that structure elements be specified as or role mapped to Adobe standard element types to provide proper processing by accessibility tools and more faithful repurposing of PDF document contents.

Acrobat 5.0 allows you to view the relationship between imported tags and standard Adobe tag types by viewing or editing the tag type Role Map.



Role Map dialog box in Acrobat 5.0

To view or edit the Role Map

- 1 From the Tags menu at the top of the Tags palette, choose Edit Role Map.
- 2 Select a tag type from the list of imported tag types on the left. The current role to which the selected tag type is mapped is displayed in the Associated Type box.
- 3 To associate the tag with a different standard Adobe tag type, select a new type from the Associated Type drop-down menu.
- 4 When finished viewing or editing the Role Map, click OK.
- 5 If you have made any changes to the Role Map, save the document to save the changes to the Role Map.

Element Hierarchy

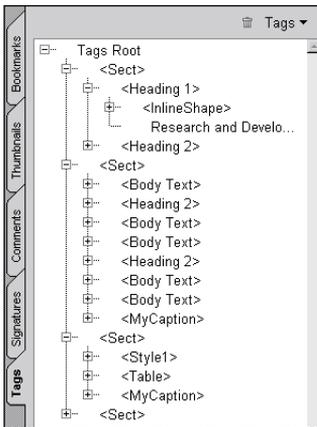
The structure and ordering of elements in tagged Adobe PDF is designed to provide accessibility and allow PDF documents to be reflowed while retaining the correct display or reading order. The order in which various elements will be read by a screen reader is represented by the logical structure tree in the Tags palette. (For information on seeing the reflow order, see “Reflowing a document” on page 15).

The logical structure tree of a Tagged Adobe PDF begins with a root level, or Tags Root, followed by the highest level of element as its immediate child, which is generally a container element representing the entire document or a section of the document. Elements are nested in hierarchical order.

An element which has one or more elements nested beneath it is called a parent element, and the nested elements are its children. Child elements on the same level under a parent element are referred to as “siblings.”

It is important to note that some attributes of a parent element are inherited by its children (such as language). Also, if a parent element is moved or deleted, all of its children will be moved or deleted as well.

A plus sign next to an element in the logical structure tree indicates that the element has children that are currently hidden. Clicking the plus sign will display all of the children on the next level. Child elements may also have children, which will be indicated by a plus sign. It is always good to open all of the child level elements before changing properties or deleting an element to see what other lower level child elements may be affected by any changes.



Structural element tags displayed as a hierarchical logical structure tree in the Tags palette of Acrobat 5.0

About Artifacts

A page element that is not an intrinsic part of the document is called an artifact. Artifacts can be any of the following:

- Comments such as online notes, graphic markups, and text markups.
- Pagination artifacts such as page numbers and running heads.
- Layout and typographic artifacts such as colored bars or rules between columns of text, horizontal rules separating footnotes from the text, and other decorative ornaments that do not convert to words.
- Printing artifacts such as crop marks, registration marks, and the document name printed outside the crop marks.

Artifacts are not referenced in the logical structure tree, because they're not considered useful when repurposing the document's content. For example, when a document's content is read by a screen reader, running headers are not considered useful information for users and need not be read.

About Element Properties

The properties of any element can be viewed and edited using the Element Properties dialog box. Editable properties include the name or title of the element, the tag type, the actual text, alternate text, and the language of the element. The last two of these are of special importance when creating accessible PDF. By supplying alternate text, for example, a description of an image or figure can be provided. The properties that can be edited are as follows:

- **Title** – This is the identifying text that will appear next to the element tag in the logical structure tree.
- **Type** – This defines the element tag type that will appear in the logical structure tree. For best results, it is suggested that you change only to one of the Adobe standard element types.
- **Actual Text** – This is the actual text that will be sent to a screen reader. If text is entered, the entered text will be read, and not the text that may comprise the displayed document content. This property should be entered only if you want something other than the content of an element to be read by a screen reader. Basically, this is replacement text.
- **Alternate Text** – This is additional or descriptive text that can be used to describe an image, formula, or other item in the document that does not translate naturally into text. For example, a pie chart may be described giving the values of the various sectors to provide users with visual impairments with more detailed information than generally available from a simple caption.

- **Language** – This identifies the language in which the document is written. In addition to the high-level language for the document, individual elements containing content in a language other than the main document language may be set to a different language to signal a screen reader to use an alternate pronunciation scheme, or to identify alternate hyphenation schemes for various languages.

To view or edit element properties:

- 1 Select an element in the Tags palette.
- 2 Choose Edit > Properties.
- 3 Enter or change properties.
- 4 Click OK.

About Class Maps

Elements may have more properties or attributes than the ones shown in the Main Properties area of the Element Properties dialog box. These values are defined and stored in a style dictionary or Class Map.

To view or edit the Class Map

- 1 From the Tags menu, choose Edit Class Map.
- 2 Any available style dictionaries, along with a listing of shared attributes will be displayed in the Classes and Attributes list.
- 3 Click on the arrow next to Class or Attribute to expand the item.
- 4 The editing command buttons will be enabled for any allowable editing functions as you select various classes and attributes.
- 5 When you are finished, click OK.

***Note:** You cannot create a new style dictionary in Acrobat, nor can you edit some attributes such as array objects.*

Checking the Read Order of Documents

When using the Make Accessible plug-in to create tagged Adobe PDF from legacy PDF files, the Make Accessible plug-in attempts to analyze the document and determine the intended structure. Since some Adobe PDF documents have complex layouts or large numbers of graphics, it is very important to always review documents that have been converted to tagged Adobe PDF to insure the file has been properly optimized for accessibility.

Acrobat 5.0 provides an Accessibility Checker to analyze a tagged Adobe PDF and look for problems that may exist. For information on using the Accessibility Checker, see *How to Create Accessible Adobe PDF Files* and Acrobat 5.0 Online Help.

In addition to using the Accessibility Checker, you should test the read order of your document by sending it to a screen reader, or reflowing the document.

Note: The TouchUp Order Tool is provided to fix reflow problems. It can be useful to locate structure problems that effect read order, but it should not be used try to correct these problems. Changing the display order in the TouchUp Order view will not change the read order displayed in the logical structure tree.

Acrobat 5.0 also allows you to save a file as RTF. If you do not have access to a screen reader, viewing an RTF version of your document will allow you to see the order in which text will be presented to a screen reader. As graphics are not exported, however, this is not a good way to check the accessibility of images.

Viewing the Logical Structure Tree

The logical structure of the document is displayed in the Tags palette. The various elements that have been tagged are displayed in descending, hierarchical order. Elements which have nested child elements are indicated by a plus sign in the box to the left of the element name.

To view the logical structure tree:

- 1 Choose Windows > Tags.
- 2 Drag the Tags palette to the Navigation pane.

To view nested elements, click the plus sign. All child elements on the next level will be displayed. A child element may also include one or more children, which is similarly indicated by plus signs on lower levels of the structure hierarchy.

If the current document is not a tagged Adobe PDF file, no structure will be displayed in the Tags palette. Instead, the message “No Tags available” will appear in the Tags palette.

Using accelerator keys

In addition to clicking on the boxes next to elements individually, Acrobat 5.0 provides a number of accelerator keys which can help move more quickly through the tree.

Arrow keys With an element selected, use the right arrow key to expand the element, and the left arrow key to close the element. The up and down arrow keys will move you up and down the structure tree, while the left and right arrow keys will move up and down a series of nested elements.

Plus and minus keys With an element selected, use the plus key on the number pad to expand an element, and the minus key to close it.

Forward slash and star key (Num pad) Use the star key on the number pad to open the tree to show all elements, and the forward slash key on the number pad to close the tree. Using the shift key in combination with the star key will open the tree one level at a time.

Control clicking Holding down the Ctrl key (Windows) or Command key (Mac® OS), will expand and close the entire tree.

Highlighting Structured Elements

It may not always be easy to determine which elements in the logical structure tree are related to content or text in the document. To view the association between elements displayed in the Tags panel and the actual page content, Associated Content Highlighting is provided in the Tags panel menu.

To highlight content

- 1 From the Tags palette menu, select Turn On Associated Content Highlighting.

- 2 Select an element in the Tags palette. The content associated with the selected tag will be highlighted in the document by a bounding box.

Note: When Turn On Associated Content Highlighting is on, other commands in the Tags panel menu will not be available. To restore the full menu, select Turn Off Associated Content Highlighting again to toggle it off.

To locate the element tag from a selection:

- 1 Use the Text Selection tool  or Graphic Selection tool  to select content within the document.
- 2 Choose Tags > Find Element From Selection. The logical structure tree will automatically scroll to allow you to view the highlighted element tag associated with the selection.

Reflowing a document

One way to check the order of elements in a document is to reflow the document using Acrobat 5.0's Reflow tool. The tagged Adobe PDF document is reflowed one page at a time in the document window. Acrobat doesn't save documents in a reflowed state.

To reflow a tagged Adobe PDF document:

In Acrobat, click the Reflow button  on the toolbar, or choose View > Reflow. To return to the view of the tagged Adobe PDF document in an unreflowed state, click the Actual Size button , the Fit in Window button , or the Fit Width  button on the toolbar, or choose a related command from the View menu.

If the reflowed document exhibits unexpected results, it may indicate a problem in the read order. Locate the problem area in the logical structure tree to determine how the structure could be improved or if there is another problem to be fixed, such as changing an element to an artifact to remove it from the read order. To fix actual reflow problems, however, you must use the TouchUp Order tool. For more information on fixing reflow problems, see Acrobat 5.0 Online Help.

Modifying the Structure of a Document

If you discover problems in the structure of a tagged Adobe PDF document that effect accessibility, you can use the commands in the Tags Palette menu to correct the structure by changing the order of elements, or creating new elements.

Rearranging elements in the structure tree

The logical structure tree displayed in the Tags palette provides an easy way to change the order in which elements are reflowed or read by a screen reader.

To change the order of an element:

- 1 In the Logical Structure Tree, click the element you want to move.
- 2 Drag the element to the desired position in the Logical Structure Tree.

Note: As you drag an element, the cursor will change to  to indicate legal moves, and  to indicate illegal moves. If you release the mouse button while the cursor is indicating an illegal move, the element will snap back to its original position in the tree.

At times you may want delete some portion of the logical structure tree and create your own tags and structure.

To delete an element:

- 1 Select an element in the Logical Structure Tree.
- 2 Do one of the following:
 - Select Delete Element from the Tags panel menu.
 - Right click on the selected menu to bring up a context menu, and select Delete Element.

Note: When you delete an element, the element and all of its children are removed from the structure tree. The associated page content is not deleted from the document. It does, however, become unmarked content.

Fixing improperly marked artifacts

An authoring application, Tag PDF Agent, or the Make Accessible plug-in may mark content as an artifact even though the content is important and needs to be made available for screen reading devices.

It is also possible that unimportant or distracting information has been marked as a structural element and you want to change it to an artifact so that a screen reader won't attempt to read it. Using the logical structure tree in the Tags palette, you can find and fix improperly marked artifacts.

To find artifacts:

- 1 Choose > Tags > Find Artifacts.
- 2 Select to search either current page or the entire document for artifacts.
- 3 Click Start Search.
- 4 When an artifact is found, it will be highlighted in the document view, and the type of artifact will be displayed.
- 5 To find more artifacts, click Find Next, or to end the search, close the Artifacts dialog box.

To change an artifact to an element:

- 1 With a high-level tag selected in the the logical structure tree, search the document for artifacts.
- 2 When the desired artifact is found, create an element tag by entering a Tag Title and selecting an Tag Type from the drop-down menu.
- 3 Click Create Tag. The new element tag will appear as the last child element of the selected element in the logical structure tree.

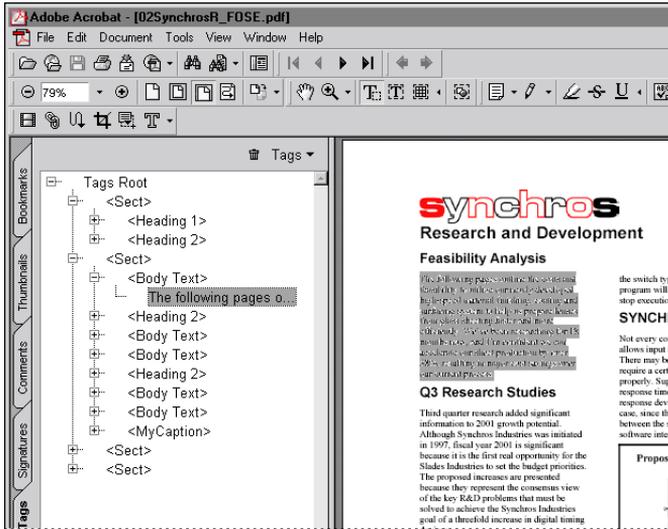
Note: Tagged content cannot be created as a child element of other tagged content. If you attempt to do so, you will be prompted to make another element selection in the logical structure tree before creating the new tagged element.

To change an element to an artifact:

- 1 Select the content in the logical structure tree (the content appears below the Tag and the tag title).
- 2 Choose Tags > Change Element to Artifact.
- 3 Choose an artifact type and a position on the page to attach the artifact.

4 Click OK.

Note: If you select the content in the document, the *Change Element to Artifact* command will not be enabled. You must select the tag content displayed under the element tag in the logical structure tree in the Tags palette to use this command.



Tag content is shown highlighted in the logical structure tree, with the corresponding page content shown highlighted in the document view.

Tagging manually to fix problem areas

Improperly marked content can create reflow and accessibility problems. If content has been tagged in such a way as to create problems for proper reflow or accessibility, you can manually mark content and assign element types to fix the problem.

If you are starting from scratch to manually tag an untagged Adobe PDF document, you must first create a Tags root. The Create Tags root command is available only if there are no available tags in the document.

To create a Tags root:

- 1 Open the Tags palette in the Navigation pane. (You may need to choose Windows > Tags and drag the Tags palette to the Navigation pane if the Tags palette isn't already present.)

Tagging manually to fix problem areas

- 2 With the Tags palette open, choose Create Tags root from the Tags menu.
- 3 With the Tags Root selected in the Tags palette, Choose Tags > New Child Element and select a high-level element such as Document.
- 4 Choose Tags > Document is Tagged Adobe PDF to set the Tagged Adobe PDF flag.
- 5 Save the document. Changes to the logical structure tree do not become part of the document until it is saved.

***Note:** It is important to set the Tagged Adobe PDF flag and save the document for the document to be recognized as Tagged Adobe PDF. To make the PDF “untagged,” it is only necessary to make sure Document is Tagged Adobe PDF is turned off (unchecked). No structural information will be lost when turning this option off, and the logical structure present in the file will still be accessible to screen readers.*

To create a new element:

- 1 Select New Child Element from the Tags palette menu.
- 2 Select the type of element you want to create from the pop-up menu.

***Note:** If an element is selected when you create a new child element, an empty structure element of the type selected will be inserted as the last child of the selected element. If no element is selected, or of the root tag is selected, the new element will be inserted as a top-level element.*

- 3 If necessary, drag the new element to the desired position in the logical structure tree, and save the document.

To add page content to the logical structure tree:

- 1 Choose Tags > Find Unmarked Content to locate any content that has not already been added to the logical structure tree.
- 2 Click Find Next. Any unmarked content found will be highlighted in the document view.
- 3 Create the appropriate element type for the content by selecting a standard element type from the Tag Type drop-down menu.
- 4 Enter a title for the tag, if desired.
- 5 Click Create Tag. Locate the newly created tag in the logical structure tree and, if necessary, drag it to the proper place in the tree.
- 6 Do one of the following:

- Click Find Next to repeat the operation.
- Close the Unmarked Content dialog box.

Selecting Text in Tagged Adobe PDF

Generally you will use one of the selection tools (Text Select Tool , or Column Select Tool ) to select text in the document. When the Document is Tagged Adobe PDF flag is on, however, selection results can sometimes be unpredictable. If you find you are having difficulty selecting the text you want, you can try turning the Document is Tagged Adobe PDF flag off.

To turn off Tagged Adobe PDF flag:

- 1 Choose Tags > Document is Tagged Adobe PDF.
- 2 Make sure the checkmark does not appear in front of the Document is Tagged Adobe PDF item in the Tags menu.
- 3 Save the document.

Adding and fixing links

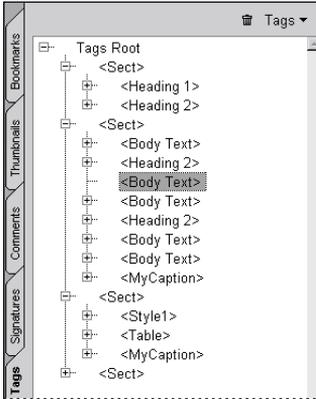
A link in an Adobe PDF document allows you to move to another part of the document or to another document. You can create new links in Acrobat 5.0 and add them to the logical structure tree so that a screen reader will be able to recognize and alert a user that there is an active link.

In most cases, the text for the link will be only a portion of the content associated with an existing element tag. In this case, you must first isolate the link text by creating separate child elements for the tag content that precedes and follows the link text, as well as a separate child element for the link text itself. The separate content items should appear in the proper order and at the same level under the original parent element.

Once the link text has been isolated, you can create a link using the Acrobat Link tool. You will then create a new element tag for the link itself and position it properly in the logical structure tree.

To add a link:

- 1 Select the link text or content in the document and choose Tags > Find Element From Selection to find the related element tag in the logical structure tree.
- 2 Select the content under the tag and press the Delete key to delete the content. The parent tag is now empty.
- 3 Click on the empty parent tag to select it.



Empty parent tag selected

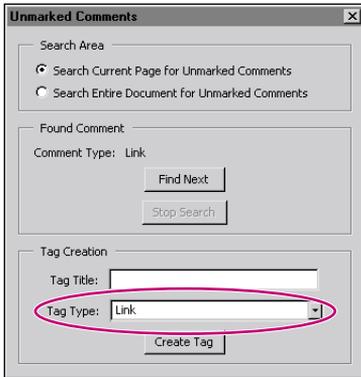
- 4 With the parent tag selected, use an appropriate selection tool to select the text or content that precedes the link text that you will use to create a link, and choose Tags > Create Child Element From Selection.



Hint: You may select the content in more than one piece. For example, if the link text appears in the middle of a sentence in the middle of a paragraph, and the entire paragraph was the original tag content of the currently selected parent tag, you might use the Column Select tool to select all of the text above the sentence containing the link text and create a child element from the selection, then use the text select tool to select all of the preceding text on the same line as the link text to create a second child element. Select the link text and create a third child element. Any text following the link text will become a fourth child element.

- 5 Continue to select text and create child elements until all of the original tag content has been tagged and displayed under the parent tag.

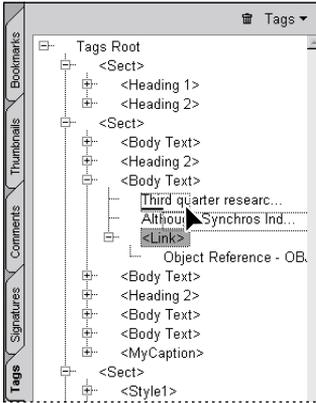
- 6 With the parent tag still selected in the logical structure tree, use the Link tool to create a link for the link text, and click Set Link when done.



Tag type Link selected in the Tag Creation dialog box.

- 7 Choose Tags > Find Unmarked Comments, and click Find Next until you find a comment of the type Link. Check the document view to make sure the link you just created is highlighted.
- 8 With the newly created link highlighted in the document view, create a tag by selecting Link in the Tag Creation pull-down menu, and click Create Tag.
- 9 A new link element and object reference should appear as the last child of the parent tag.

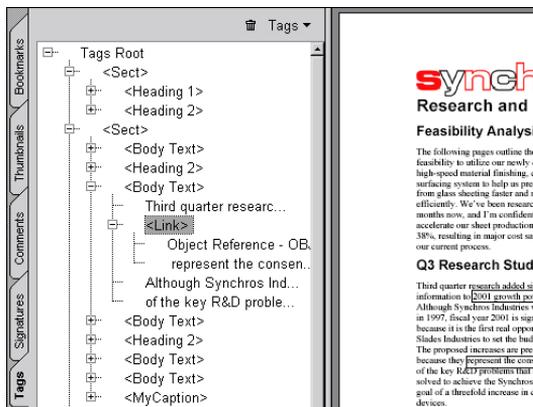
- 10** To place the new link element in the correct position in relation to the link text in the structure tree, click on the link tag (<Link>) and drag it into position directly above the link text. (The insertion line should be above and just to the left of the link text to move the link element tag to the proper position.)



Dragging the new link element to the proper position in the logical structure tree.

- 11** Finally, create the correct association between the link element and the link text content by dragging the link text so that it is inserted as a child to the link element directly below the object reference. When correctly positioned, the link element should be a child of the original parent element, and should have two child elements: an object reference element followed by the link text content.

12 Save the document to save changes to the logical structure tree.



The link object reference and tag content in the proper order under the link element tag.

Creating and tagging form fields

The Make Accessible plug-in will attempt to correctly identify and tag existing form fields in a PDF document. If, however, a document page contains multiple form fields, or if the layout is fairly complex, it may be difficult for Make Accessible to determine the proper order in which to display the form fields, or the relationship of the form fields to surrounding text.

If a document contains only a few form fields, you can optimize the structure by reordering the element tags in the logical structure tree. For more complex documents, however, you may want to delete any incorrect structures in the Tags palette and create form fields and tag them manually.

To create a form field:

- 1 Select the form tool .
- 2 Drag the cross-hair pointer to create a field of the required size.
- 3 In the Field Properties dialog box, enter a name in the Name text box, and select a format from the Type menu.
- 4 Be sure to enter a short description of the form field. This description will be passed to a screen reader to help visually impaired users identify the form field.
- 5 Select the options for your field type, and click OK.

- 6 Click the Appearance tab, and select attributes for the form field.
- 7 Select the hand tool  to display the finished form field.

For more information about creating form fields, see Acrobat 5.0 Online Help.

To create a form field element tag:

- 1 Choose Tags > Find Unmarked Comments.
- 2 Click Find Next until you find the form field for which you want to create an element tag in the logical structure tree. Form fields have the Comment type “Widget.” Make sure the correct form field is highlighted in the document view before continuing.
- 3 Use the Tag Type pop-up menu to set the tag type to Form.
- 4 Enter descriptive text in the Tag Title box. This title will help you identify the form field in the logical structure tree.
- 5 Select the desired parent element in the Tags palette, and choose Tags > Create Tag.
- 6 If necessary, drag the form element to the correct position among its siblings to ensure proper reading order.
- 7 Save the document to save changes to the logical structure tree.



Tag type set to Form.

Changing structure element types

In addition to changing an element's place in the structure hierarchy, you can change the element type.

To change an element's type:

- 1 Select the element you want to change in the Logical Structure tree.
- 2 Choose Edit > Properties.
- 3 Enter a new element type in the Type text box.
- 4 Click OK.

Working with figures and alternate text

Figures and other graphic data can be inaccessible to a user employing screen reading technology to access a PDF document unless descriptive text, also known as alternate text, is provided. Alternate text can be applied to figure elements (similar to the HTML Alt tag used with a Figure/Image element).

To provide alternate text for figures:

- 1 Select the figure element in the Logical Structure Tree.
- 2 Choose Edit > Properties, or right click and select Properties from the context menu.
- 3 Enter descriptive text in the Alternate Text text box.
- 4 Click OK.

Note: *If alternate text has not been provided for a figure, the Make Accessible plug-in may take any isolated text in the proximity of the figure and assign it as alternate text for the figure. If the text has not been otherwise left in the structure, it is best to create a new element for the text, and add appropriate alternate text for the figure as described above.*

Design Issues and Accessible Adobe PDF

The challenge of creating a fully accessible document is made even greater when working with complex layouts, or ambiguous layouts such as tables. The way computers display certain kinds of graphics, fonts, and layers of text and graphics can also create problems that can effect accessibility.

Working with tables

Tables pose a special challenge for screen readers and other assistive technologies, as by their nature, they present textual or numerical data in layout that can be easily referenced by sighted persons without having to read all of the information contained in the table.

Problems with tables can be minimized by always defining a table with rules around the table and between columns, rows, and cells. “White space” tables, where data is organized in a table format without rules, can create unexpected results, as it is difficult for the Make Accessible plug-in to readily recognize such layouts as tables.

It is also helpful to use table formats in the authoring application, such as table column heading, row heading, table cell data, etc.

Basically, a table should have the following hierarchy of elements: **<Table>**, a container element with attributes specifying placement and bounding box information; Table Row elements **<TR>** for each row of table cells as immediate children of the container element; a Table Data element **<TD>** or Table Heading element **<TH>** is the immediate child of each Table Row element, which contains the actual data or heading content of the associated table cell.

Tables commonly include merged cells to create a column or row heading that straddles or spans two or more associated columns or rows. For such a structure to be accurately reflected in the logical structure tree, **ColSpan** and/or **RowSpan** attributes should be properly set.



Hint: To see if these values are correct, open the Element properties dialog box and click on the arrow next to Attributes under Classes and Attributes. Further expand Attribute Objects until you find ColSpan and RowSpan. The value under these attribute keys represent the number of rows or cells each spans or straddles.)

Working with fonts

The most common problems encountered with fonts involve fonts that do not provide adequate encoding information or which use ligatures for special character combinations such as *ff*, *fl*, *fi*, *ft*, etc.

Some things you can do to eliminate font incompatibility problems include using the most recent version of Acrobat Distiller to create PDF, rather than using older releases of Acrobat Distiller. PDFWriter should not be used.

Also, it can be helpful to perform the conversion using Acrobat Distiller® on the same machine and system that was used to author the original document. This will eliminate problems that may arise if Acrobat Distiller is forced to substitute fonts because it is unable to find the font that was used in the creation of the document.

Note: In some cases, TrueType fonts that have gone through a PostScript® driver and Acrobat Distiller can no longer be searched, copied, cut, or pasted. This may also occur in the case of some “protected” fonts. Using Acrobat Distiller on the same system the PostScript file was created on can help alleviate this problem. The Base 14 fonts and Type 1 fonts rarely have these problems and should be used whenever possible.

Finally, when designing and authoring a document that is intended to be accessible, it is best to set your authoring software to not use ligatures, if possible. If this is not possible, a global search and replace of ligatures with their respective individual letters should be attempted.

It is also possible to replace the ligatures with their individual letter equivalents following conversion to PDF using Acrobat’s Text TouchUp Tool, but this approach can be very time consuming.

Working with complex vector graphics

During conversion to Tagged Adobe PDF, a single figure (such as a pie chart) may produce a structure comprised of multiple text and graphic elements.

In such cases, you may want to select and delete all structure related to the figure in the logical structure tree and create a simplified structure manually.

One way to avoid this problem is to always provide a single-ruled bounding box around figures, illustrations, charts, and graphs before converting to PDF.

Working with complex layouts

Generally speaking, the more columns of text, tables, charts, figures, or sidebars encountered in a layout, the more challenging it is to determine the intended read order of elements.

In authoring documents, the simpler and more straight-forward the layout, the more accessible the document will be. If a more complex layout is necessary, avoid overusing unnecessary ornamentation such as background images, text overlaying graphics, overlapping figures, etc., which can result in confusing structure. Most problems created by complex layouts can be alleviated by manual tagging, but depending on the complexity and number of elements involved, this can be a tedious process.

Working with layers

When converting documents that contain multiple layers to Tagged Adobe PDF, layered objects may be arranged in the drawing order rather than the intended read order. This can create problems when the document is reflowed or read by a screen reader.

There is generally no good work around for this problem when it occurs.

About Tag Adobe PDF Agent

Tag Adobe PDF Agent tips

For users of Adobe Capture 3.0 for high-volume conversion of paper documents to PDF, Tag Adobe PDF Agent is available to allow tagging of PDF documents to be incorporated in an Adobe Capture workflow.

Some of the advantages to using Tag Adobe PDF agent include:

- Tag Adobe PDF Agent allows users to specify more than one language dictionary for processing documents.
- Tag Adobe PDF Agent recognizes Table of Contents and index layouts, and can automatically create hyperlinks to the appropriate views in the document.
- Tag Adobe PDF Agent has a Move Graphics to Artifact option, which can be especially useful when converting forms to Tagged Adobe PDF.

Foreign language processing

Currently, when using the Make Accessible plug-in with Acrobat 5.0, language processing of documents is limited to the dictionary for the version of Acrobat 5.0 that is being used. Using Adobe Capture 3.0 with the Tag Adobe PDF agent, users are not limited to the native language of the Acrobat user interface.

The Tag PDF Agent provides access to a number of dictionaries, and also allows more than one dictionary to be selected for processing multiple-language documents. For example, both English and French dictionaries could be selected for documents that contain both languages.

Scanned document processing

For users who process Searchable Image PDF documents (Image + Text), the Tag PDF Agent used in conjunction with Adobe Capture 3.0 provides OCR review features not available with Acrobat 5.0. The Quick Fix and Reviewer tools in Capture 3.0 allow a user to review and correct OCR suspects that might otherwise render poorer quality scans inaccessible.

Working with previously tagged content

When processing documents that have been previously tagged it is sometimes beneficial to remove the original tag information and completely replace its structure.

Once such case involves HTML source documents that have used a table format for multi-column text. When such documents are converted to Adobe PDF, either by using Web Capture or PDFMaker via Microsoft Word, paragraphs may be represented as multiple <TD> cell tags within a table, creating unreliable logical order. Using Tag PDF Agent, this structure can be removed and replaced with paragraph tags in the correct logical order.

Simple form processing

Another advantage of PDF Tag agent is its option to treat all graphics as artifacts for the purpose of making a document accessible.

This is especially useful when dealing with a form created with text on one side of the page, and graphics such as blank lines, check boxes, and the like on the other side of the page. In this case, the Make Accessible plug-in would tag the graphics as images and assign the nearest text as alternate text for the image.

Tag PDF agent, however, allows you to discard the graphics by treating them as artifacts, leaving the text tagged in the proper logical order, and therefore accessible by a screen reader. Using Acrobat 5.0, you could then create form fields for the graphics and link them to the appropriate text. For more information on how this is done, see “Creating and tagging form fields” on page 25 of this guide.

For more information about how Tag Adobe PDF Agent can help streamline your workflow, visit the Adobe Capture Web site at www.adobe.com/products/acrcapture.

Appendix: Adobe Standard Elements

About Adobe Standard Element Tag Types

The following is a list of the standard Adobe element tag types that appear in the Create a New element sub-menu of the Tags palette menu, as well as the Tag Type pop-up menu in the Element Properties dialog box. Adobe strongly encourages the use of these tag types to provide the best results when repurposing tagged content.

Block-level Elements

All elements that are comprised of text laid out in paragraph-like form are referred to as Block-level Elements. All block-level elements are part of the document's logical structure and are represented in the logical structure tree. Block-level elements can be further classified as follows:

Container Elements These are the highest level of element and provide grouping of other block-level elements:

- Article Element (**Art**)– A self-contained body of text considered to be a single narrative.
- Division Element (**Div**)– A generic block-level element or group of block-level elements.
- Document Element – (**Document**) The root element of a document's structure tree.
- Part Element (**Part**) – A part element can be used to define a large division of a document, and may group together smaller divisions, such as article elements, division elements, or section elements.
- Section Element (**Sect**) – A general container element type, comparable to Division `<DIV Class="Sect">` in HTML, which is usually a component of a Part element or an Article element.

Special Text Elements These identify text that is used in a special way other than the usual narrative paragraph. Examples include:

- Block Quote Element (**BlockQuote**) – One or more paragraphs of text attributed to someone other than the author of the immediate surrounding text.
- Caption Element (**Caption**) – A brief portion of text that describes a table or a figure. A caption element is different than a Label Element, which merely identifies, but does not describe a table or a figure.
- Index Element (**Index**) – A sequence of entries that contain both identifying text and reference elements that point out the occurrence of the text in the main body of the document.
- Table of Contents Element (**TOC**) – An element that can contain a structured list of items and labels identifying those items. A Table of Contents element can have its own discrete hierarchy.
- Table of Contents Item Element (**TOCI**) – An item contained in a list associated with a Table of Contents Element.

Heading and Paragraph Elements These are paragraph-like block-level elements which include specific level headings and generic paragraphs (**P**). A Heading Element (**H**) should appear as the first child of any higher level division. Six levels of heading (**H1 – H6**) are available for applications that don't hierarchically nest sections.

Label and List Elements These are special block-level structural elements that are used for structuring lists:

- List Element (**L**) – A List can be any sequence of items of like meaning or other relevance.
- List Item Element (**LI**) – Any one member of a List. The immediate child elements of a List Element should be List Item elements. A List Item Element may have as a child a Label Element (optional) and a Label Body Element (required).
- Label Element (**LBL**) – A Label can be a bullet, a name, or number that identifies and distinguishes an element from others in the same list.
- List Body Element (**LBody**) – The descriptive content of a List item.

Table Elements Table elements are special structural elements for structuring tables. They include:

- Table Element (**Table**) – A Table is two-dimensional arrangement or layout of data or text cells which contains Table Row elements as children, and may have a Caption element as its first or last child element.
- Table Row Element (**TR**) – One row of headings or data in a Table. A Table Row Element may contain Table Header Cell elements and Table Data Cell elements.
- Table Data Cell Element (**TD**) – A Table cell that contains non-header data.
- Table Header Cell Element (**TH**) – A Table cell that contains header text or data describing one or more rows or columns of a Table.

Inline-level Structural Elements Inline-level Structural elements are used to identify a span of text having specific styling or behavior. They are differentiated from Block-level Structural elements, and may be contained in or contain Block-level Structural elements. The standard Inline-Level Structural elements are:

- Bibliography Entry Element (**BibEntry**) – A description of where some cited information may be found, which may contain a Label element as a child element.
- Quote Entry Element (**Quote**) – An inline portion of text attributed to someone other than the author or the text surrounding it. It is different than a Block Quote, which is a whole paragraph or multiple paragraphs as opposed to inline text.
- Span Entry Element (**Span**) – Any inline segment of text. A common use of a Span Entry Element is to delimit text associated a given set of styling properties.

Special Inline-level Structural Elements Like the Inline-level Structural Elements already described, these elements are used to describe an inline portion of text that has a special styling or behavior. They include:

- Code Entry Element (**Code**) – Computer program text embedded within a document.
- Figure Entry Element (**Figure**) – A graphic or graphic representation associated with text.
- Form Entry Element (**Form**) – A PDF form annotation which can be or has been filled out.
- Formula Entry Element (**Formula**) – A mathematical formula.

- Link Entry Element (**Link**) – A hypertext link embedded within a document associated with a PDF link annotation that goes to another place in the same document, or to another document.
- Note Entry Element (**Note**) – Explanatory text, such as a footnote or endnote, which is referred to in the main body of text.
- Reference Entry Element (**Reference**) – A citation to text or data found elsewhere in the document.