



Authoring for
Accessibility and Reflow
in Microsoft Word
and Adobe Acrobat



Module 1: Introduction to Accessibility Issues



Introduction

This module introduces you to the challenges of creating accessible electronic documents for individuals with disabilities. Creating accessible documents requires that we rethink the way we present information. It requires using software that produces documents that can preserve the intended content ordering and provide alternate descriptions for visuals such as pie charts, tables, and graphics. Adobe Tagged PDF provides the functionality needed to support document accessibility.

During this course, you will learn guidelines to produce optimal Adobe Tagged PDF files from a particular authoring application, such as Adobe InDesign or Microsoft Word. The hands-on exercises included in this training course allow you to troubleshoot accessibility problems in a document so that the content sequence of the original is retained when the document is accessed through an alternate device such as a screen reader. You will also learn how to further enhance a Tagged PDF file using Adobe Acrobat's accessibility tools.

In addition to this course, you can also refer to the Web sites and other resources listed in the [References](#) section in each module for further material.

Learning Objectives

At the end of this module, you will be able to:

- Describe aspects of document design that can cause accessibility problems
- Describe the difficulties in presenting information that is normally in graphics and tables to someone who is visually impaired
- Understand that Tagged PDF provides structure information that is used by screen readers in presenting information to a visually-impaired person
- Understand the process for creating and troubleshooting accessible documents

References

- <http://access.adobe.com>
- <http://www.adobe.com/products/acrobat/solutionsacc.html>

Product Information

- *How To Create Accessible Adobe PDF Files*
 - PDF Version: <http://www.adobe.com/products/acrobat/pdfs/accessbooklet.pdf>
 - HTML Version: <http://access.adobe.com/booklet1.html>
- *Advanced Techniques for Creating Accessible Adobe PDF Files*
 - <http://www.adobe.com/products/acrobat/pdfs/CreateAccessibleAdvanced.pdf>
- White Paper: *Enhancing the accessibility of the Web with Adobe Acrobat software*

- PDF Version:
<http://www.adobe.com/products/acrobat/pdfs/accesswhitepaper.pdf>
- HTML Version: <http://access.adobe.com/whitepaper1.html>
- Section 508 Analysis:
http://www.adobe.com/products/acrobat/pdfs/Sect_5085.pdf
- *Acrobat 5.0 and Accessibility Frequently Asked Questions*
<http://www.adobe.com/products/acrobat/pdfs/Acro5AccessFAQ.pdf>
- Section 508 Press Release
<http://www.adobe.com/aboutadobe/pressroom/pressreleases/200106/20010621508.html>
- FOSE Press Release (discussing Acrobat 5.0 as it relates to Government and Accessibility)
<http://www.adobe.com/aboutadobe/pressroom/pressreleases/200103/20010320fose.html>

Product Demonstrations

- Adobe Acrobat 5.0 and Accessibility Video Series
 - Working with Microsoft Office 2000 Files
<http://www.adobe.com/products/acrobat/movie1.html>
 - Working with Existing PDF files
<http://www.adobe.com/products/acrobat/movie2.html>
 - Working with Forms
<http://www.adobe.com/products/acrobat/movie3.html>
 - Usability Enhancements of Acrobat 5.0
<http://www.adobe.com/products/acrobat/movie4.html>
 - Working with Screen Readers
<http://www.adobe.com/products/acrobat/movie5.html>

Downloads

- Acrobat 5.05 Update (you must have Acrobat 5.0 installed to use this)
 - <http://www.adobe.com/products/acrobat/update.html>
- Acrobat Reader 5.0 Download (the version that includes accessibility)
 - standard presentation page:
<http://www.adobe.com/products/acrobat/readstep.html>
 - text-only page:
<http://www.adobe.com/products/acrobat/alternate.html>
- Additional Adobe product downloads for accessibility
<http://access.adobe.com/downloads.html>

Section 508 Information

- <http://www.access-board.gov/sec508/guide>

- <http://www.section508.gov/>
- <http://www.access-board.gov/news/508-final.htm>
- <http://www.itpolicy.gsa.gov/cita/fap.htm>

Vendor Information

- Freedom Scientific Jaws (<http://www.freedomscientific.com>)
- GW Micro WindowEyes (<http://www.gwmicro.com>)
- Dolphin Oceanic HAL (<http://www.dolphinoceanic.com>)
- <http://www-3.ibm.com/able/overview.html>
- <http://www.microsoft.com/enable>

Contents

Topics	Exercises
Section 508 and Electronic Documents	
How Acrobat Helps With Accessibility	
Accessibility Design Considerations	

File to Download for Exercises in this Module

None



Section 508 and Electronic Documents

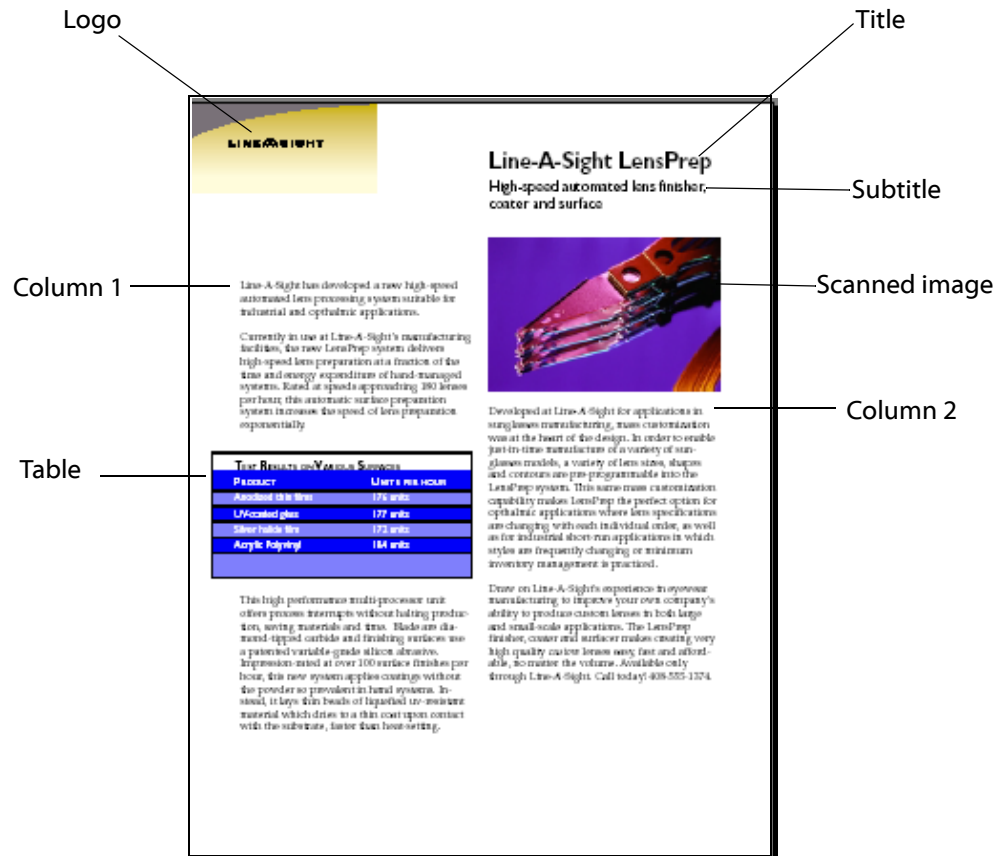
The World Wide Web has revolutionized the flow of information to people around the globe. However, not everyone has been able to partake in this ready access to information. In particular, individuals who are blind, have low visual acuity, or have motor impairment can find interacting with computer technologies challenging and frustrating.

Instructor Notes: Mention that Section 508 covers a broad range of disabilities in many situations; access to electronic documents is just one area. Because documents are by nature visual, the focus of this course is toward providing access to electronic documents for blind and visually impaired persons.

The Architectural and Transportation Barriers Compliance Board (Access Board) issued accessibility standards for electronic and information technology on December 21, 2000. This document is known as Section 508 and requires Federal agencies' electronic and information technology is accessible to people with disabilities. Information technology also encompasses electronic documents and the software used to read them.

Accessibility should be a relatively easy, low-effort addition to the documentation production process. This can be true provided that you consider accessibility during the document design phase. Most documents are geared towards sighted individuals. Layouts of information that interweave graphics and text with the intention of being read in a particular reading order can result in a confused presentation from the screen reader. A screen reader is a voice-synthesized program that converts the text of a document to an audible stream. People who rely on screen magnifiers to read information can only view a small portion of the document at a time and may have problems navigating a document with an artistic layout. Sighted people can easily grasp trends from data presented in graphs and tables, but that visual context is completely lost for a visually-impaired person.

The following document illustrates some of these points. Should the screen reader ignore the logo? Where should it start reading? Does the picture convey something that is important for non-sighted viewers to understand? If so, what is the best way to explain this in text? How can the information presented in the table be summarized effectively? In order for a vision-impaired reader to make sense of this document, we must somehow convey the structure as well as the content of the document. For example, what order should information be read. What should be said about the pictures and tables?



Proper presentation of information depends on several factors. Encoding the proper, or logical, reading order for a document is one of the most fundamental steps towards accessibility. This is especially critical for documents with columnated formats or where there are several distinct text blocks. Designing a document for accessibility means rethinking the message that you intend to convey with visuals. What is a table of data supposed to tell you if you cannot easily reference the row and column headings to give you context for the data contained within it? The bottom line is that the simpler a document's format and layout, the easier it will be to create a truly accessible document.

Instructor Notes: It is possible to make visually complex documents accessible, but it can take a great deal of work depending on the complexity of the document and the authoring software.

As equally important as the document is the software that a person with disabilities uses to access the document. The availability of the following features greatly enhances this user's interaction with software:

- Ability to process the logical read order of a document.

Encoding the logical read order in a document is useless if the software application cannot communicate that information to a screen reader.

- Enhanced keyboard navigation

Most visually-impaired users will not be able to utilize the point-and-click interface that is so ubiquitous in software. Likewise, motor-impaired users need succinct mechanisms for interacting with applications. Applications must include the capability of accessing menus, commands and program functions through keyboard shortcuts and keystroke combinations.

- Ability to reformat, or reflow, document content

Most of us are so accustomed to using large monitors to view Web pages, documents, and software applications that it never occurs to us what would happen if our view into the world was suddenly reduced to a three-inch window. Yet, this is exactly the challenge that users with low visual acuity face. Magnifying the font size is not a viable solution since the viewer must scroll widely in all directions to cover the document. Instead, an application should reflow the magnified text into the available screen space such that the text reformats itself to conform with that space. Text reflow is also used, for example, to reformat a document for display on a Palm Pilot, although that is not a primary concern for accessibility. Text reflow is a good example of a feature that benefits users who do not have disabilities. Text reflow and logical read order are related to accessibility but they are handled separately. This fact will be illustrated with examples later in the course.

Instructor Notes: Point out that logical read order and text reflow are two different issues. Logical read order is, or can be, auditory, while text reflow is a visual phenomenon. Most of the course will concern logical read order. Text reflow will be covered in the last module.



How Acrobat Helps With Accessibility

We mentioned before that an accessible document must be able to give a visually-impaired person cues about the reading order of the text and alternate explanations of graphics and possibly tables. Adobe Tagged Portable Document Format (PDF) provides these capabilities.

Converting documents to Tagged PDF is a robust, long-term solution for creating accessible documents. Tagged PDF represents various components of a document, such as chapters, heading styles, blocks of text, tables, graphics, and so on, as tag elements. The tag structure is similar to markup languages such as HTML and XML. A document's structure is represented as a hierarchy of tag elements. The order in the hierarchy represents the reading order of the document. Since the content is represented with tag elements, other applications can extract the information and reuse it for other purposes. For additional technical information about Tagged PDF, please refer to section 9.7 Tagged PDF in the PDF Reference, Third Edition, version 1.4.

Instructor Notes: PDF Tags are essential for accessibility because they provide the hooks into the document that other applications, such as screen readers, can use. However, tags are only part of the total equation in the sense that the application must decide how to make use of the tag information provided to it. The document developer has no direct control over this part of the equation; screen readers and other accessible devices evolve and become more sophisticated over time. The primary goal of the document developer in terms of accessibility is to make sure that the available hooks are in place so that applications can access them for information about the document. PDF Tags generally provide more document information than is currently used by alternate access devices. Tags placed in a document today will be utilized in increasingly sophisticated ways as access technology evolves over time.

Tagged PDF offers the following benefits

- You can associate additional information with a particular tag element, such as a graphic, by filling in its alternate text property.
- Other applications can automatically reflow text and associated graphics to fit a page of a different size than was assumed for the original layout.
- The document's content can be converted to other common file formats (such as RTF, HTML, and XML) while preserving the structure and basic style information.
- It retains one of the primary advantages of PDF, the ability to preserve 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.

Instructor Notes: The core of PDF tag support is that the logical order of the content of the document can be determined independently of the document's visual appearance and layout by traversing the tag structure hierarchy and presenting the contents of each node, which is what devices like screen readers do.

Tagged PDF documents are readily available to the general public since there are more than four hundred million copies of the free Adobe Acrobat Reader in use. Over two hundred government agencies worldwide have adopted PDF. Adobe Acrobat 5.0 improves the accessibility of functionality in the Adobe Acrobat software and the information in Adobe Portable Document Format (PDF) files.

Acrobat assists a user who wishes to read an accessible document by providing the following features:

- Provides usability enhancements, including enhanced keyboard shortcuts, support for high-contrast viewing, and the ability to zoom in and reflow text on the screen.
- Supports screen readers: Provides direct support for screen readers via the Microsoft Active Accessibility (MSAA) application programming interface (API) for Windows.® MSAA enables Acrobat 5.0 to integrate with assistive technology products including the newest versions of screen readers from vendors such as Freedom Scientific (<http://www.freedomscientific.com>), GW Micro (<http://www.gwmicro.com>), and Dolphin Oceanic (<http://www.dolphinoceanic.com>).

For users who wish to create accessible documents, Acrobat provides the features described below to both create and manipulate PDF files for accessibility:

- Optimizes legacy Adobe PDF files for accessibility: Transforms an untagged Adobe PDF file into a Tagged Adobe PDF file via the Make Accessible plug-in for Acrobat 5.0 for Windows (<http://www.adobe.com/support/downloads/detail.jsp?hexID=88de>). Resulting files can be read more clearly with assistive technology.
- Creates Tagged Adobe PDF files when converting Microsoft Office 2000 files, MS Office XP files, Adobe InDesign 2.0, Adobe FrameMaker 7.0, and so on to Adobe PDF, making it easier for people who use screen reader software to navigate a document in the proper reading order.
- Includes tools to help troubleshoot and optimize Adobe PDF files for accessibility, including the accessibility checker, the Tags palette, and more. The Tags palette lets an author review and touch up document reading order and add new tag elements to the document structure, such as form fields.
- Repurposes text: 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 reused with a variety of applications including word processors and non-MSAA-compliant screen readers.
- Secures files while retaining accessibility: Provides new levels of security that enable an author to prevent copying and pasting of document content while still making it accessible to assistive technologies via MSAA.

In terms of the accessibility requirements for software applications, mentioned in the previous section, Adobe Reader 5.0 and Acrobat 5.0 provide the following forms of support:

- Keyboard navigation

Reader 5.0 provides users with keyboard shortcuts and key combinations for access to all program functions. In Acrobat 5.05, functionality that is also found in Reader is fully accessible. However, some functions unique to Acrobat 5.05, such as the freehand drawing tools used for electronic markup, are not available via keyboard.

- Logical order

The sequential order of the PDF Tags structure determines the order in which the elements of the document, headings, paragraphs, alternate text for graphics, and so on, are processed by devices like screen readers.

At the present time, only some of the information present in the Tags structure is available to screen readers. Consequently, screen readers primarily use the sequential positioning of elements in the PDF Tags structure. In the future, one would expect such devices, in conjunction with improvements in authoring software, to make use of additional structural information about the document, such as table heading tags and lists, to provide additional context about the document to visually-impaired users.

For that reason we encourage adding formal structure to electronic documents — selecting paragraph styles rather than formatting text with tabs, spaces, and carriage returns, for example.

- Content reflow

Acrobat 5.0 provides a TouchUp Order Tool for specifying the flow order of elements of the document when it is repositioned into a different layout or format. In regard to accessibility issues, the ability to reflow and reformat content, in addition to determining the sequential order in which that content is presented, allows a user with partially impaired eyesight to magnify the text in the document sufficiently to read it, and also to have it properly formatted within the viewing window and presented in the correct logical sequence.



Accessibility Design Considerations

Whether you're publishing a document in HTML, XML, 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 cannot 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 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.
- Authors need to be aware of the features within their authoring applications that support accessibility and make full use of them.

[Instructor Notes: Recurring theme: the advantage of controlling the PDF Tag structure from the original document using the original authoring software.](#)

We strongly recommend that you design for accessibility as much as possible in your documents and select publishing tools that support accessibility features. While there are tools that allow you to modify a Tagged PDF structure, they are currently time-consuming to use. Fixing a document's Tagged PDF can take anywhere from half an hour to an entire day, depending on the magnitude of accessibility problems that are found. In addition, it is not possible to save and reapply your fixes to a document's Tagged PDF on the next revision of the document. For example, you put in 4 hours of work fixing the Tagged PDF structure of a document. Three months later, your department finishes an update to that same document. You will have to redo the fixes on the revised document. Your goal is to minimize the amount of post-PDF processing that you need to do on each document.

[Instructor Notes: Software development in the area of structuring and tagging documents for repurposing and accessibility is probably at the first generation stage and will continue to evolve rapidly. If accessibility is a major concern in your documentation efforts, we suggest always using the latest versions of both your authoring software and Acrobat.](#)

We also recommend that you use the most recent versions of your authoring software. Older software does not do a good job creating Tagged PDF. Often, the resulting Tagged PDF requires a significant amount of work to fix problems that occur during the conversion process.

Document Design

In terms of actual document design, the most problematic areas are complex layouts and visual presentation of information. Visual presentation of information encompasses tables, pie charts, flowcharts, bar charts, graphics, pictures, and so on. If you intend on using a multi-columned layout, make sure that your authoring software has built-in tools to support this and that the content converts accurately to Tagged PDF. Layouts found in newspapers and magazines, where multiple unrelated articles appear on a page and continue on different pages, pose particular challenges. The authoring software must provide a mechanism to allow you to indicate the read order. In other words, you must be able to define where the continuation of each story can be found such that it is converted to Tagged PDF. Without this capability, you are faced with a very time-consuming task of defining the read order in the Tagged PDF so that it is accessible.

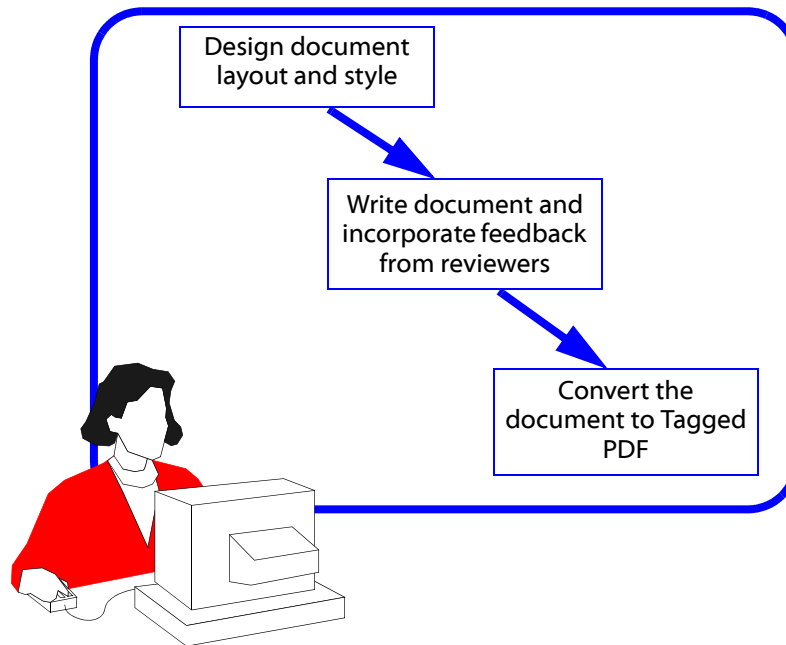
You should seriously evaluate the information that a visual is conveying. Is the information already present in the text? Is the visual simply providing color and images that are not essential to the message conveyed by the document? The alternate text that you tie to a visual should not repeat the caption. If the visual is a table, bar chart, pie chart, and so on, you should summarize the findings. For example, if a bar chart is showing the projected income over the next four quarters, state that and include the actual income figures in the alternate text. If a table is summarizing information, such as product requirements on different operating system platforms, you may want to consider including it in the text of your document.

Because accessibility is such a new area, the available authoring software varies in its capability to produce optimal Tagged PDF. For example, the work required to retain the reading order for a visually complex document, such as a brochure, may vary from application to application. Unless the logical structure of such documents is carefully prepared, usually involving iterative testing, it is not unusual for the sequence of the Tagged PDF structure to be almost but not quite right in terms of conveying the logical flow of the document's contents. You are always better off adjusting the original document to produce the desired PDF Tag sequence. Otherwise, you will have to adjust the Tagged PDF structure using the Acrobat Tags Palette.

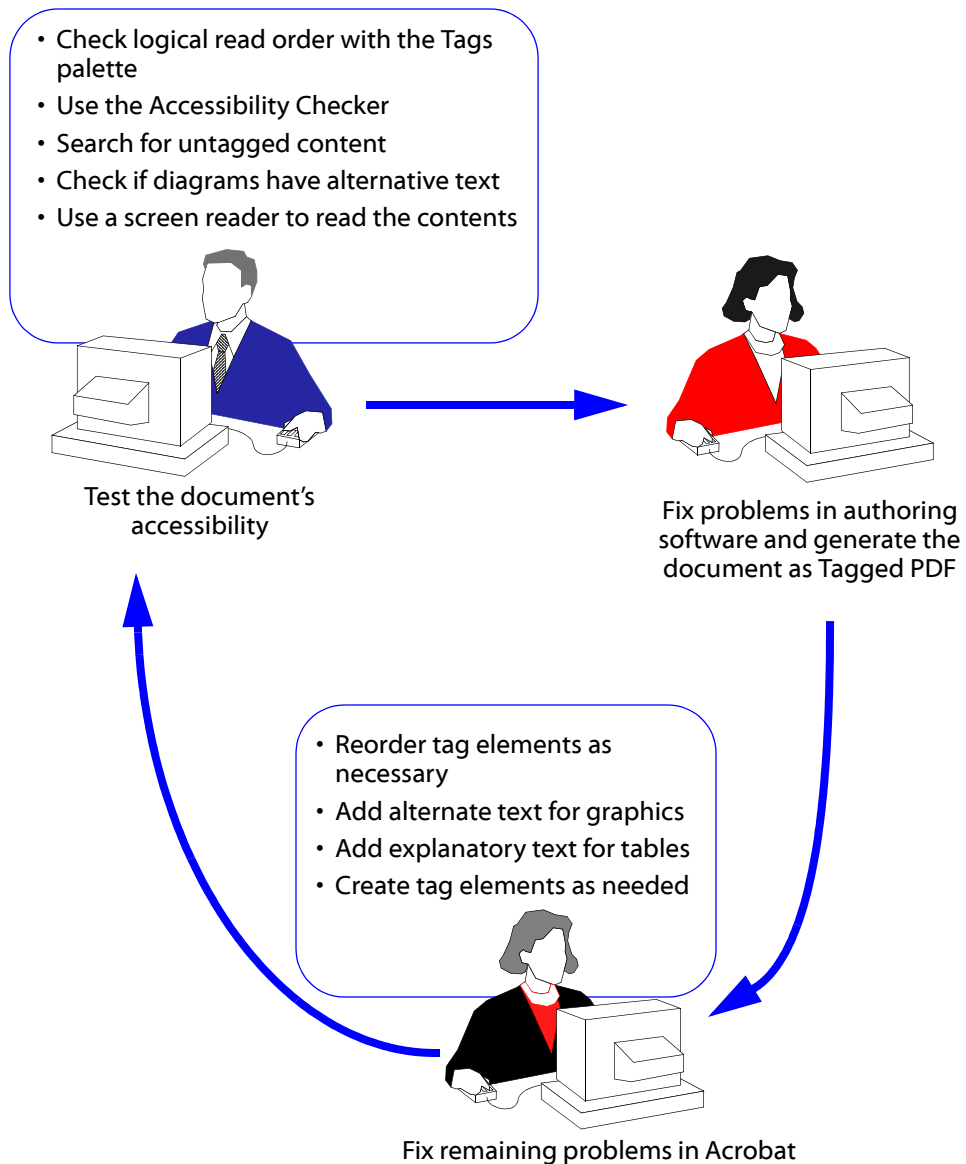
Workflow for Creating Accessible Documents

Creating an accessible document is an iterative process. Accessibility is still evolving. Expect that the techniques and guidelines will change over time as well as the capability of various software applications to support this process. However, the authoring software that you use as well as the types of documents your department produces will influence the workflow.

Within the authoring application, the user creates a document template that defines the overall look and appearance of the document. The template also includes the paragraph and character styles that will be used by the author. Once the document has been completed and reviewed, the author can generate a Tagged PDF file.

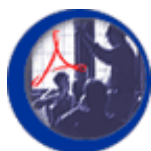


Once the author has created the Tagged PDF file, someone must verify that the content is actually accessible. The diagram below illustrates the workflow needed to fine tune a document's accessibility. Use Acrobat's Tags palette or save the file as **Text (Accessible)** to check the reading order. The Tags palette also includes other tools that allow you to search for page content that is causing problems, such as form fields and cross-reference links. You can use a screen reader to read the document the same way a visually-impaired person would access the document. Once accessibility problems have been identified, the author should return to the authoring application to fix as many of the problems as possible. Any remaining problems will have to be fixed using tools in Acrobat. For information on how to check a document's accessibility and fix problems within the authoring tool, refer to the next section, Module 2, in this course. For information on how to fix accessibility problems using Acrobat's tools, refer to [Module 3: Using the Tags Palette](#).



Before you release a document template for general distribution, test it with several existing documents that use a variety of different ways of presenting information. Work out any accessibility-related problems that you identify. This will eliminate the work that you will need to do post-processing the tagged PDF. There may be some issues that you identify that will have to be addressed by using alternate features within the authoring tool or by changes in people's work habits.

Module 2: Accessible Documents in MS Word



Introduction

This module introduces guidelines that you should use to create accessible documents in Word 2000 or higher. When you install Adobe Acrobat, it automatically installs PDFMaker into Word. You can configure PDFMaker to control a variety of different options that are used to produce a Tagged PDF document. You can use different tools and techniques to verify whether the resulting Tagged PDF file is truly accessible.

Learning Objectives

At the end of this module, you will be able to:

- Describe guidelines that you need to follow when you create documents in Microsoft® Word to produce an accessible document
- Configure PDFMaker to control the Tagged PDF file produced from Microsoft Word
- Troubleshoot accessibility problems using tools available in Adobe Acrobat
- Fix several different types of accessibility problems in a Microsoft Word source document

References

- Microsoft's Accessibility Web site
 - <http://www.microsoft.com/enable>
- *How to Create Accessible Documents*
 - PDF Version: <http://www.adobe.com/products/acrobat/pdfs/accessbooklet.pdf>
 - HTML Version: <http://access.adobe.com/booklet1.html>
- Adobe Acrobat 5.0 and Accessibility Video Series
 - Working with Microsoft Office 2000 Files
<http://www.adobe.com/products/acrobat/movie1.html>
- *Adobe® Acrobat® Help* (online document included with Acrobat 5)
- *Adobe PDFMaker Help* (online document included with PDFMaker, available by clicking **Help** in the PDFMaker dialog)

Contents

Topics	Exercises
Guidelines for Accessibility	
Generating a Tagged PDF Document	Creating a Tagged PDF from Word

Topics	Exercises
Testing Your Tagged PDF	Testing for Accessibility
	Correcting Problems in Word

File to Download for Exercises in this Module: `AccessWord.zip`


`AccessWord.zip` contains the following files:

- `UserGuideStart.doc`
- `Graphics.ppt`



Guidelines for Accessibility

When you install Acrobat 5.0, it installs Adobe PDFMaker 5.0 in your Windows Office 2000 or higher applications. When you use PDFMaker 5.0 with Word 2000 or higher, or any other Office application, you can create tagged Adobe PDF files that preserve hyperlinks, styles, bookmarks, and the structure of tables present in the source document. Make sure you are using Word 2000 or higher since it does a much better job of creating tagged PDF than its predecessors.

In general, if you want to use special formatting, such as a multicolumn layout for a document or present information in a structured format, you should make use of Word's tools. For example, instead of creating a table by using tabs to space out your rows, use either Word's **Table > Insert Table** or **Table > Draw Table** menu items to create a proper table. Likewise, if you want to create a two-column format for a page, do not use tabs to separate your columns. Use Word's Columns toolbar button . In addition to producing an accessible document, you will have a much easier time creating your document.

Traditionally, most lengthy documents include a table of contents. Tagged PDF documents can make use of the built-in bookmark feature. This can replace the traditional table of contents and utilize Acrobat's ability to present bookmarks side-by-side with document content. Bookmarks are also readily available to disabled users.

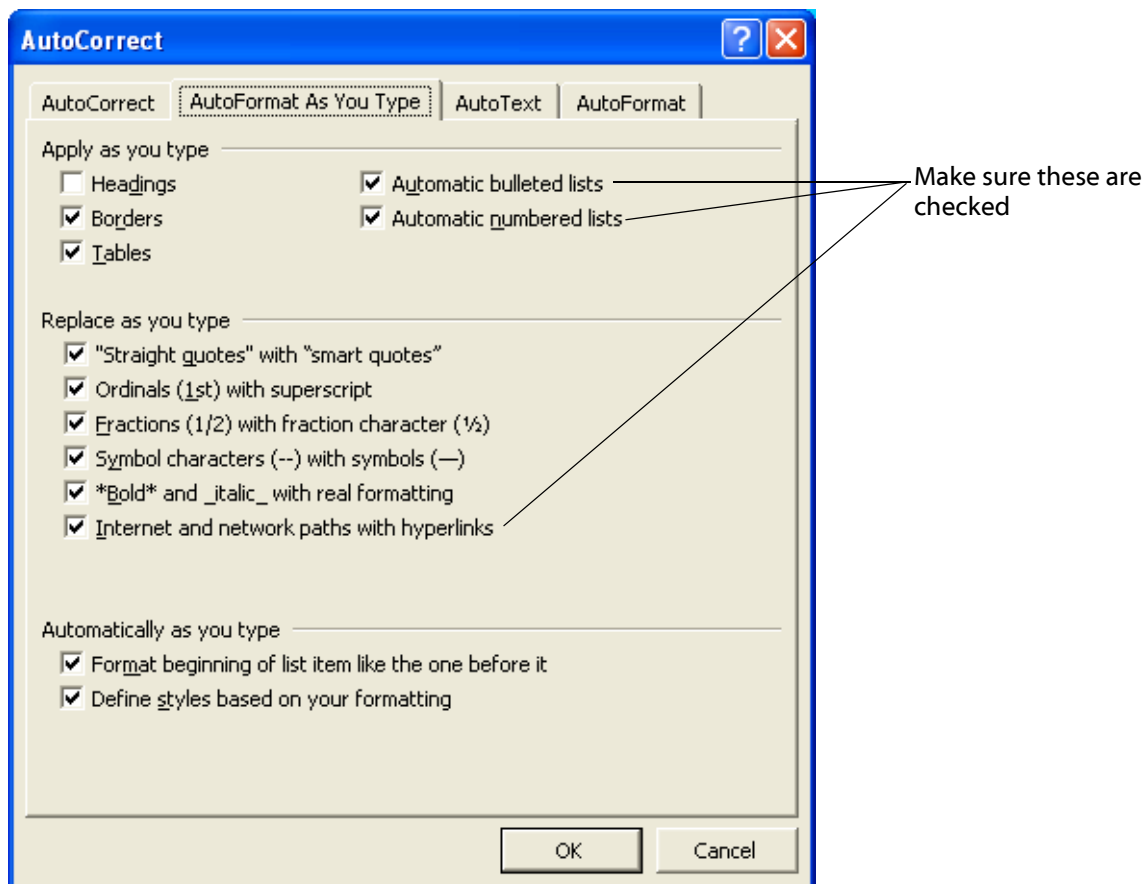
If you are like most Word users, you probably are not that diligent about using styles to format your document. By default, Word 2000 and 2002 are set up to automatically format your Word document. They will recognize numbered and bulleted lists and handle them correctly. They will automatically identify Internet, network and e-mail addresses and format them as hyperlinks. Using automatic formatting results in a cleaner tagged PDF file that will not require much postprocessing for accessibility.

To verify that automatic formatting is on, do the following:

1. Bring up Word 2000 or 2002. Select **Tools > AutoCorrect**. The AutoCorrect dialog appears.
2. Click on the **Autoformat As You Type** tab. Make sure the following are checked:
 - Automatic bulleted lists
 - Automatic numbered lists
 - Internet and network paths with hyperlinks
 - Define styles based on your formatting

Instructor Notes: This is a good time to introduce one of the recurring themes in this module, that the key to effectively producing and maintaining accessible PDF documents generated from MS Word lies in the way the original Word document is formally structured. For example, using the Column tool bar button to create multiple columns gives PDFMaker additional information about how to structure the tags that represent lines of text in the columns. In this case, PDFMaker will know to order them line by line down the left column, then line by line down the

right column, the way the columns are naturally read, rather than reading each line straight across the two columns.



If you plan on using your tagged PDF file to generate the content in other file formats, such as XML or RTF, you should be much more deliberate in using styles. This is especially true for lists. Careful adherence to styles in your document will insure that the formatting information is correctly exported.

Graphics

With respect to accessibility, graphics are the most challenging aspect of a Word document. Word has a variety of different ways that you can insert graphics into your document: creating your graphics with the native graphics tool, importing visuals developed in other graphics applications, cutting graphic elements from an outside application and pasting into Word, and so on. However, unless you use the appropriate technique, a tagged PDF generated from a Word 2000 or 2002 document with graphics will not retain its proper reading order.

To insure that your graphics retain their proper reading order, you must make sure that the graphic has been inserted inline with the text. Word offers five different text wrapping styles: inline with text, square, tight, behind text, and in front of text. However, only inline retains the graphics' position relative to the document's text. The other wrapping styles are treated identically to floating objects.

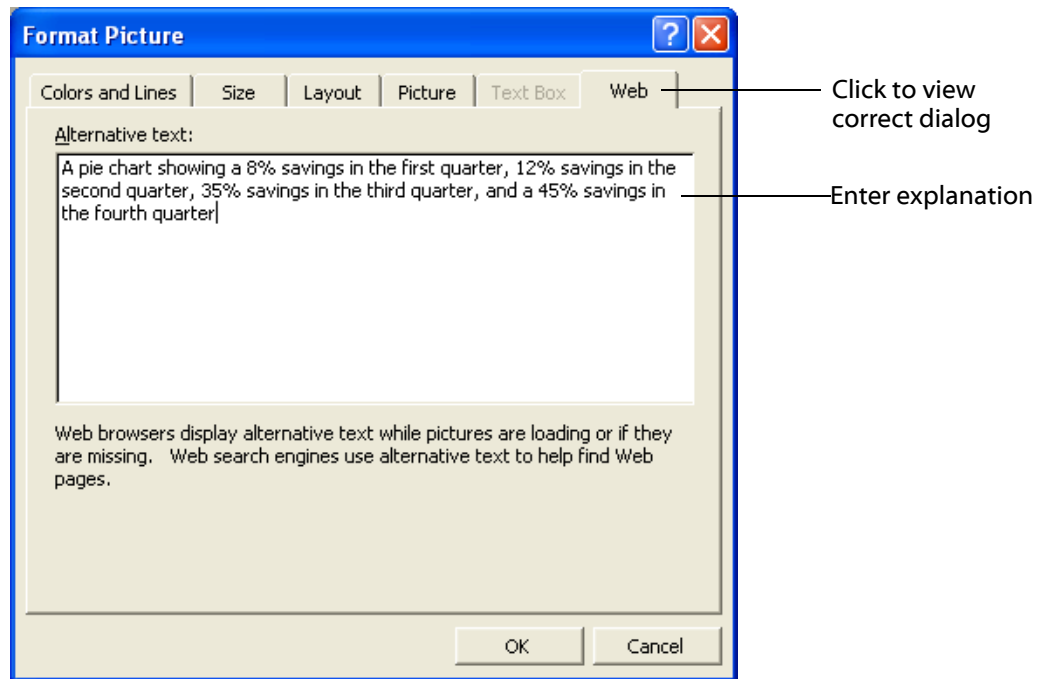
The simplest way to insert a graphic, and have it remember its place in the document, is to use **Insert > Picture**. We strongly recommend that you compose any composite graphics, such as a bitmap screen capture with callouts, in an outside application and save it as a standard graphic format, such as .bmp, .tiff, .jpeg, and so on. For a complete list of supported file types, bring up Word, select **Insert > Picture** and view the choices in the **Files of Type** combo box.

An alternative approach is to copy the graphic and paste it into Word. By default, this operation does not insert the graphic as an inline image. Consequently, you will need to edit the properties of the graphic in order to change its text wrapping. To do this, place the mouse over the graphic and right click to bring up a popup menu. Select the **Format** menu item. Word displays the **Format** dialog. Click on the **Layout** tab and change the wrapping style to inline. For best results pasting graphics, use **Edit > Paste Special** instead of **Edit > Paste**. **Paste Special** allows you to select a particular encoding for your graphics, resulting in better control over the final image.

Instructor Notes: Point out that the Alternative Text for the inline image will be read by a screen reader at whatever point the graphic has been inserted into the document. The placement of a graphic in the visual stream of elements may or may not correspond to the appropriate placement for the graphic's alternative text in the logical read order.

Avoid using text boxes for simple labelled graphics or even as callouts. Avoid creating watermarks in your document since you must use a text box to create a watermark effect in a document. Do not paste graphics into a text box. Although a text box appears in a particular place in your document, they are considered floating objects on a page. This means that they have no fixed position. When PDFMaker generates a tagged PDF from a document that uses text boxes, those text boxes will be placed behind all other objects, like paragraphs and headings, on a page.

Once you have inserted your graphic, you can attach alternate text that will be used by a screen reader to provide an explanation to a visually-impaired person. Using the **Web** tab of the **Format** dialog of a picture, you can enter the desired text.





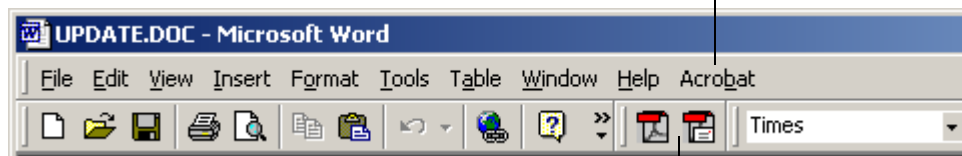
Generating a Tagged PDF Document

To allow authors to use PDFMaker with Microsoft Office applications, you should first install Office, then Acrobat 5, which automatically loads the PDFMaker macro. Among other things, this macro adds a top-level **Acrobat** menu to the Microsoft Office application's menu bar and a PDFMaker 5 tool bar to the set of available tool bars. PDFMaker provides access to and support for Distiller job options.

Acrobat tools in Microsoft Word application

Standard Microsoft Word tools

Acrobat 5 menu



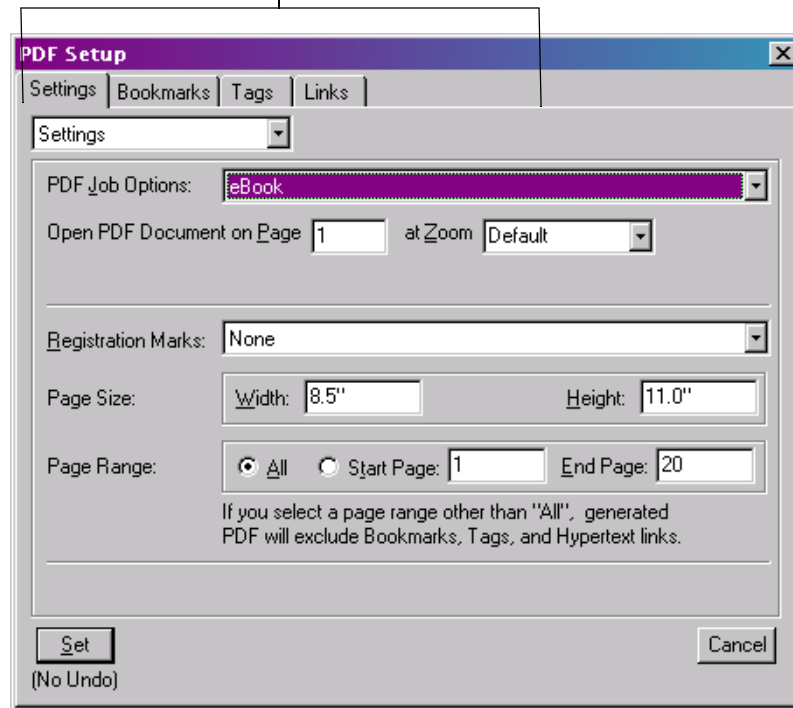
Acrobat 5 tools

To configure the PDF conversion process, you need to invoke the PDFMaker dialog by selecting **Acrobat > Change Conversion Settings**. The dialog contains five tabbed sections whose function are summarized below.

- Control the distiller options used to create the PDF using the **Settings** tab
- Specify security access to your PDF using the **Security** tab
- Control whether or not your PDF is a Tagged PDF as well as some of the document content that PDFMaker converts into tag elements using the **Office** tab
- Specify which styles are converted into bookmarks using the **Bookmarks** tab
- Define the opening view of the document using the **Display Options** tab

Each of the five tabbed sections are described in greater detail in the subsections that follow.

Configure PDF generation using these tabbed dialogs



PDFMaker Distiller Settings

PDFMaker has a powerful set of job options that users can customize should they need to change the quality or size of an Adobe PDF document. For example, they can select fonts to embed in the new Adobe PDF file, define how to compress images, provide color management information for images, define security settings, and customize the conversion process in many other ways.

PDFMaker has the following sets of predefined job options:

- **eBook** job options are for Adobe PDF files that will be read primarily on-screen—on desktop or laptop computers or eBook Readers, for example. This is PDFMaker's default job option set.
- **Screen** job options are for Adobe PDF files that will be displayed on the World Wide Web or an intranet, or that will be distributed through an e-mail system for on-screen viewing.
- **Print** job options are for Adobe PDF files that are intended for desktop printers, digital copiers, publishing on a CD-ROM, or to send to a client as a publishing proof.

- **Press** job options for are Adobe PDF files that will be printed as high-quality final output to an imagesetter or platesetter, for example.

The **Screen** or **eBook** sets are generally more appropriate for documents during the review stage. For more information on setting and customizing PDFMaker job options, see the "Acrobat Distiller Options" chapter in the Adobe Acrobat online Help.

PDFMaker Security Settings

PDFMaker makes it easy to control access to confidential information within Adobe PDF documents. Authors can choose to encrypt the document and specify the level of user access.

PDFMaker uses Acrobat's Standard Security method which supports 128-bit RC4 encryption, allowing authors to control access to documents using the highest level of protection possible. Authors can also set encryption to 40-bit RC4, which allows a secure Adobe document to be opened with earlier versions of Acrobat. Acrobat's security methods support blocking unauthorized users from repurposing or changing the document's content, including the addition of comments. They also let authors restrict printing entirely or allow printing only at low resolution.

The combination of password-protection and access permissions provides authors with a powerful security solution. Passwords prevent unauthorized people from accessing a document or modifying its security access.

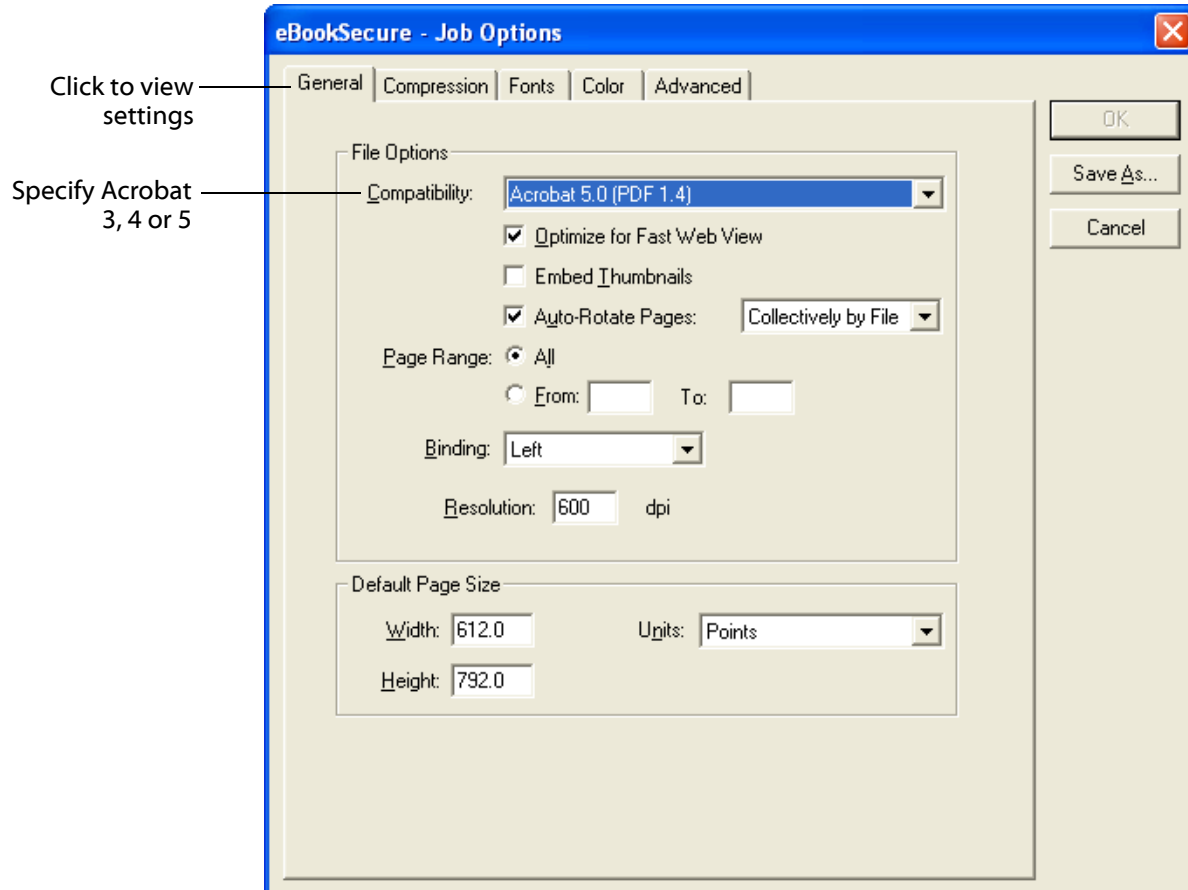
Using 128-bit Encryption

The 128-bit RC4 option is available only if you select Acrobat 5.0 compatibility in the General job options for PDFMaker. The default setting is compatible for Acrobat 4.0.

To modify the compatibility setting

1. In Word, bring up the PDFMaker dialog by selecting **Acrobat > Change Conversion Settings**.
2. Click the **Settings** tab.
3. Make sure you have the job option that you desire in the **Conversion Settings** combo box. Click **Edit Conversion Settings** to tailor the option.

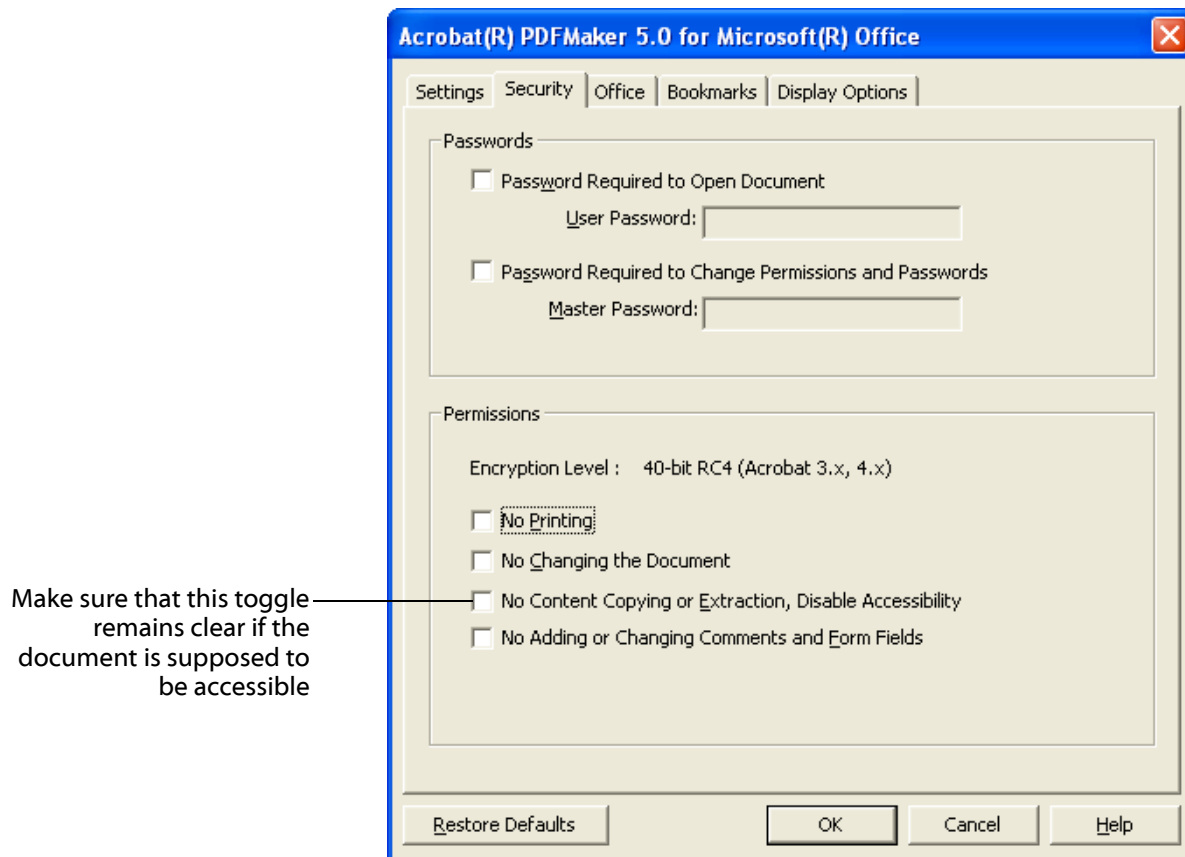
- The Job Options dialog appears. Click the **General** tab.



- Select Acrobat 5.0 in the **Compatibility** combo box. Click **OK** and either overwrite an existing job option or create a new one.

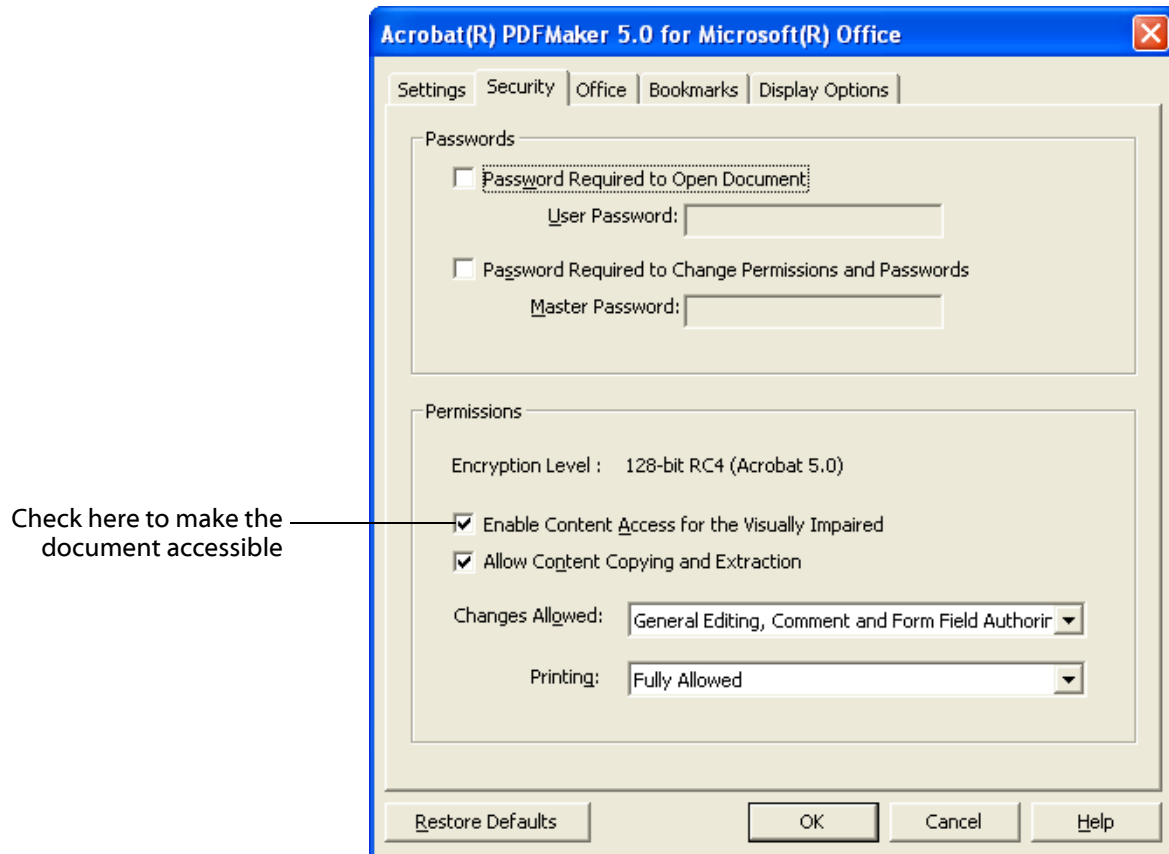
Security and Accessibility

You can inadvertently set up a document so that it is not only secure but also inaccessible. If you are using Acrobat 4.0 compatibility, the **No Content Copying or Extraction, Disable Accessibility** toggle must remain clear, as shown below.



If you are using Acrobat 5.0 compatibility, permission for copying or extracting content has been separated from accessibility. Consequently, you need to make sure that the **Enable Content Access for the Visually Impaired** is checked.

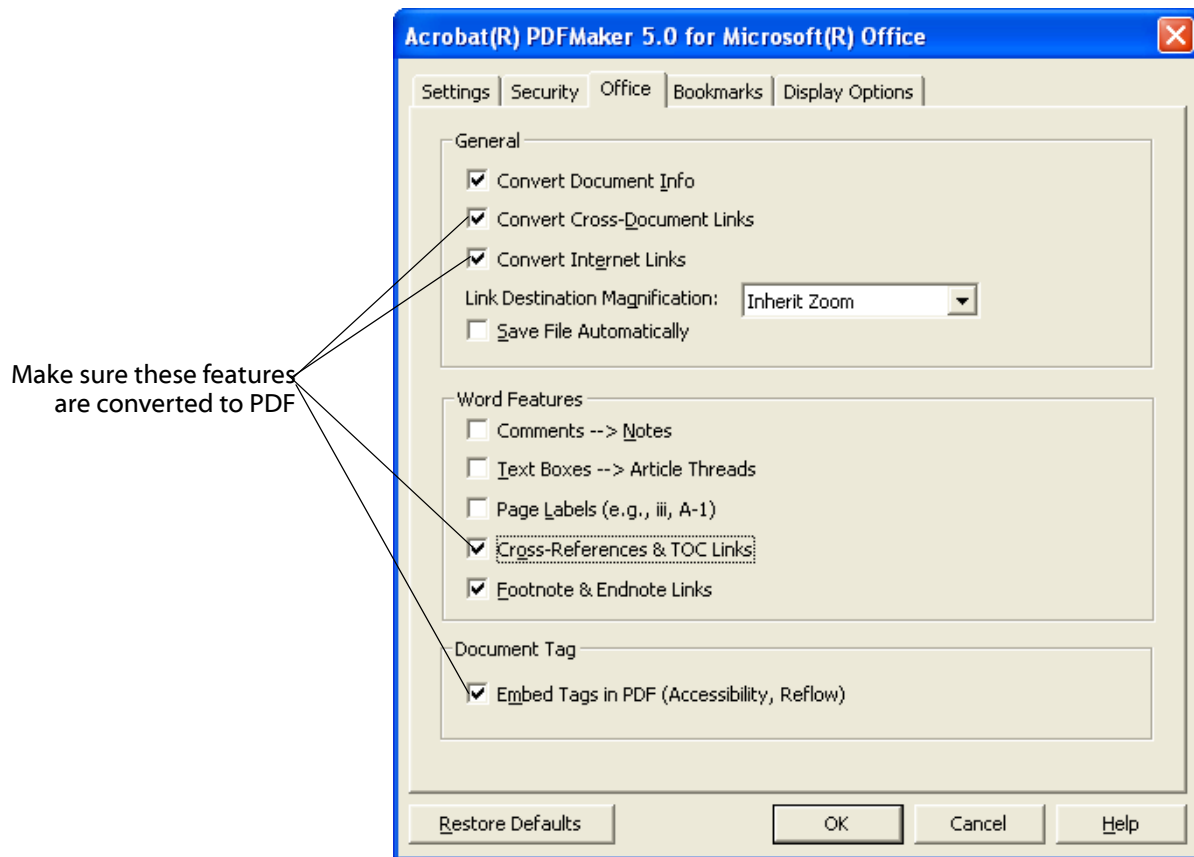
Instructor Notes: Make sure students understand how to avoid inadvertently blocking access to the file by others, including screen readers. A common question in Adobe related FAQ documents is, "Why can't so-and-so see the file I sent her?", and this is almost always related to overly-restrictive security settings.



PDF Generation Settings

PDFMaker has a set of options that users can customize to control the resulting PDF file. For example, you can choose to convert URLs and network addresses into links. Likewise, you can convert comments made in Word into Acrobat annotations. For accessibility, it is important to make sure that PDFMaker creates a Tagged PDF document, not a standard PDF. Since page headers and footers, including page numbers, do not convey much information to a visually-impaired person, you can direct PDFMaker not to convert page labels. Your decision to convert Word comments, linked text boxes, and footnotes largely depends on whether your document contains these features.

The list below summarizes the PDF configuration settings available in PDFMaker.



- Convert Document Info — convert document properties, such as author, subject, keywords, and so on, to their PDF equivalents.
- Convert Cross-Document Links — convert hyperlinks that point to other documents into PDF links
- Convert Internet Links — convert URLs to PDF Weblinks
- Link Destination Magnification — specify the opening magnification of a page for the destination of the link
 - Inherit Zoom — preserves the magnification setting of the original document
 - Fit Page — resizes the page to fit entirely in the Acrobat window
 - Fit Width — resizes the page so it fits the width of the Acrobat window
 - Fit Height — resizes the page so it fits the height of the Acrobat window
 - Fit Visible — resizes the page so its text and graphics are visible (width-wise) in the Acrobat window
- Save File Automatically — saves the Word file before the conversion process begins
- Comments --> Notes — converts Word comments to PDF comments.
- Text Boxes --> Article Threads — converts linked text boxes to PDF article threads
- Page Labels — converts Word page number formats, such as ix, v, A-1, and so on, to their PDF equivalents
- Cross-References & TOC Links — converts cross-references into PDF links, and convert table-of-contents entries into PDF links to their destinations.

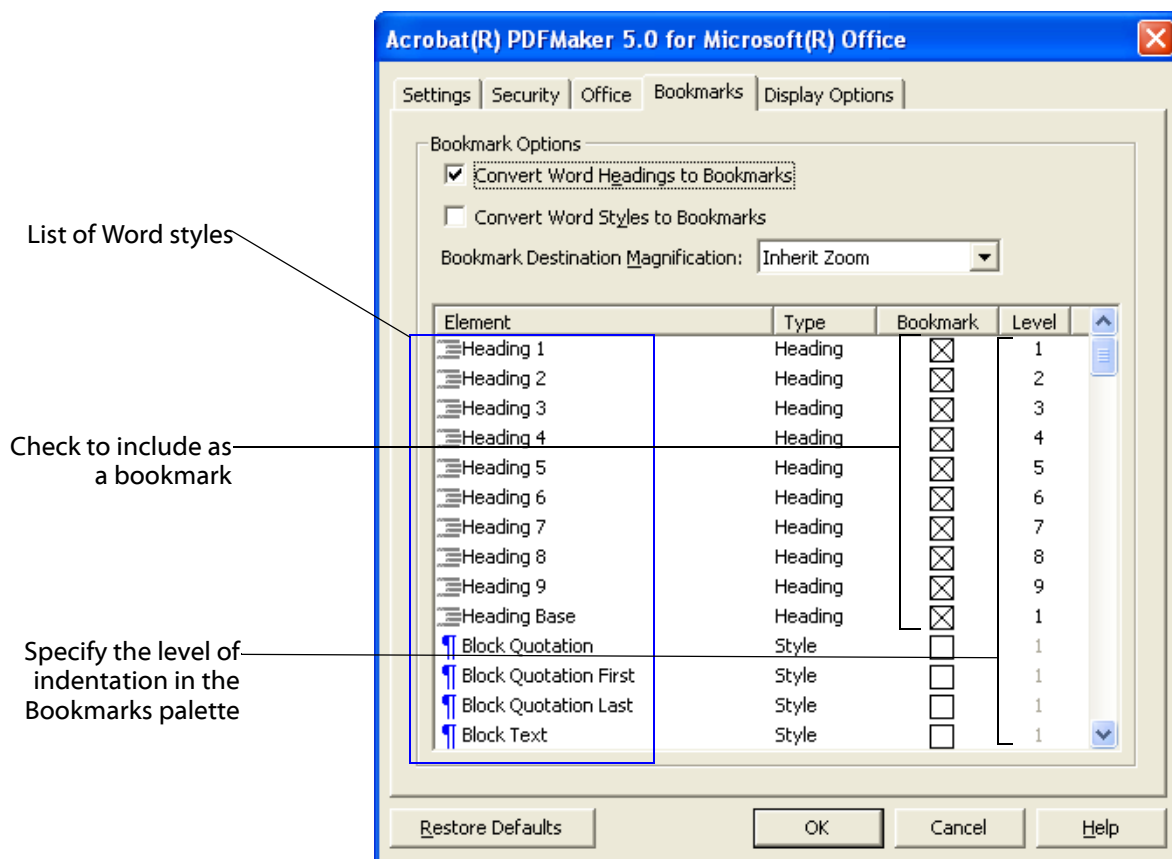
Instructor: Due to a bug in the current version of PDFMaker, cross references (including TOC entries) are not converted into link tag elements. In order to make cross references accessible, you must manually tag them. The module on the Tags palette includes an exercise that addresses this problem using the Find Unmarked Comment command. Mention this to students, this should be fixed in a later version of PDFMaker.

- Footnote and Endnote Links — converts footnotes and endnotes into PDF links to the respective citations
- Embed Tags in PDF — when checked, creates tag structure elements in the PDF file to preserve content ordering

Bookmark Settings

If you have a long document, you can define what styles will create bookmarks. A bookmark is a type of link with representative text in the Acrobat navigation pane. Each bookmark in the navigation pane goes to specified location in the document, typically the beginnings of a chapter or section. Most document bookmarks are virtually identical to a table of contents, allowing a user to easily view the organizational structure of the document as well as quickly jump to the desired spot.

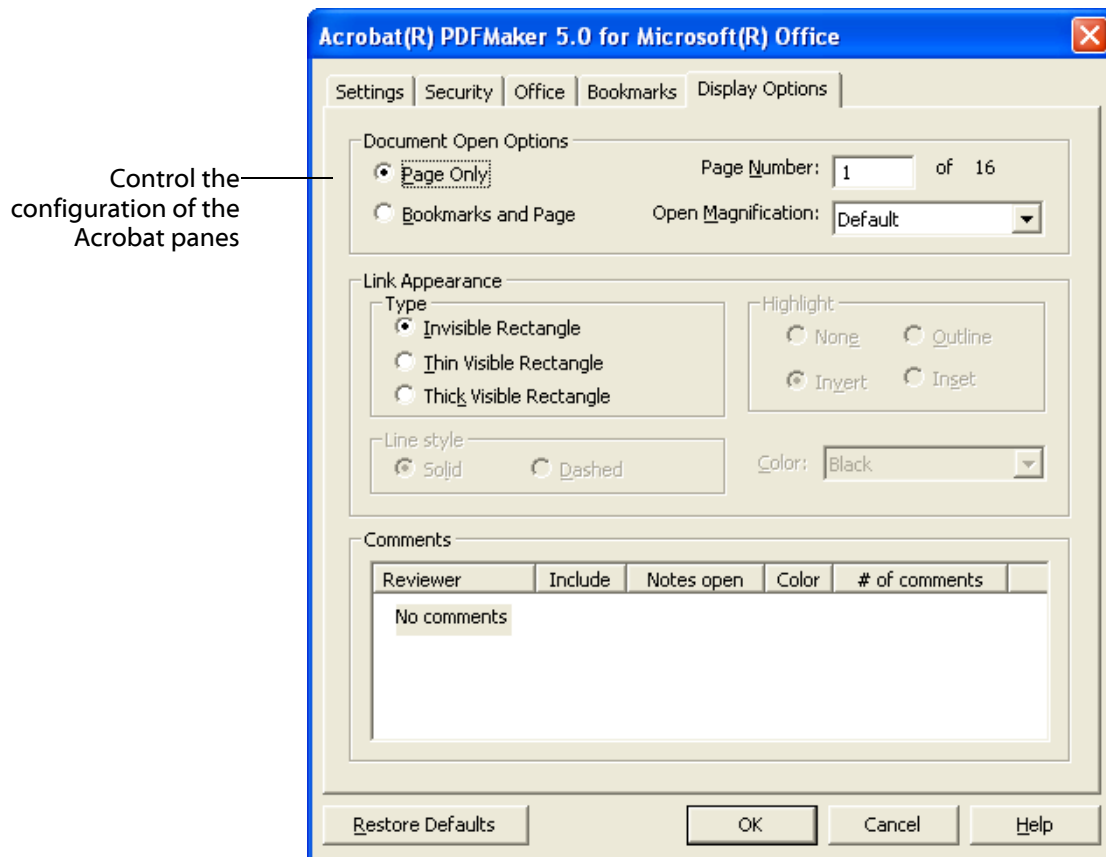
To configure bookmarks, use the Bookmarks tabbed dialog. Place a check mark in the bookmark toggle for each style you want converted.



Instructor Notes: Screen Reader users can use Bookmarks to jump to associated places in the document, avoiding the tedium of having to sequentially read through the entire document to get to a particular point. Select an appropriate level of detail of styles to use as Bookmarks to provide useful jump points into the document content for users with screen readers.

Defining Opening Views

You can also define how the Acrobat 5 work area appears when a user first opens a specific document. For each document the author can specify:



- The initial page view, that is, whether users see only the document pane, or whether they also see the Bookmarks, Thumbnails, and Comments Palettes.
- The opening page number for a multi-page document.
- The magnification level.
- The type of visual indicator for cross reference links, URLs, network addresses and e-mail addresses.
- If comments are present, which reviewers comments should be converted.



Exercise: Creating a Tagged PDF from Word

In order to do this exercise, you must have Word 2000 or higher and Acrobat 5 installed on your machine. You must also have downloaded and unzipped the file **AccessWord.zip** from the Accessibility training Web site.

In this exercise you will configure Tagged PDF generation from a Word 2000 or higher document and then use PDFMaker to create a Tagged PDF document.

At the end of the exercise you will be able to:

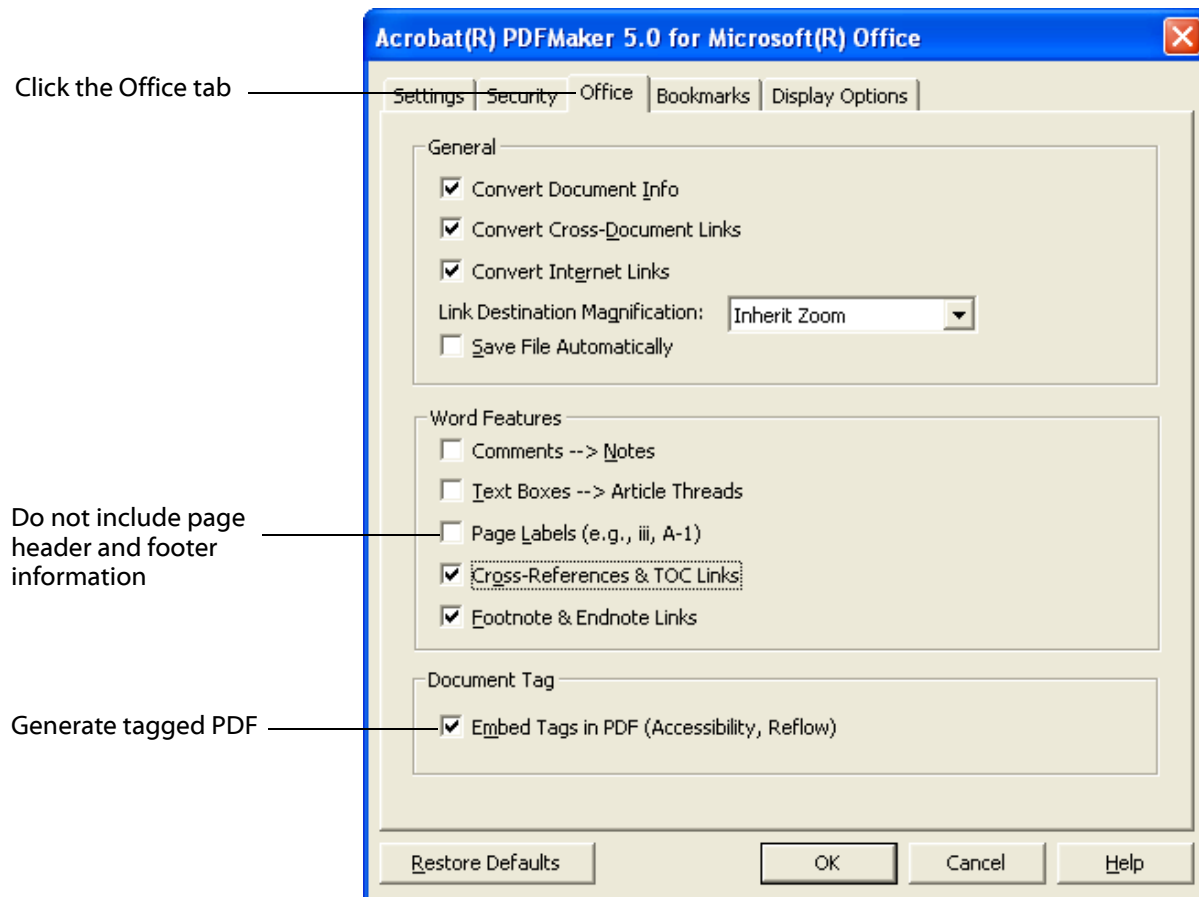
- Control whether you create a standard PDF or Tagged PDF document
- Specify which Word styles are converted into PDF tag elements
- Tailor the styles used to create bookmarks for a Tagged PDF file
- Specify initial display settings when Acrobat opens the Tagged PDF file
- Use the PDFMaker macro to create a Tagged PDF document

1. Open **UserGuideStart.doc** in Microsoft Word 2000 or higher.

This document is the Word 2000 document that you will be using for the remaining exercises in this module. It contains a short user guide for a small application used to record times for an after-school daycare program.

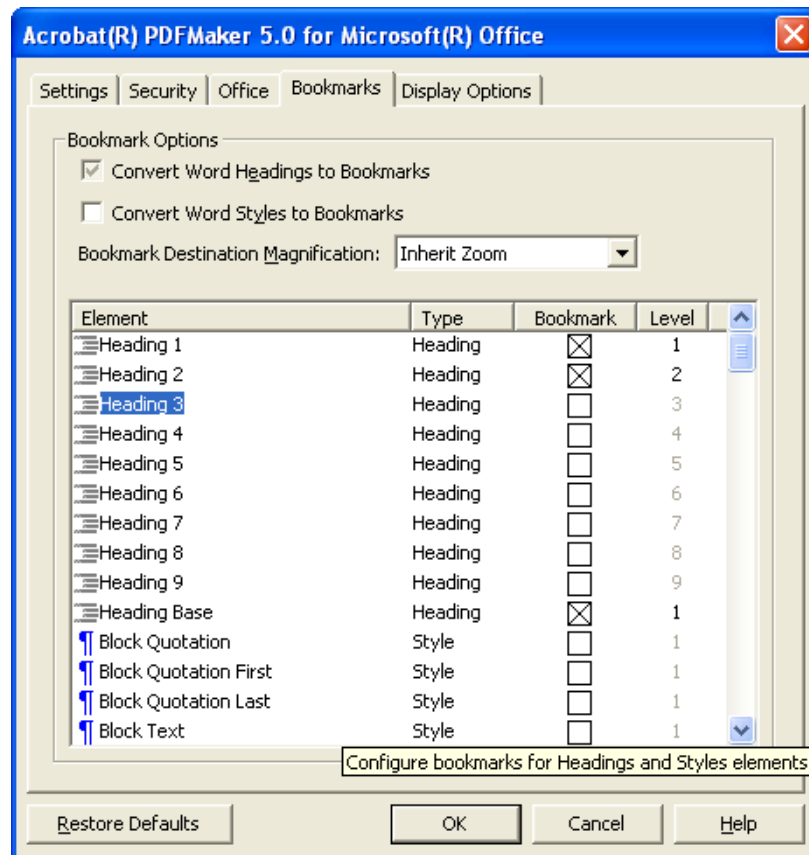
2. Select **Acrobat > Change Conversion Settings**. The **Acrobat PDFMaker** dialog appears.
3. To set up PDFMaker to generate tagged PDF, as opposed to standard PDF, click the **Office** tab. Options for converting the document to PDF appear.
 - a. Make sure that the **Page Labels** check box is not selected and the **Embed Tags in PDF** check box is selected.

Since this document does not include comments and also does not scatter document content in linked text boxes, there is no need to have PDFMaker convert these Word features.



4. Click the **Bookmarks** tab.
 - a. Make sure that the Heading 1, Heading 2, and Chapter Title styles are converted into bookmarks, by clicking on the corresponding check boxes.
 - b. Change the level for Heading 1 from 1 to 2.

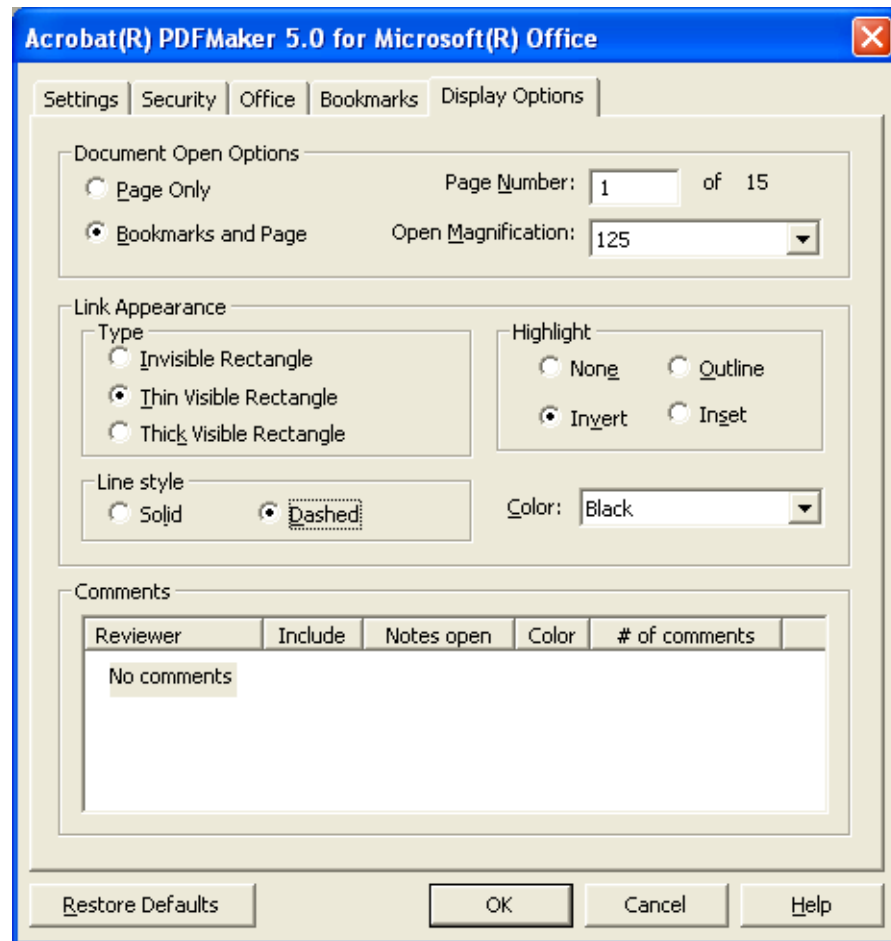
- c. Change the level for Heading 2 from 2 to 3. .




5. Click the **Display Options** tab.

You will configure Acrobat's opening view to show both the document and bookmark panes, a magnification of 125%, and a thin, dashed rectangle for links.

- a. Under **Document Open Options**, click the **Bookmarks and Page** radio button, select 125 in the **Open Magnification** combo box.
- b. Under **Link Appearance**, click the **Thin Visible Rectangle** radio button and under **Line style**, click the **Dashed** radio button.



6. Click **OK**.
7. Click on the Acrobat icon , the **Save PDF File As** dialog appears. Browse to the desired folder and enter **UserGuideStart.pdf** as the file name. Double check that the **Save as Type** is PDF. Click **Save**. Remember the location of the PDF file, you will need it for the next exercise.



Testing Your Tagged PDF

It normally takes some time to gain enough familiarity with the ins and outs of your authoring software, with respect to accessibility. While it might be tempting to take the first Tagged PDF file you produce and immediately post it to your Web site, we strongly recommend against this. Since accessibility is a new area, much of the software that is available has its own idiosyncrasies in terms of what it can and cannot do well in converting a file to an accessible format. PDFMaker is no different.

Once you have a Tagged PDF document, you can use Acrobat's tools to verify the logical read order of your document and to make sure that the page content is truly accessible. Your final test will be using a screen reader. Use the checklist below:

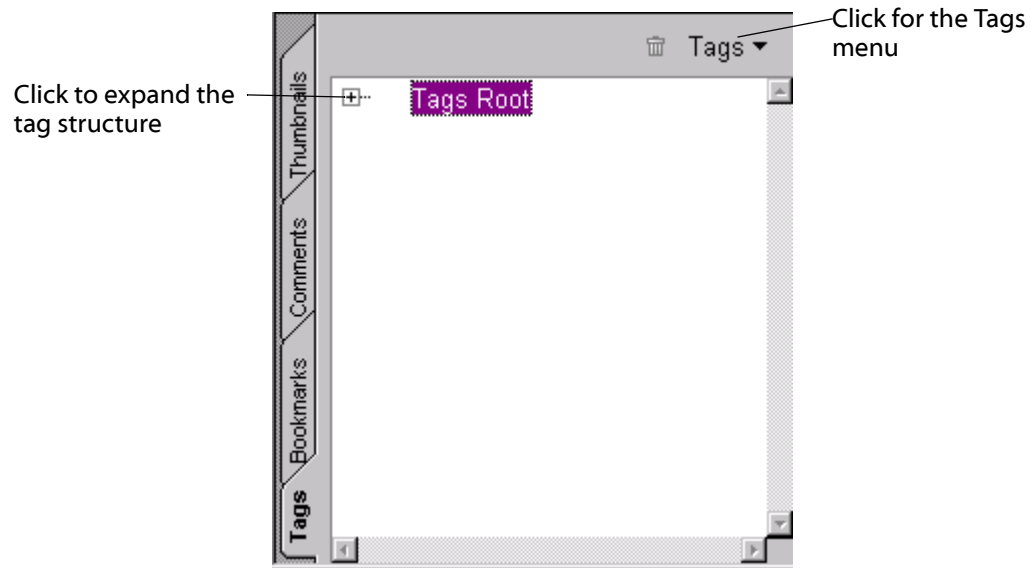
- Step through your document using the Tags Palette to view the logical read order
- Use the Accessibility Checker to flag potential problems
- Search for page content that is not included in the PDF tag structure
- Save your document as text to visually confirm the reading order
- Use a screen reader to check alternate text for graphics

[Instructor Notes: Point out here, and reinforce during the exercises, that thorough accessibility testing requires the use of multiple testing methods. Point out that screen readers and saving as text are useful methods for checking logical flow, but that they will not provide information about untagged elements in the document. The Accessibility Checker will do the latter, but is not useful for testing logical flow.](#)

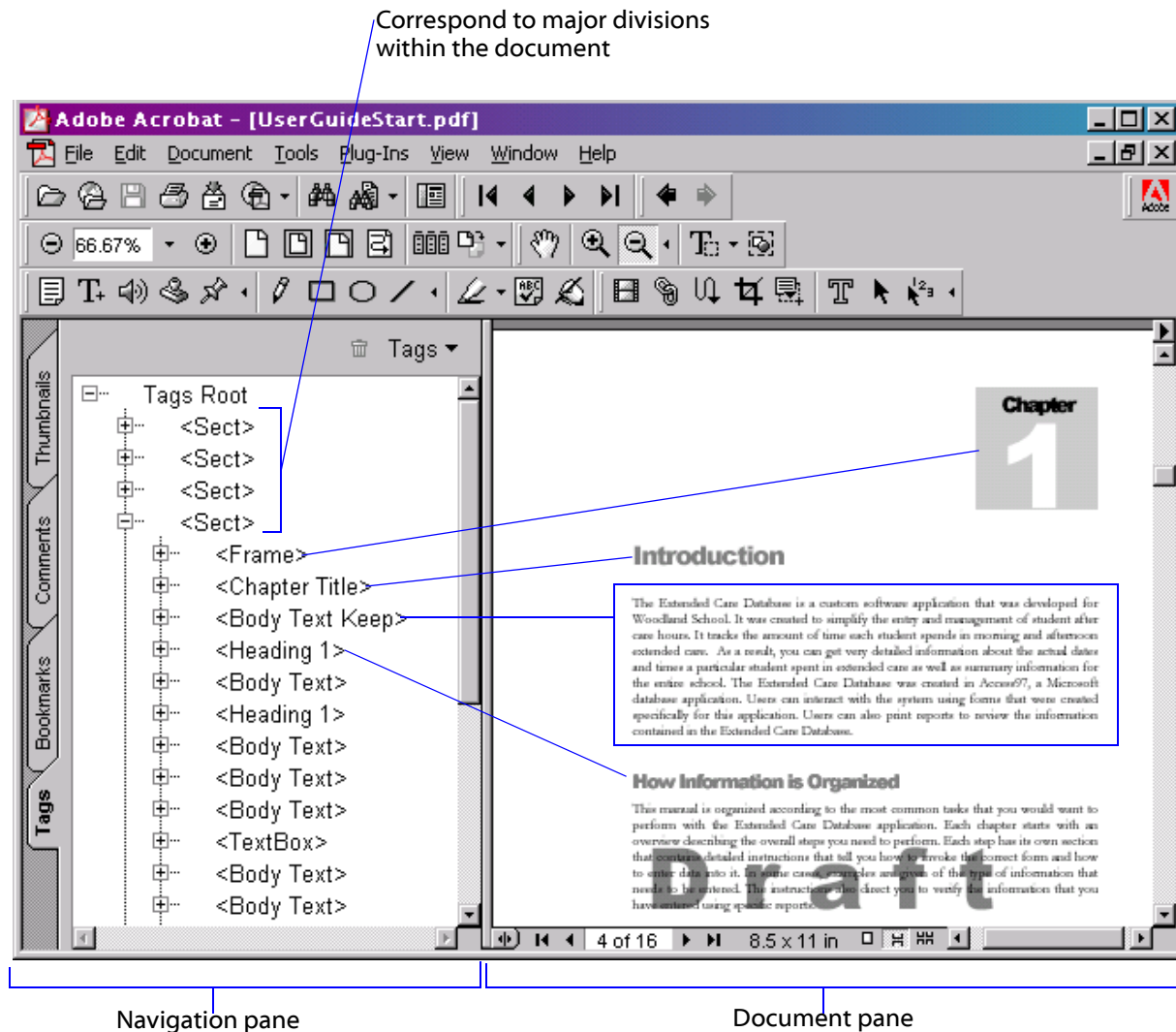
Acrobat Tags Palette

The Tags palette provides a means of viewing and manipulating a document's Tagged PDF. As mentioned in [Module 1: Introduction to Accessibility Issues](#), the order of the PDF tag elements reflects the logical reading order of the document. Tag elements are organized in a tree structure that reflects the document's structure. Major sections of a document, such as chapters, are intermediate nodes in the structure tree. These intermediate nodes can be further expanded to show the tag elements that comprise it. You can traverse the tree to get to individual subsections, tables, graphics, headings, paragraphs, and so on.

When you first open the Tags palette for a Tagged PDF document, you only see the root node of the collapsed structure tree. Click on the + sign to expand the tree. The upper right hand corner contains a menu that contains items you can use to find problems or modify the tag elements themselves or their placement. You will learn how to use the Tags palette to modify the tree, and therefore the reading order, in [Module 3: Using the Tags Palette](#).



The diagram below illustrates how you can map tag elements in the Tags palette to a document's components. Depending on the authoring software you used as well as the type of document, the actual tag names may differ from what you see here. Acrobat does not treat any one tag type differently from another.



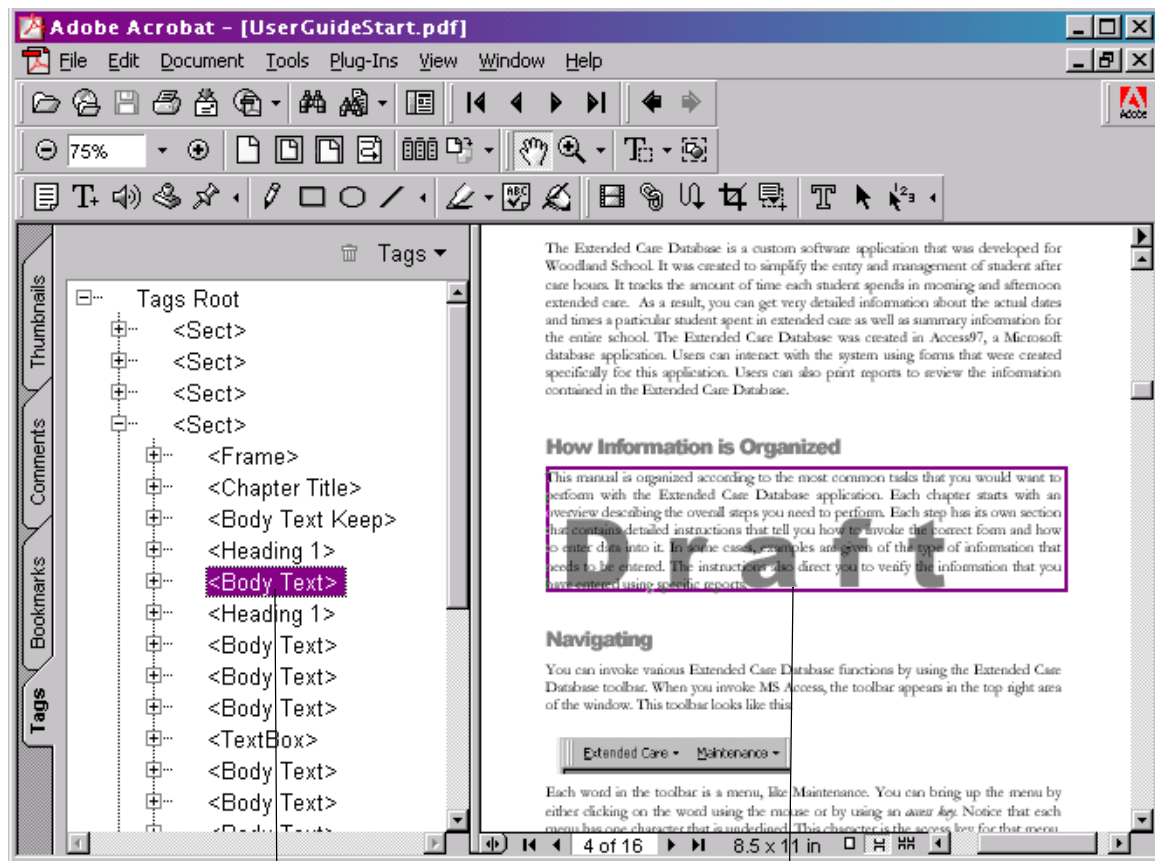
Verifying Read Order

The Tags palette enables you to map specific pieces of page content to their associated tag element. One benefit of this is that you can use this association to verify the reading order of your document by stepping through the elements in order.

To verify read order

1. Bring up the Tags palette. Expand the Tags Root.
2. Click a tag element to select it. Bring up the Tags palette menu by either right clicking on the element or by clicking **Tags** in the navigation pane. Click **Turn On Associated Content Highlighting**.

3. The document pane now highlights the page content that corresponds to the selected element.



Selected tag element

Associated page content



Turning on content association is a modal operation, in other words, you cannot execute any other Tags menu command while it is active. To turn off content association, use the Tags Menu and select **Turn Off Associated Content Highlighting**.

The Accessibility Checker

The Acrobat Accessibility Checker is a tool that analyzes a document and identifies potential problems that could impact accessibility. The Accessibility Checker does not look for problems with read order. While the tool can be helpful, you need to be the final judge on whether a problem warrants fixing. The Accessibility Checker can look for the following types of problems:

- No language specification

In the future, screen readers may be able to change languages from document to document. Specifying the language in which the document is

written, will provide the necessary information. Currently, assistive technologies do not utilize this feature.

- No alternate explanations for graphics

Visuals that convey important information for a document should have an alternate text explanation for visually-impaired people. There are some cases where you may choose not to provide alternate text because the graphic does not contribute any significant content, such as a company logo or border graphics for a page.

- Characters that do not have a corresponding Unicode encoding

Unicode is an international character standard that uses standard values for essentially all the major character sets in the world. Without a Unicode encoding, a screen reader will not be able to read the character.

Instructor: The problem we are solving here is how to figure out what character corresponds to each glyph on the page. Without the Unicode encoding, we may not be able to identify "normal" characters, either. This depends on the way the font information is encoded in the file. For instance, to save space, only the characters used may be included in the file, and they may be assigned arbitrary indexes in the compressed font. So the document may contain the word "cat", but all we will know is that it uses characters 1, 2, and 3 in the font.

Unicode is useful because it is a well-defined, standard way of identifying characters. The real accessibility benefit here is being able to extract the text reliably. It is something like this:

A = 121343

a = 134224

\$ = 352544

There are various fonts and character encodings that are non-standard so the letter "A" could be represented as a value = xyz for one font like Times and another character set for Chancery could represent the letter "A" = 123

The problem is that a screen reader must know what the character is to pronounce the letter "A". Without a standard for character encoding you would not be able to use screen readers.

- Document components, like text, graphics, and so on, that have no corresponding tag element

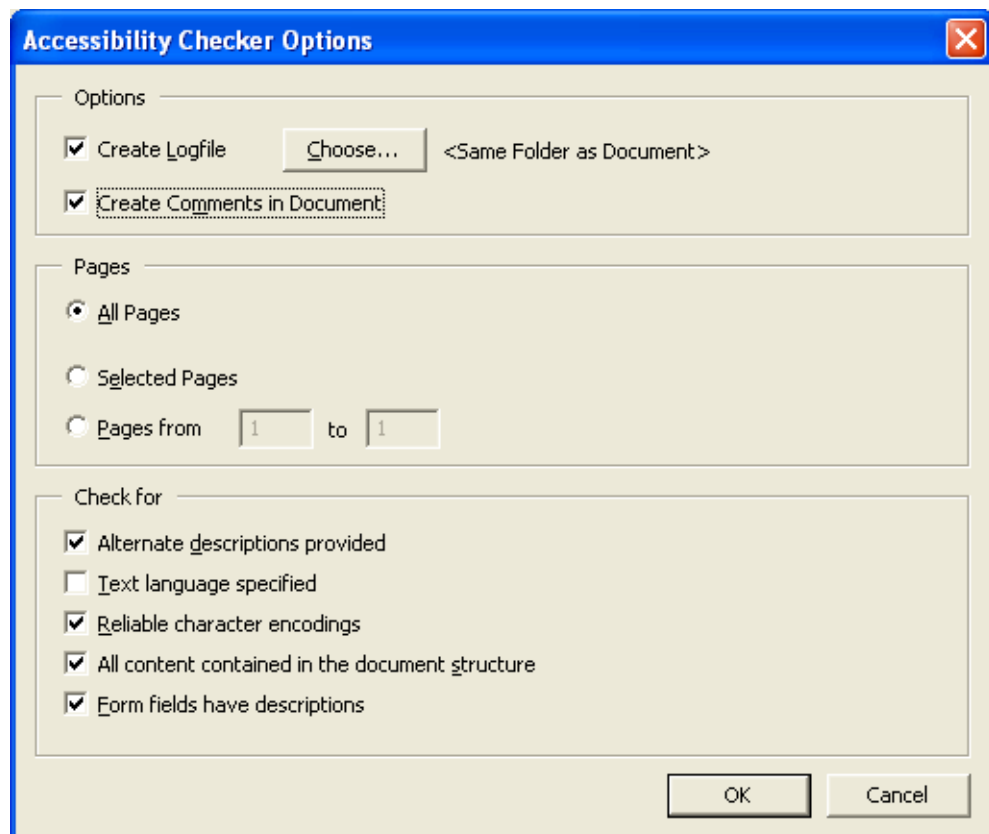
There may be occasions when the conversion process is unable to convert text, symbols, or visuals into a corresponding tag element. You should investigate these errors since any content without an associated tag element will be invisible to assistive technologies such as screen readers.

Some graphics are present to add color and visual appeal to a document. These document elements, which are referred to as *artifacts*, do not need

corresponding tag elements since they are not adding to the message of the document.

- Form fields that have no descriptions

If your document is an electronic form, visually-impaired users will need explanations for each of the data fields. Otherwise, it will not be clear what kind of information they have to provide. Acrobat displays the field's explanation as pop-up help, which is read by a screen reader. This course does not cover specific issues pertinent to making electronic forms accessible.



You can configure the types of problems for which the Accessibility Checker searches. By default, the Accessibility Checker reports a summary of its findings in a dialog box. While the summary reports the number and types of problems found, it is difficult to correlate those problems back to specific locations in the document. You should always enable comment creation, via the **Create Comments in Document** check box. This allows the Accessibility Checker to highlight the offending document element and attach a comment describing the type of problem it found. You can easily view the problems using the Comments palette. If you plan on fixing the accessibility problems using Acrobat's tools, make sure you delete the comments before saving the PDF. For more information on how to use the Comments palette, see *Adobe Acrobat Help* (online help included with Acrobat 5).

You can optionally specify that Acrobat write out its summary into a log file. You can also direct the Accessibility Checker to process particular pages, a range of page or the entire document.

Saving as Text

Instructor Notes: If students are using Acrobat 5.05 or greater running under Windows, have them Save As Text (Accessible), rather than Text (Plain). Text (Accessible) is only available with Acrobat 5.05 or greater running under Windows. You can save the PDF file as Text (Plain), but Alternate Text for graphics may or may not be displayed. Actual Text for graphics is displayed in both Text (Accessible) and Text (Plain) files.

In addition to using the Tags palette, you can also save the Tagged PDF file as text. This allows you to view the content in the same order that a screen reader would process them. You can quickly detect problems with reading order and missing content.

If text is missing from the text file, it is due to that text not having a corresponding PDF tag element. Depending on the authoring software that you use, you may be able to modify the list of formatting styles or tags used by the application and converted by PDFMaker into a tag element. If the authoring software does not provide a way to control formatting style conversions, you will need to create tag elements for the content that does not appear in the text file.

If you are running Acrobat 5.05 or greater under Windows, save the text file as a **Text (Accessible)** type rather than as a **Text (Plain)** type. The **Text (Accessible)** option is installed with the MSAA plug-in while the **Text (Plain)** option is installed with the Save As XML plug-in. **Text (Accessible)** more closely resembles what the screen reader sees. You can save the file as a **Text (Plain)** type, but Alternate Text for graphics may not be displayed in the file. **Text (Plain)** files will display Actual Text for graphics correctly.

Screen Readers

Instructor Notes: The use of a screen reader is almost essential for complete and accurate testing of document accessibility. We strongly recommend installing at least one for use by the class. Ideally, each student workstation would have a screen reader installation along with a headset to reduce noise. Alternatively, the instructor can have a single copy of the screen reader installed on a workstation and can demonstrate its use to the entire class at appropriate times. In the latter case, the instructor should call for students' attention toward the end of testing exercises, and use the screen reader to demonstrate the results of document modifications made during the exercise.

Screen readers can be difficult and confusing to use. We recommend that you spend time practicing with one before delivering the course. Learn the basic navigation commands needed to use a screen reader for the kinds of testing described in these exercises, such as start at the top, start over, read next line, read last line, jump to hyperlinks, and so on. Free demo downloads of popular screen readers, with documentation, are available from the URLs listed below.

Using a screen reader to read a Tagged PDF file is the benchmark of accessibility. Obviously if you don't hear the content you are expecting, something is wrong. While we do recommend that you use a screen reader to test your document, we also recommend that you save its use for last. Listening to a screen reader process a document is a time-consuming task. A screen reader's focus is the application that has the keyboard focus. Since each screen reader is different, please refer to that product's documentation to learn how to use it. Learning how to effectively use a screen reader takes training and time.

Screen readers are the best way to test any extra explanatory text that you have tied to a graphic. Acrobat preferentially searches for the alternate text for a tag element and passes that to the screen reader. If the alternate text is not present, Acrobat will read the actual text for the tag element and pass that to the screen reader.

Three major vendors of screen readers, GW Micro, Freedom Scientific, and Dolphin Oceanic, offer free downloadable demonstration versions of their products. You can use the demonstrations for 30 minutes, then you will need to reboot your machine to use the screen reader again. You can try them out at

- GW Micro — <http://www.gwmicro.com/demo/>
- Freedom Scientific — http://www.freedomscientific.com/fs_downloads/jaws.asp
- Dolphin Oceanic — <http://www.usa.com/download/demos.htm>



Exercise: Testing for Accessibility

In order to do this exercise, you must have Acrobat 5.05 and Microsoft Word 2000 or higher installed on your machine. You can optionally download a demonstration version of GW Micro's, Freedom Scientific's, or Dolphin Oceanic's screen reader.

In this exercise you will use a variety of different techniques to verify that the PDF file you created is accessible to a visually-impaired person.

At the end of the exercise you will be able to:

- Use the Acrobat Tags Palette to check a document's current read order
- Use the Acrobat Accessibility Checker
- Use a screen reader to verify the read order

Getting Started

The Tags Palette is a useful tool for examining the structure behind a tagged PDF document. Normally, the Tags Palette is a separate floating window. However, the default set-up hinders the work you will need to do with the document. Your first task is to dock the Tags Palette in Acrobat's navigation pane.

1. Invoke Acrobat. Open `UserGuideStart.pdf`. Select **Window > Tags**.

A dialog appears that contains two tabbed sections, one for **Fields** and the other for **Tags**. In order to view the Tags palette and the PDF document side by side, you need to dock the Tags palette in to the navigation pane. Use the steps that follow to accomplish this.

2. Place the mouse cursor over the **Tags** tab in the Tags palette.
3. Keep the left mouse button depressed as you drag the Tags palette to the navigation pane. Release the mouse button when you reach the navigation pane. You should now see an additional tab for Tags in the navigation pane.

Checking Read Order with the Tags Palette

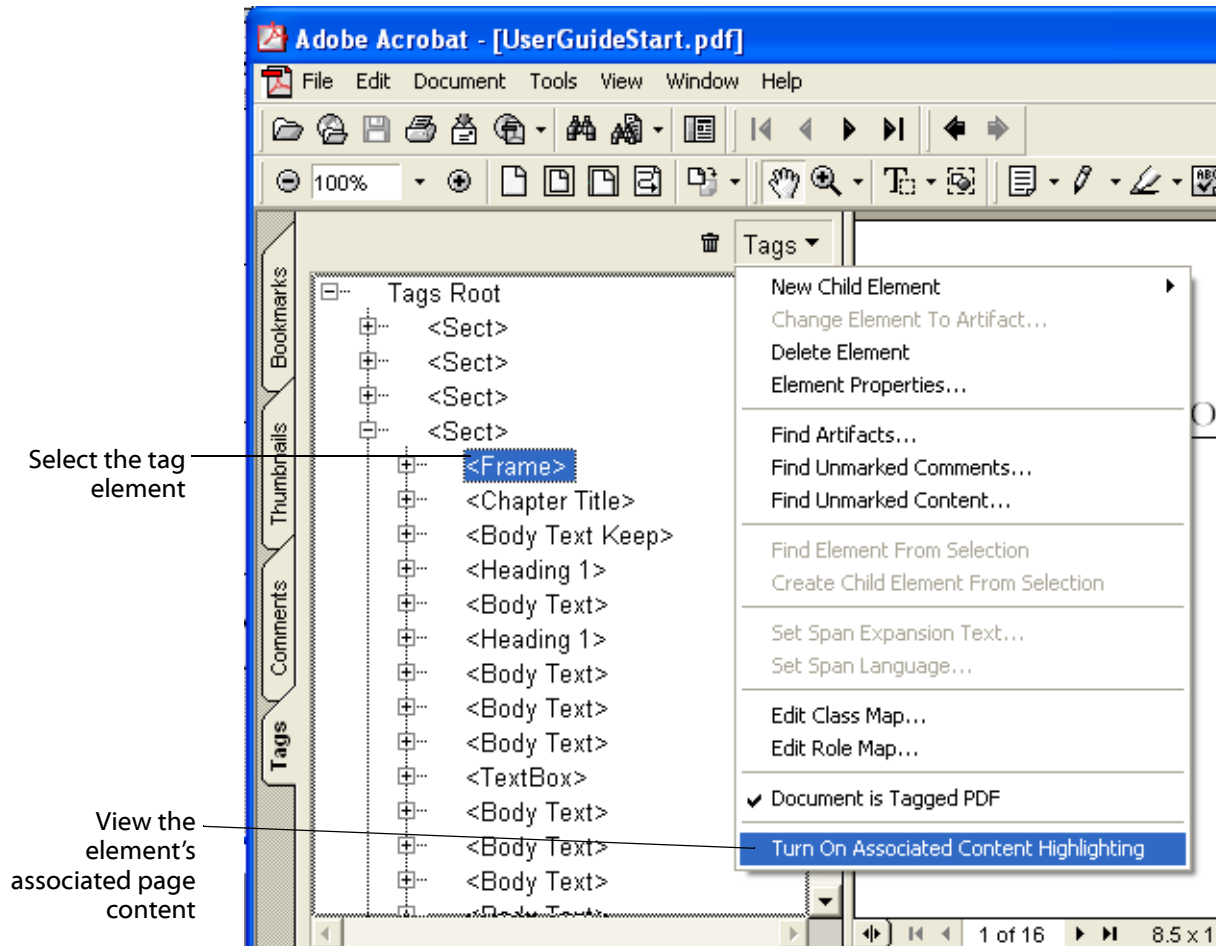
1. With `UserGuideStart.pdf` open in Acrobat, click on the **Tags** tab.

You will see the Tags root.

2. Expand the Tags root to view the seven sections in the document.

The first three sections correspond to the cover, title page and the table of contents. The last four sections correspond to the chapters.

3. Expand each of the last four sections.
4. Click on the first child element, `<Frame>`, under the first expanded section. Now click on the **Tags** menu. Select **Turn On Associated Content Highlighting**.



5. Acrobat automatically moves to the page containing that tag element and boxes the contents of the tag on the screen so you know which document components map to which tag elements.

Instructor Notes: Reinforce the idea that the sequence of the tag structure determines the logical read order of the document.

6. Use the down arrow key to move through each element in the order it would be processed by a screen reader.

As you move through the document, you should notice problems with the read order. Most of these are caused by using text boxes for some of the simple diagrams that are in the document. Make note of the problems that you find so you can fix them in Word or PowerPoint in [Correcting Problems in Word](#).

Take a moment to examine the table tags that are used in the first chapter (first expanded section). Drill down by expanding the structure tree under the **<Table>** tag. Tagged PDF uses elements similar to HTML to encapsulate the table headings and cells.

The last paragraph in the first chapter contains e-mail addresses. When you expand the **<Body Text>** tag, notice that PDFMaker has converted these into a **<Link>** tag. These are processed specially by screen readers.

In Chapter 3 (third expanded section), take a look at how the list elements are organized in the structure tree. Note that the list items in multi-level lists are broken out into individual list tag elements in the structure tree. PDFMaker has not captured the entire list structure the way a person would have. However, it actually does not have any negative impact since the logical reading order is preserved. Assistive technologies do not currently have access to the structural context of the tags. As a result, they do not tell you that you have encountered a list with six items.

7. Turn off the associated content highlighting when you are finished.

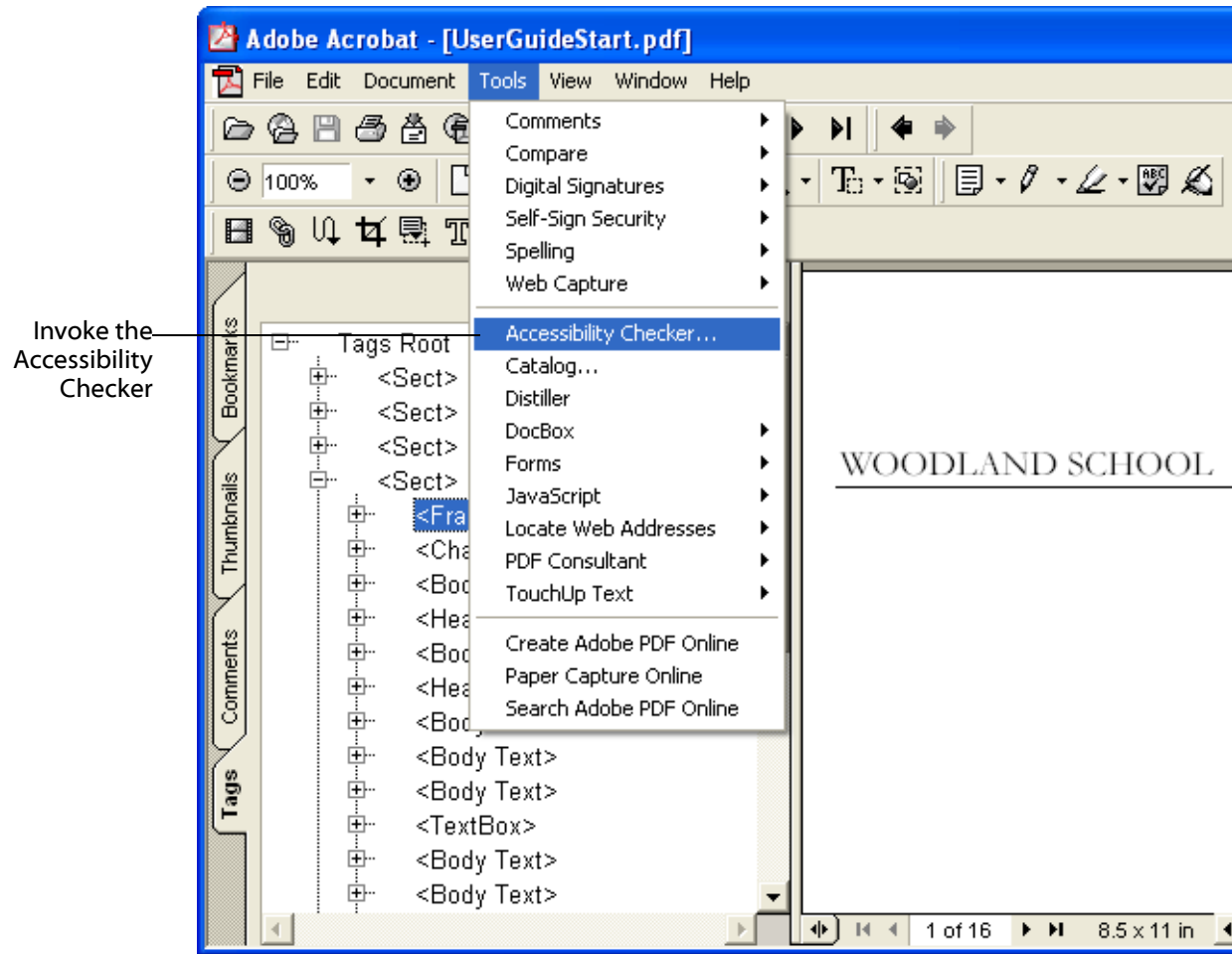
You can compare your findings with those listed in [Solutions](#). Be aware that the page number shown in Acrobat does not match the document page number since the document's page number does not include the cover, title or table of contents pages.

Using the Accessibility Checker

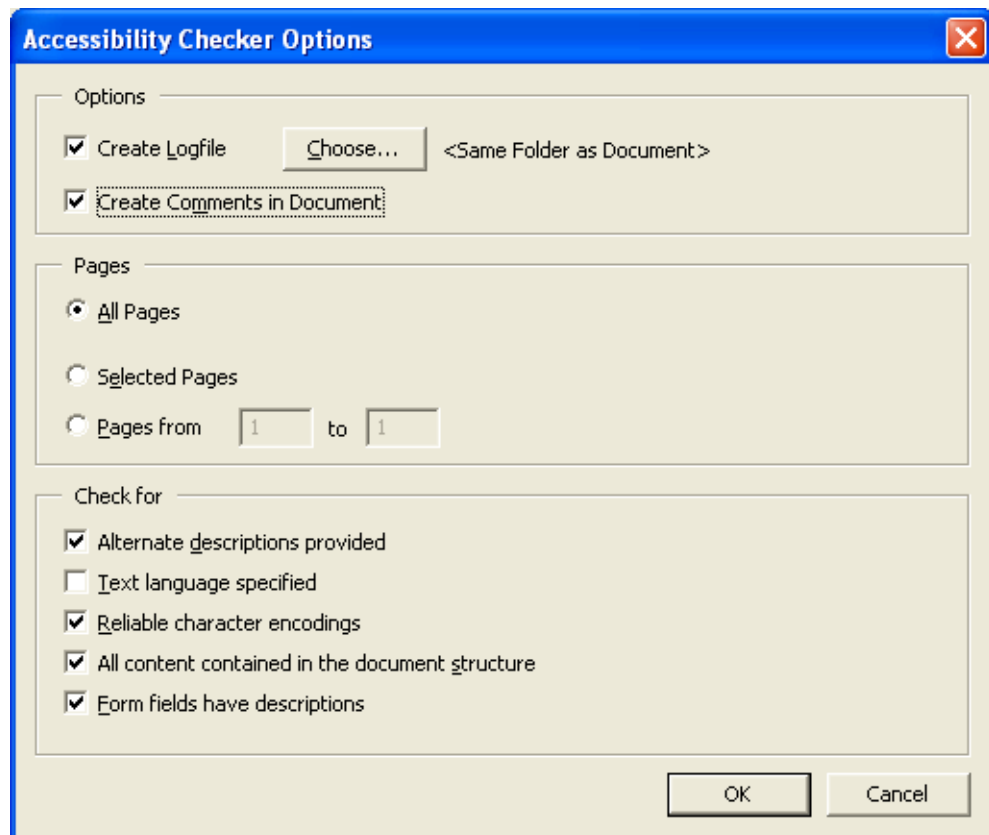
Acrobat provides a tool called the Accessibility Checker to check various aspects of accessibility for a document.

Instructor Notes: Remind students that the Accessibility Checker flags omissions and other problems with the PDF Tag structure, but does not check the tag sequence and logical read order of the document.

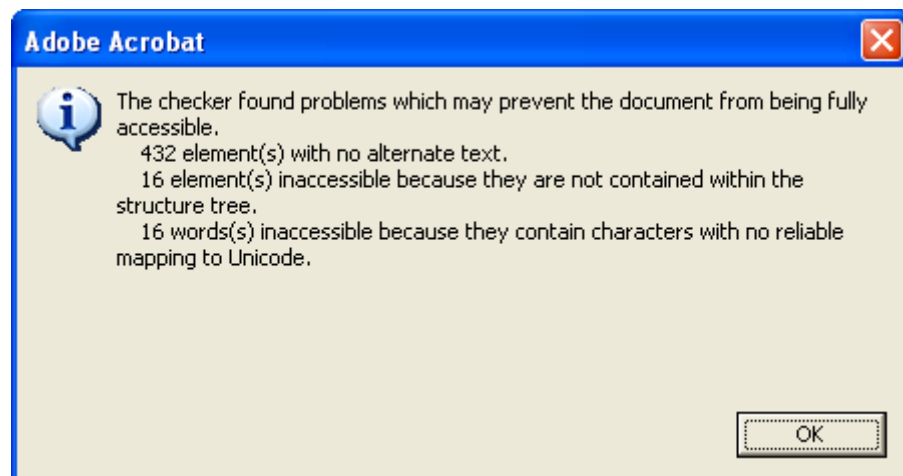
1. Select **Tools > Accessibility Checker**.



2. The Accessibility Checker Options dialog appears.
 - a. Check the **Create Logfile** check box, this creates a logfile of accessibility warnings that you can later reference.
 - b. Check the **Create Comments in Document** check box.
 - c. For now, we are turning off any language checking, so in the **Check for** section, the **Text Language Specified** check box should be unchecked.
 - d. The remaining check boxes should be checked.
 - e. Click **OK**.



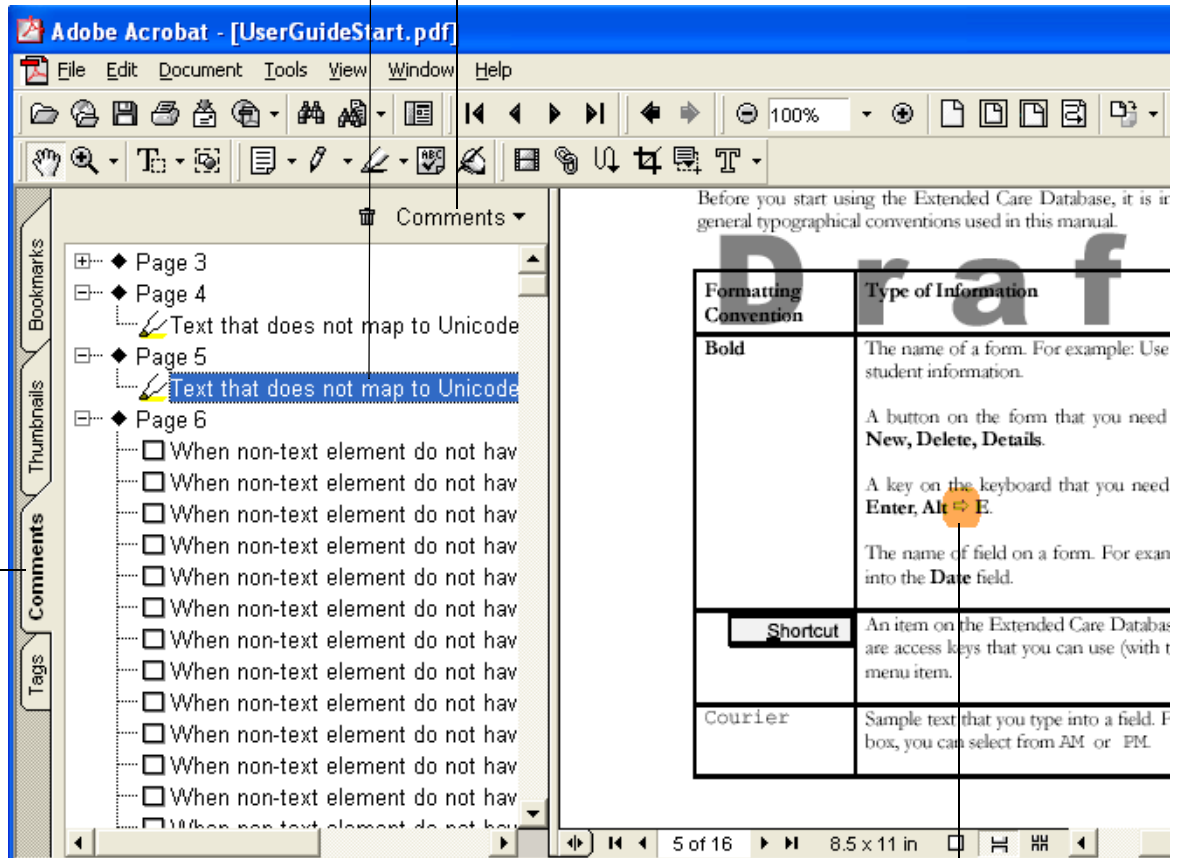
3. On initial perusal, the errors may not make much sense. Fortunately, since you have turned on the comment creation, you can page through the document and note where the Accessibility Checker has flagged problems in the document.



Click here to view problems
flagged by the Accessibility
Checker

Double click comment to view it

Click the Comments menu for other options



Acrobat highlights the page
content that is causing this
particular problem

4. In the document pane, you can view a comment by simply double clicking on it. In the navigation pane, you can also click on the **Comments** tab. The Comments menu allows you to sort the comments by page, author, and so on. Since Acrobat generated all the comments, it is generally more useful to sort these comments by page.
5. On analyzing the log entries, it becomes apparent that they fall into three main categories:
 - document content that is not part of the PDF tag elements
 - missing alternate text for diagrams
 - no unicode mapping for a particular character

Make note of the problems found, you will correct them in the next exercise.



Exercise: Correcting Problems in Word

In order to do this exercise, you must have Acrobat 5, Microsoft Word 2000 or higher and Microsoft PowerPoint 2000 or higher installed on your machine.

In this exercise you will apply the guidelines that you learned to correct the accessibility problems that you found in `UserGuideStart.pdf`. Your goal is to repair all the problems that you found in the previous exercise.

[Instructor Notes: Re-emphasize the goal of fixing the PDF tag structure in the original Word document, prior to the PDF conversion.](#)

At the end of the exercise you will be able to:

- Replace graphics constructed with text boxes
- Insert graphics using **Insert > Picture**
- Correct floating graphics by modifying their wrapping style to be inline
- Associate alternate text with graphics
- Substitute characters with Unicode encodings for those that do not have Unicode encodings

Now that you have finished testing `UserGuideStart.pdf` for accessibility problems, you need to edit the file in Word to fix the problems. In summary, you should have identified the following problems.

- Ordering problems with the graphics on pages 1, 2, 4, 6, 7, and 11.

Most of the errors that you found stepping through the document with the Tags palette are caused by using text boxes with the graphics or by not modifying a graphics properties so it is an inline image instead of a floating image. You will correct these problems in Word by replacing the culprit screen shots with inserted pictures or by modifying the properties.

- Missing alternate text

As long as you are working with graphics, you can also add alternate text for them.

- Text without a corresponding PDF tag element

Since the missing tags are all in the table of contents, we can work around this particular problem by using Acrobat bookmarks instead of Word's table of contents.

[Instructor: These particular problems are due to a bug in PDFMaker that is not correctly handling table of content cross references.](#)

- Characters without a Unicode counterpart
1. Make a copy of `UserGuideStart.doc` so you can refer to it for the graphics and their placement on a page.

2. Open **UserGuideStart.doc** in Word. Fix the problems caused by the text boxes. Delete all the text boxes.

All the screen shots used in **UserGuideStart.pdf** are contained in **graphics.ppt**, a PowerPoint 2000 file. You can select an appropriate technique to reinsert the graphics.

- Use a different application to create a **.jpeg**, **.tif** or other supported graphic format by copying and pasting the graphics from **graphics.ppt**. Use **Insert > Picture** to get the image into the Word document.
 - Copy the graphic from **graphics.ppt**. Use **Paste > Special** to paste the image. Right click on the graphic to invoke a popup menu. Select **Format Picture**. Bring up the format picture dialog and modify the text wrapping style to be inline.
3. Construct an alternate flowchart in a drawing application of your choice to replace the one created from text boxes on page 4. Insert the new flowchart into the document.
 4. Add alternate text to each graphic.
 - a. Right click on the graphic to invoke the popup menu and select **Format Picture**.
 - b. When the Format dialog appears, click on the **Web** tab and enter the explanation.
 - c. Save your changes.
 5. Delete the table of contents to eliminate the error that elements are missing from the structure tree. You have already used bookmarks to represent the table of contents.
 6. Delete the Draft watermark.
 - a. Select **View > Headers and Footers**.
 - b. Click on the text box that contains the word "Draft." Delete the text box. You may need to do this on both odd and even pages.
 - c. Click **Close** in the Header and Footer dialog to return to the normal view of the document.
 7. Do a global search and replace to substitute ">" for the arrow symbol that is causing the Unicode errors.
 8. Save your changes. Generate another Tagged PDF file and use the testing techniques to evaluate the document's accessibility.



Solutions

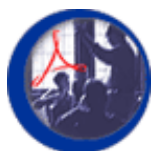
This section contains solutions for parts of the exercises from this module.

Problems in UserGuideStart.pdf

The following is a list of known read order problems and other issues that you can find by using the associative content mapping in the Acrobat Tags palette. The page numbering is that used in the Word document, to map it to the page numbers shown in Acrobat, add 3.

1. Page 1, the screen shot for the Extended Care menu is placed last behind all other page elements. This is due to placing the screen shot in a text box.
2. Page 2, the screen shot and call out for the combo box are placed last relative to the other page elements. This is due to using a text box.
3. Page 4, the flowchart elements are not in proper order. This is due to using text boxes for the flowchart boxes.
4. Page 6, the screen shot is placed last. Blame those pesky text boxes.
5. Page 7, we cannot even see the screen shot but it is there and out of order again.
6. Page 9, if you drill down into the first list item, you will notice that the text from the watermark has been included in the text for the list item. This will be read by a screen reader.
7. Page 11, yet another screen shot that is out of order.

Module 3: Using the Tags Palette



Introduction

This module introduces various kinds of accessibility problems that you may face in your Tagged PDF. While we highly recommend that you fix as many problems as possible in the authoring software, there may be cases where this is not possible. We encourage you to keep the number of repairs that you need to make using the Acrobat Tags palette to a minimum to avoid having to redo them with each new revision of a document.

Instructor Notes: Modifications made directly to the PDF Tag structure in Acrobat cannot be saved. Each time the PDF document is updated and re-generated from the original authoring tool, for example, InDesign, FrameMaker, or MS Word, manual changes made directly to the PDF document must be reapplied. Acrobat 5 provides a great deal of flexibility in regard to modifying the PDF Tag structure, but make sure students understand the advantages of avoiding post-PDF conversion changes to the Tags structure when possible.

This module is organized as a series of exercises to allow you to get the hands-on practice that you need with the Tags palette.

Learning Objectives

At the end of this module, you will be able to:

- Understand the difference between logical read order versus page content order
- Rearrange PDF tag elements to fix logical read order
- Modify the reflow of a document to fix page content order
- Create new tag elements to add explanations or fix problems with links or text in existing tags
- Use Tag palette commands to find untagged content and dynamically create tag elements

References

- *How to Create Accessible Documents*
- *Advanced Techniques for Creating Accessible Adobe PDF Files*
 - <http://www.adobe.com/products/acrobat/pdfs/CreateAccessibleAdvanced.pdf>
- *Adobe® Acrobat® Help (online document included with Acrobat 5)*

Contents

Topics	Exercises
Logical Read Order and Page Content Order	Fixing Logical Read Order Problems
	Fixing Text Reflow Problems

Topics	Exercises
Fixing Content Problems	Creating New Tag Elements
Converting Missing Content	Converting Missing Content to Tags

File to Download for Exercises in this Module: TagsPalette.zip

TagsPalette.zip contains the following files:

- sample_text_reflow.pdf
- UserGuideStart.pdf
- UserGuideSolution.pdf
- MktPlan.pdf
- MktPlanSolution.pdf



Logical Read Order and Page Content Order

Tagged PDF is created when a source file is converted, exported, or Saved As PDF. The specific method for generating the PDF file depends on the authoring tool used for the source document.

You will work with Tagged PDF documents in the exercises that follow. First, however, you should be aware of two related but distinct characteristics of accessible documents, *logical flow*, and *text reflow*:

- We use the terms *logical flow* and *logical structure*, and occasionally *logical read order*, interchangeably.

The physical manifestation of *logical flow* is the order in which a screen reader speaks the content elements of the document. The order of content elements in a document's logical flow does not necessarily have to be the same as the visual flow or placement of elements in the document.

- The hierarchical order of the Tagged PDF structure determines the logical flow of the document. Consequently, we use the term *Tagged PDF structure* to mean almost the same thing as logical flow or logical structure.
- We use the term *text reflow* or *reflow* to refer to the order in which content elements of the document are wrapped when you enable Reflow in Acrobat. Once Reflow is enabled, Acrobat wraps and fits the content of the document to the document window when it is reformatted to different proportions, or when the display is magnified.

Reflow allows a PDF document to fit the reading area whether that be the user's desktop or handheld PDA. The Reflow view is active only when the PDF document is Tagged PDF. Tables and Figures do not reflow, but rather scale to fit the reading area. The order of elements in a reflowed document does not necessarily have to be the same as the visual flow of elements in the original document

- Text reflow is conceptually and physically separate from logical flow and tagged structure. The two do not affect each other; you must handle each separately.
- Text reflow order is important in regard to providing accessibility to people who are visually impaired but not functionally blind. These users can use the text magnification and reflow capabilities that environments like Acrobat 5.0 provide. Reflow capabilities are important for accessibility because text magnification by itself obscures visual context and forces users to scroll widely about the document in order to follow its logical flow. With reflow capabilities, magnified text of any size is reflowed properly into the available space.



The Acrobat Reflow feature works reliably for tagged Adobe PDF documents that contain Roman language text. It does not work reliably for tagged Adobe PDF documents that contain Asian language text.



Reflowing a document is for on-screen viewing only. You cannot save or print the reflowed state of a document.

The table below compares and contrasts logical read order and page content order. Note that if you want to insure that your PDF document is accessible to the blind via screen readers, you must verify the document's logical read order. If you want readers who are visually-impaired, but not blind, to be able to view a PDF document using a screen magnifier, you must verify the page content order. In general, you should do both.

	Logical Read Order	Page Content Order
Used by screen readers	✓	—
Used for reflow	—	✓
Used for repurposing (saving to other formats)	✓	—
How to view	Tags palette	Touchup order tool or select reflow tool
How to edit	Reorder tag elements in the Tags palette	Select desired text with the touchup order tool
How to verify	Select File > Save As , choose Text (Accessible)	Select View > Reflow

Fixing Logical Read Order

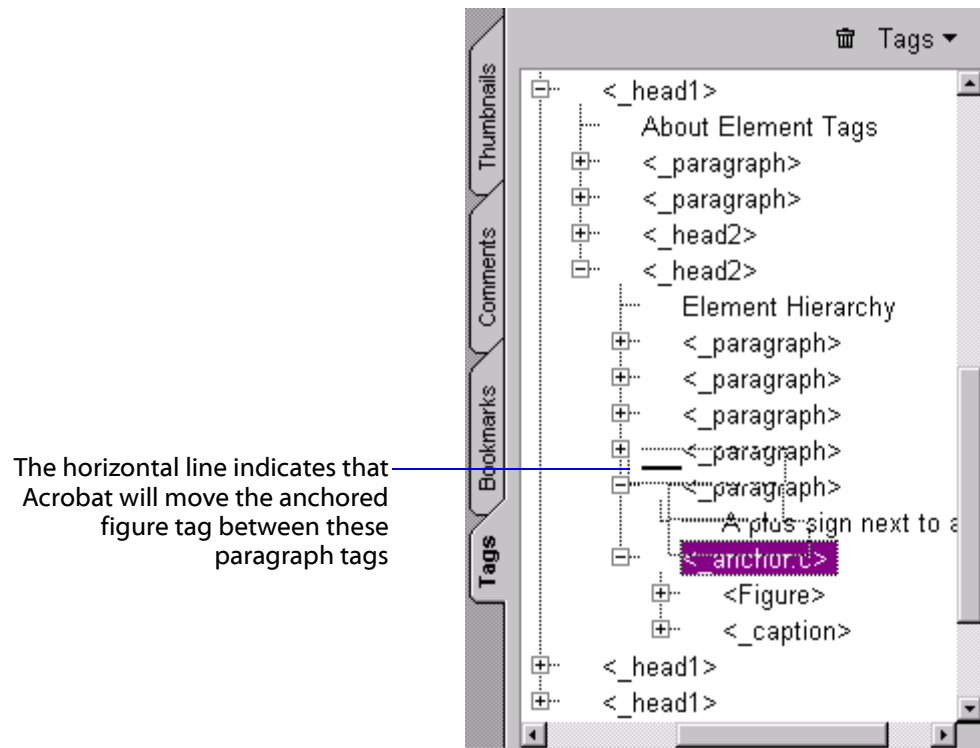
Instructor Notes: The following paragraphs and screen shots are an overview and introduction to using the PDF Tags Palette to manipulate the PDF Tag structure. Students will have a chance to do this in the next exercise. If you have access to a screen projector or display that can be seen by the entire group, you may want to demonstrate the basic principles to the class as they are introduced below.

In the first exercise in this module, you will use the PDF Tags Palette to correct some problems with the logical read order of a document. The following is an overview of the basic procedures:

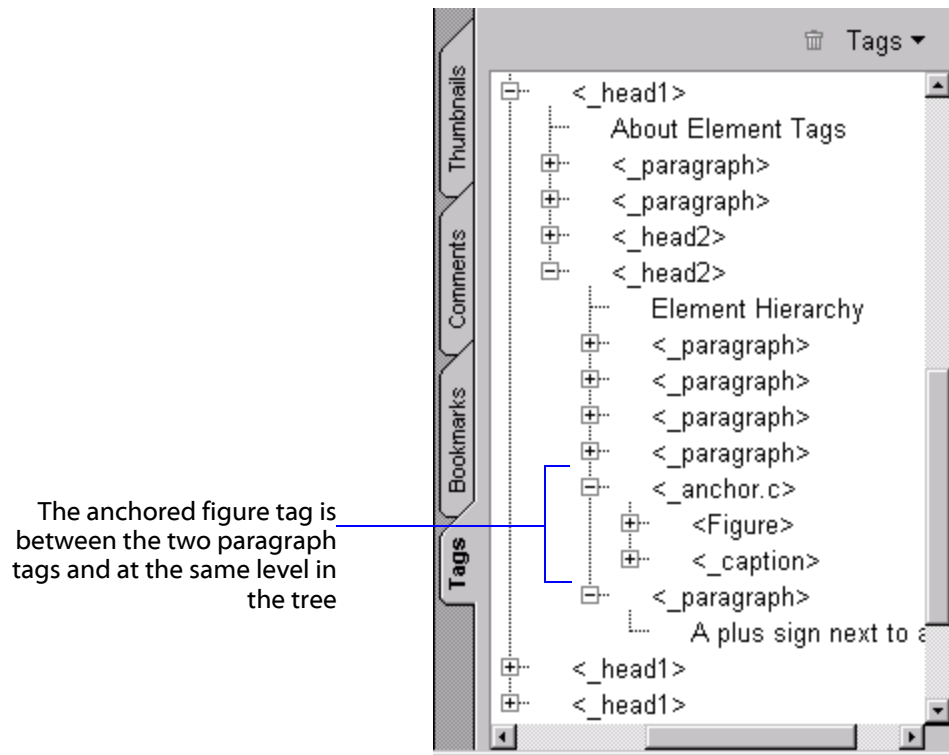
When you open a PDF document in Acrobat 5, a *Tags* tab is displayed in the Palette Window at the left. If the loaded document is a Structured PDF file, clicking on the Tags tab opens the Tags Palette and displays the document's PDF Tag structure. At first, you may see only one element, the *Tags Root*, which can be expanded to reveal the entire PDF Tag hierarchy. You can also reorder the Tag elements in the Tags Palette, which in turn affects the logical flow of the document without affecting its physical appearance.

The interface works in much the same way as moving files in the Windows Explorer. Using the left mouse button, you can select a tag element with the

mouse. You can move selected tag to different locations within the structure tree by keeping the left mouse button depressed. Once you reach the desired location, release the mouse button. You can also select more than one tag, provided that they are sibling tags in the structure tree. By using the Shift key and the left mouse button, you can select contiguous tags. You can use the Ctrl key and the left mouse button to select non-contiguous tags. When you move a tag or group of tags, the insertion mark is shown as a horizontal line.

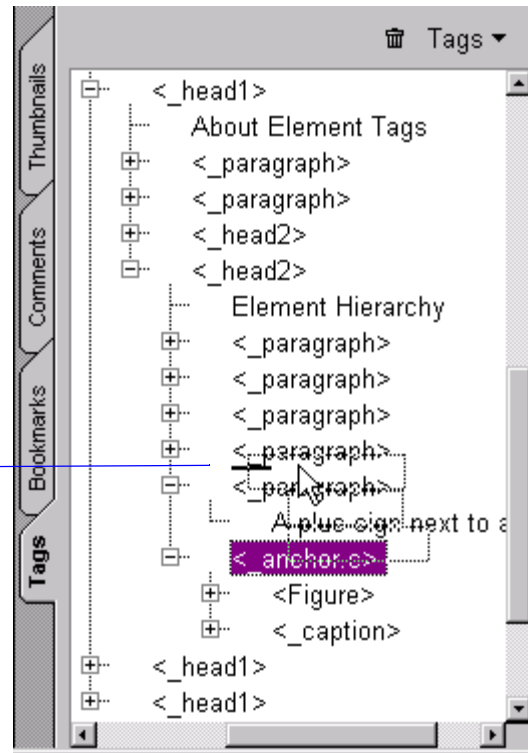


When you release the left mouse button, in the diagram above, Acrobat inserts the **<anchor.c>** tag and its child tag elements between the two paragraph tags. The resulting structure tree is shown below.



You need to pay attention to the horizontal position of the insertion mark. It indicates the level within the structure tree where Acrobat will insert the tags. In the diagram below, the insertion mark is even with the paragraph tag above it. This indicates that Acrobat will insert the selected tag as a child element of the paragraph tag instead of a sibling element. The difference is subtle but critical. Compare the two insertion examples and the results. Then try some experiments of your own.

The horizontal position of the insertion mark indicates that Acrobat will insert it as a child under the paragraph tag

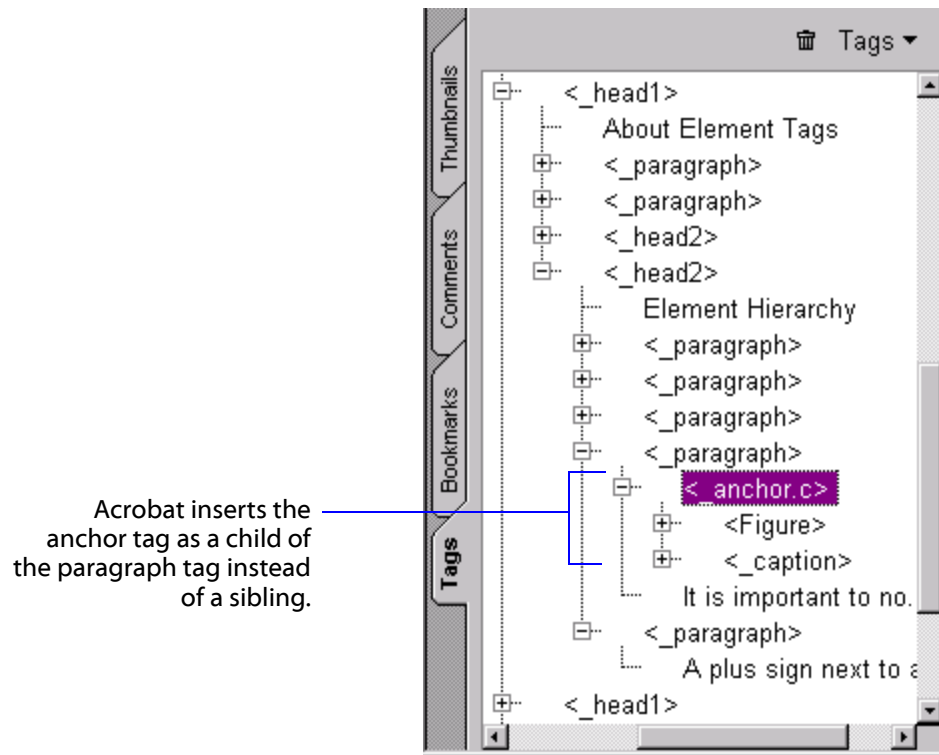


The diagram below shows the result. Acrobat has inserted the **<anchor.c>** tag as a child of the **<paragraph>** tag.

Instructor Notes: When Turn On Associated Content Highlighting is active, no other commands from the Tags menu can be selected.




If you see a circle with a slash through it instead of the insertion mark, Acrobat cannot move the selected tag elements to that location. If you see this symbol even in areas of the tree where the tag can legitimately belong, you are in content association mode. To check if you are in content association mode, right-click the mouse in the Tags palette to bring up the pop-up menu. If you only see one menu item, **Turn Off Associated Content Highlighting**, click it to exit content association mode.



Finding the Associated Element for Document Content

[Instructor Note: Demonstrate the following procedure to the class if possible.](#)

In the second module in this course, you learned how to step through the structure tree and view the associated document content for each element. There may be times when you are looking at the document and want to find the associated tag element for a particular paragraph, diagram, and so on. Rather than laboriously stepping through the structure tree, you can reverse the process by first

highlighting the desired page content, with the Text Select tool . Then, open the Tags drop down menu at the top of the Palette Window, and select **Find Element From Selection**. Acrobat expands the structure tree as appropriate and then highlights the associated element.



Exercise: Fixing Logical Read Order Problems

In order to do this exercise, you must have Acrobat 5.05 installed on your machine. You must also have downloaded and unzipped the file **TagsPalette.zip** from the Accessibility training Web site.

In this exercise you will:

- Use the Tags palette to find a paragraph's associated element in the structure tree
- Use content association to examine the logical read order
- Rearrange tag elements in the structure tree to fix the logical read order

At the end of the exercise you will be able to:

- Find the associated tag element for any document content
- Modify reading order by moving tag elements
- Understand that the insertion mark indicates the level of the structure tree
- Verify the revised reading order

You will work with and modify the sample file **UserGuideStart.pdf**. You can find an example of the correctly ordered document in **UserGuideSolution.pdf**.

1. Make a copy of **UserGuideStart.pdf** and open it in Acrobat.

This document, which is a short user guide, has problems with logical ordering. You will fix the logical read order by rearranging the Tag elements in the PDF Tags Palette. Since the goal is to understand how the PDF Tag Palette works, ignore for a moment that it would be preferable to fix the ordering problem by editing the original source document.

2. **Open** the Tags palette.
3. Go to page 4 of the document. Page 4 is the first page of Chapter 2.

You will see a several text boxes that were used to create a flowchart. A quick way to jump into the structure tree is to click on the Text Select tool and use the mouse to select the text in the document pane. Use the procedure below.

- a. Use the **Text Select** tool to select a portion of the text in the first text box. For example, highlight the text "Get the current list of students...".
- b. Click on the drop down **Tags** menu, located at the upper left of the Palette Window. Select **Find Element From Selection**. This opens up the structure tree and positions the cursor on the right element.
- c. Use the **Text Select** tool to select a portion of the text in the last text box. For example, highlight the text "Back up last year's data...".
- d. **Click** on the drop down **Tags** menu, located at the upper left of the Palette Window. Select **Find Element From Selection**. Notice that the last text box is located before the first text box in the PDF Tag hierarchy. This

means that the text in the last text box will be read by a screen reader before the text in the first text box, and as such will definitely not convey the intended meaning of the chart.

4. **Click** on the **Tags** drop down menu, or left-click on one of the elements in the Tags Palette, and select **Turn On Associated Content Highlighting**. Click on each of the **<TextBox>** elements in the Tags Palette that are between the last and the first text boxes you selected in Step 3, above. Notice that the text boxes are placed in the PDF Tag hierarchy in reverse order of what should be their logical order.
5. Use the following procedures to rearrange the text box elements in the structure tree so they are in the correct logical order:

Instructor Notes: Remind students that they will not be able to move Tags Palette elements while Associated Content Highlighting is On.

First, note the positions in the Tags hierarchy of the first and last of the five **<TextBox>** elements in the chart that you are going to rearrange.

Open the **Tags** drop down menu and select **Turn Off Associated Content Highlighting**.

If you have expanded the tags for the first and last text boxes, you may find it easier to collapse them again before moving their positions. Click the parent tag's minus sign to do so.

Click on the last **<TextBox>** Tag in the group of five to select it.

Instructor Notes: There is an extraneous, non-printing <TextBox> element toward the top of the chart's border. If anyone asks, it is a tag for an empty text box whose origins are unknown. It will not be read by a screen reader and can be ignored for our purposes here. However, students may confuse it in the Tags Palette with the first of the five actual text box tags, and you may need to point this out to them.

Holding down the **left mouse button**, move the last **<TextBox>** Tag upwards in the hierarchy to a place just above the first **<TextBox>** Tag in the group of five. Move the cursor to the left as far as you can without causing the cursor icon to change from a horizontal bar to a circle with a line through it. **Release** the left mouse button to place the Tag in the new position.

Note that if you do not place the Tag as far to the left as you can, it will end up as a child element of the Tag above, which is not what you want. If that happens, grab it again and move it to the right place. The tricky thing about moving tag elements is to make sure they are inserted in the right level in the structure tree. It is very easy to wind up with elements as child elements instead of sibling elements. Pay attention to the horizontal position of the insertion mark. You may have to experiment a few times to get it right.

6. If the tags structure gets hopelessly messed up, close the document, without saving it, reopen it and start again. Also make sure the text box elements are in the same order as in the document pane.
7. **Save** your changes.

8. Test your changes by saving the file as **Text (Accessible)**.
9. Open the text file with an editor and search for a portion of text from one of the text boxes. Check to see that the sequence of text from all of the text boxes flows logically as it should.



Exercise: Fixing Text Reflow Problems

In order to do this exercise, you must have Acrobat 5.05 installed on your machine. You must also have downloaded and unzipped the file **TagsPalette.zip** from the Accessibility training Web site.

In this exercise, you will:

- Reflow a sample PDF document and note various problems with reflow order.
- Use the touchup order tool to fix reflow problems.
- Test the results.

At the end of the exercise you will understand how to control the reflow order of a document, and will understand how text reflow differs from logical order.

Remember, you need to find and fix reflow problems if you want your PDF document to be comprehensible to people who need to use screen magnifiers.

Instructor Notes: Be sure to point out to students who are using InDesign as their authoring tool that text reflow in Acrobat can be specified and controlled in the original InDesign document. This is preferable to fixing the text reflow as a post processing step in Acrobat for the same reason that it is preferable to fix logical flow in the original authoring tool, namely that changes made to text reflow order in Acrobat will be overwritten if the PDF document is re-generated from the original source. The last exercise in the InDesign module covers controlling text reflow in InDesign. It is a supplemental module, so not all InDesign students may have had time to go through it.

Examining Reflow Problems

1. Open the file **sample_text_reflow.pdf** using Acrobat 5.

This file was created by exporting an InDesign document to PDF format. Note the normal sequence in which the elements of the document are read:

- title
- left-margin hanging text and paragraph heading, or vice versa
- column one of paragraph one
- column two of paragraph one, and so on.
- Note that the main graphic is located toward the bottom of the page.

2. Find the reflow tool  in the toolbar and click it.

The contents of the page reflow sequentially to fit the window. The document contents are reformatted to fit the window after the user changes the window size or font magnification.

Notice a number of discrepancies in the way the text is sequenced when reflowed as compared to the original, as shown in the following figure. To name a few:

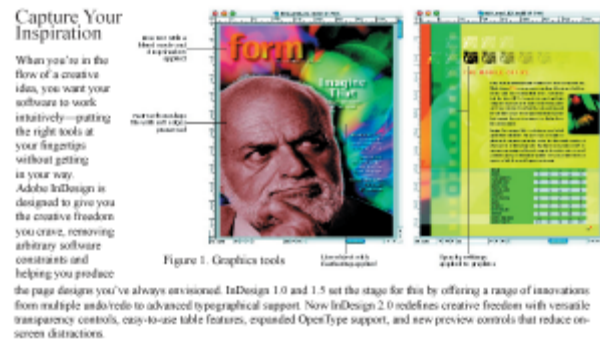
- The heading for the first paragraph is the lead item

- The multi-column paragraph on page one is split by the graphic from the bottom area of the page, the heading from the second section, and the hanging text in the left margin.
- The Title is at the bottom of the page.

Original layout
sequence



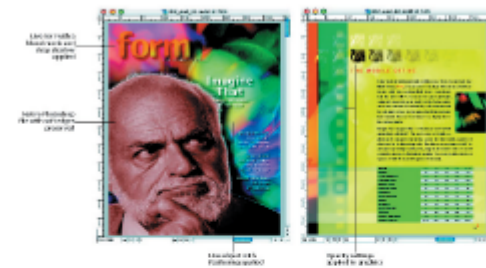
Next screen shot is
of the document
after text reflow



New Feature Highlights

Adobe InDesign 2.0 software sets new standards for professional layout and design, delivering the creative tools you've always wanted. How do we know? Because you—and designers and print professionals like you—us to define what a page layout application should really be. With InDesign 2.0, you can:

- **Capture your inspiration.** InDesign 2.0 introduces groundbreaking creative features that transform how you approach page design. For the first time, you can create editable drop shadows, feathering, and transparency directly in a desktop page layout application. You can even place Adobe Photoshop® files with transparency backgrounds and soft edges. In addition, InDesign 2.0 takes the pain out of creating tables, so you design. Expanded OpenType® font support and new preview controls complete the new creative features in InDesign 2.0.
- **Be the master of deadlines.** InDesign is packed with intuitive enhancements that streamline elaborate design tasks, saving you time and freeing you to be more



When you're in the flow of a creative idea, you want your software to work intuitively—putting the right tools at your fingertips without getting in your way. Adobe InDesign is designed to give you the creative freedom removing arbitrary software constraints and helping you produce

Setting new standards for professional layout and design

Capture Your Inspiration

creative. Now, with InDesign 2.0, you can prepare long documents as easily as you would short documents, including generating tables of contents and detailed indexes. The enhanced paragraph composer (formerly a composer) automatically sets optimal line breaks for each paragraph. An efficient new printing interface delivers precise, reliable results. And performance enhancements ensure that you can work more productively with

- **Work more efficiently through tight Adobe integration.** InDesign 2.0 integrates better with other leading graphics applications—Adobe Photoshop, Illustrator®, and Acrobat®—than any other page layout application. It shares commands, tools, palettes, and keyboard shortcuts with these applications, as well as core Adobe technologies such as the new Adobe Color Engine included in InDesign 2.0, Photoshop 6.0, Illustrator 9.0/10, and Acrobat InDesign 2.0 exports Acrobat 5.0 PDF files. The Japanese and Roman versions of InDesign 2.0 can exchange version 2.0 files because of their common file format. InDesign 2.0 also offers native support of the Mac OS operating system.



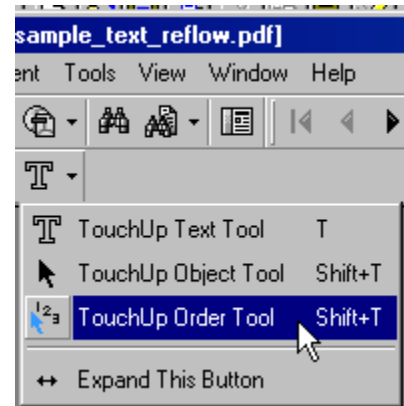
the page designs you've always envisioned. InDesign 1.0 and 1.5 set the stage for this by offering a range of innovations from multiple undo/redo to advanced typographical support. Now InDesign 2.0 redefines creative versatile transparency controls, easy-to-use table features, expanded OpenType support, and new preview controls that reduce on-screen distractions.

Figure 1. Graphics tools

Adobe® InDesign® 2.0

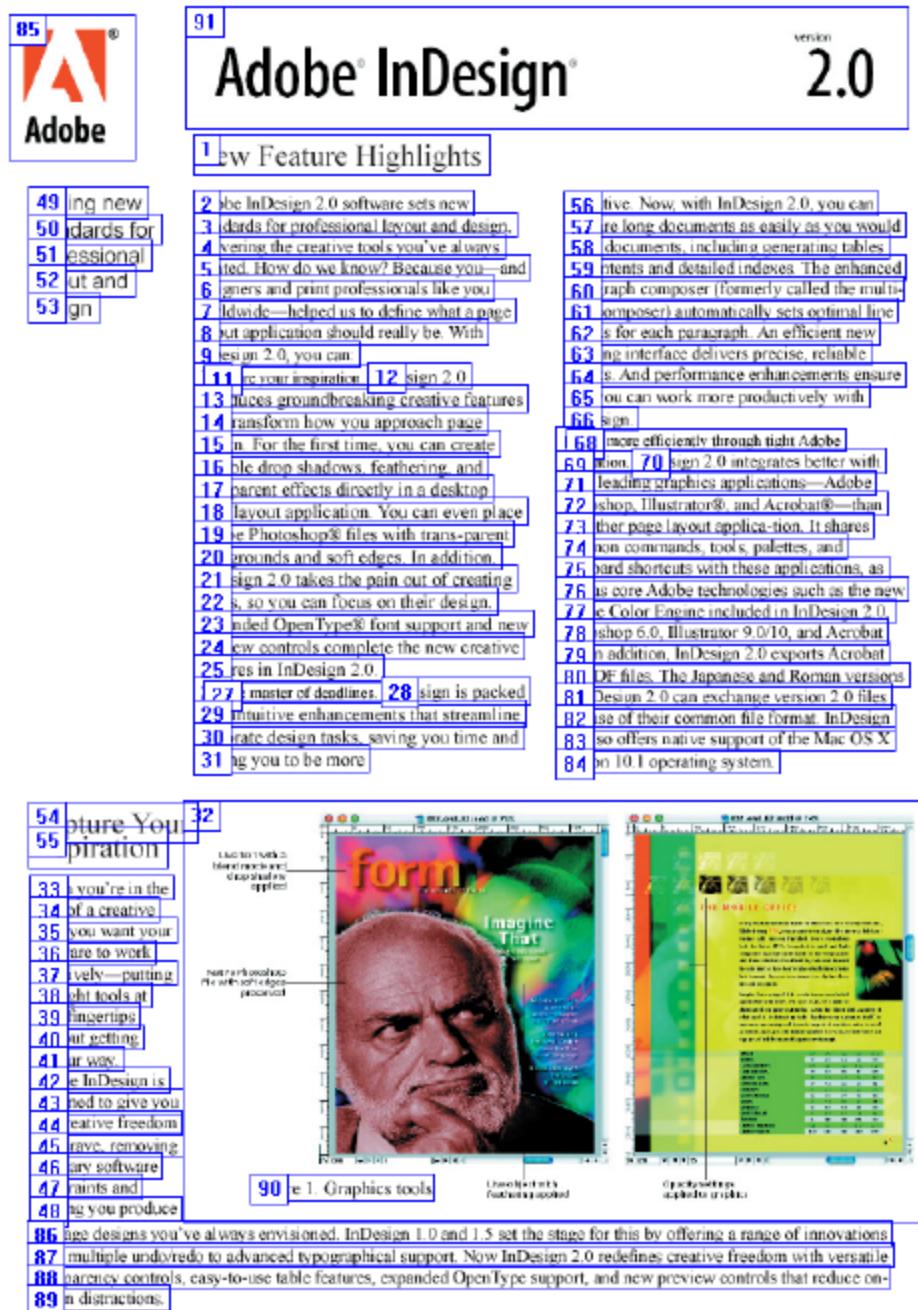
Fortunately, there is a Acrobat tool, the **TouchUp Order Tool**, which is found under the **More Tools** option next to the **TouchUp Text Tool** icon. Unfortunately, it is not the easiest or most intuitive tool to use. Adjusting the text flow order of a document can be tedious, especially for documents that are not visually simple.

3. Click on the **Actual Size** icon to return the document to normal view, then zoom in on the page to at least 150%.
4. Open the **More Tools** menu next to the **TouchUp Text Tool** and select the **TouchUp Order Tool**.



Instructor Notes: Changing the reflow order is not difficult, but it can be tricky. You are advised to go through this exercise yourself a few times before presenting it.

Notice the boxes placed around every frame in the document, each with a number, as shown in the following figure. The number determines the sequence in which the text in each box appears when the text is reflowed. A quick glance will show you why the reflow sequence of the elements on page one does not match the logical sequence. The Title is numbered last (91) on the page, and the two columns of the first paragraph are numbered in discontinuous sequences (31, then jumping to 56), to name a few issues.



Fixing The Reflow Order

Now the fun part. The numbering scheme is tricky to change without making a mistake. If you get to an unrecoverable state, click on some other tool, like the Hand Tool, to start over. Use the following advice loosely and try experimentation.

1. **Right-clicking** on one of the numbered boxes will display a reflow command menu. Start the text flow sequence with the Title. Therefore, non-intuitive as it

may sound, right-click any numbered box *except* the Title, and select **Make First**.

2. **Left-click** the Title frame.

Its number should change to 1.

3. **Left-click** each frame on the page in the sequence in which you want it to flow.

The numbers of the frames continue to change in sequence as you click on them.

4. You also can use the options in the right-mouse button menu to modify numbers in the boxes. The **Swap with Next** and **Swap with Previous** commands from this menu are very useful for fixing minor local reordering problems without endangering the order of previous frames. It takes a bit of practice to use the options effectively.
5. If you hold down the control key and left click a number, Acrobat places that number in the *next* box you left-click, and the numbering sequence continues from there. The rest of the numbers shift one place to replace the number you just moved. This seems to be the most effective method for fixing mistakes in sequencing.
6. Using the TouchUp Order Tool is similar to solving a sliding puzzle game; be prepared for surprises. For example, you lose the numbering sequence if you open the right mouse button menu. In that case, the next number box you click returns to number 1. You can fix that by **Ctrl-left-clicking** the number you really wanted there, and then clicking the box that now holds the errant number 1.
7. The page that is most out of sequence is page one. The other two are mostly correct. The TouchUp Order Tool can be frustrating, but do your best to fix the text sequence order without spending an inordinate amount of time on it.
8. Save the file and click the **Reflow** Tool to see the results.

You should see a marked improvement in the sequencing of the text over the original, similar to the screen shot below.
9. Narrow the size of the window and increase the zoom level.

The text should reformat correctly. You may have trouble changing the sequence number for the text caption under the main graphic, probably because it lies fully inside a graphics frame. As a consequence, it will end up out of sequence at the end of the document, as is the case in the screen shot below. The ultimate solution is not to fully embed captions inside graphics frames in the original document

Adobe® InDesign® version 2.0

Setting new standards for professional layout and design

New Feature Highlights

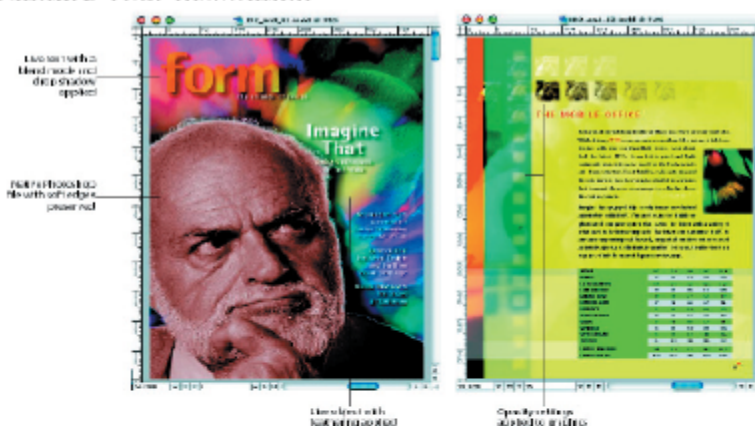
Adobe InDesign 2.0 software sets new standards for professional layout and design, delivering the creative tools you've always wanted to know? Because you—and designers and print professionals like you worldwide—helped us to define what a page layout application is. With InDesign 2.0, you can:

- **Capture your inspiration.** InDesign 2.0 introduces groundbreaking creative features that transform how you approach page design. Now, you can create editable drop shadows, feathering, and transparent effects directly in a desktop page layout application. You can place Adobe Photoshop® files with transparent backgrounds and soft edges. In addition, InDesign 2.0 takes the pain out of creating and focusing on their design. Expanded OpenType® font support and new preview controls complete the new creative features in InDesign 2.0.
- **Be the master of deadlines.** InDesign is packed with intuitive enhancements that streamline elaborate design tasks, saving you time to be more

creative. Now, with InDesign 2.0, you can prepare long documents as easily as you would short documents, including generating table of contents and detailed indexes. The enhanced paragraph composer (formerly called the multi-line composer) automatically sets optimal line and paragraph. An efficient new printing interface delivers precise, reliable results. And performance enhancements ensure that you can work productively with InDesign.

- **Work more efficiently through tight Adobe integration.** InDesign 2.0 integrates better with other leading graphics applications—Adobe Photoshop®, Adobe Illustrator®, and Acrobat®—than any other page layout application. It shares common commands, tools, palettes, and keyboard shortcuts, as well as core Adobe technologies such as the new Adobe Color Engine included in InDesign 2.0, Photoshop 6.0, Illustrator 6.0, and Acrobat 5.0. In addition, InDesign 2.0 exports Acrobat 5.0 PDF files. The Japanese and Roman versions of InDesign 2.0 can exchange files because of their common file format. InDesign 2.0 also offers native support of the Mac OS X version 10.1 operating system.

Capture Your Inspiration



When you're in the flow of a creative idea, you want your software to work intuitively—putting the right tools at your fingertips with your way. Adobe InDesign is designed to give you the creative freedom you crave, removing arbitrary software constraints and helping you design the page layouts you've always envisioned. InDesign 1.0 and 1.5 set the stage for this by offering a range of innovations from multiple advanced typographical support. Now InDesign 2.0 redefines creative freedom with versatile transparency controls, easy-to-use table of contents and detailed indexes, and new preview controls that reduce on-screen distractions.

Figure 1. Graphics tools



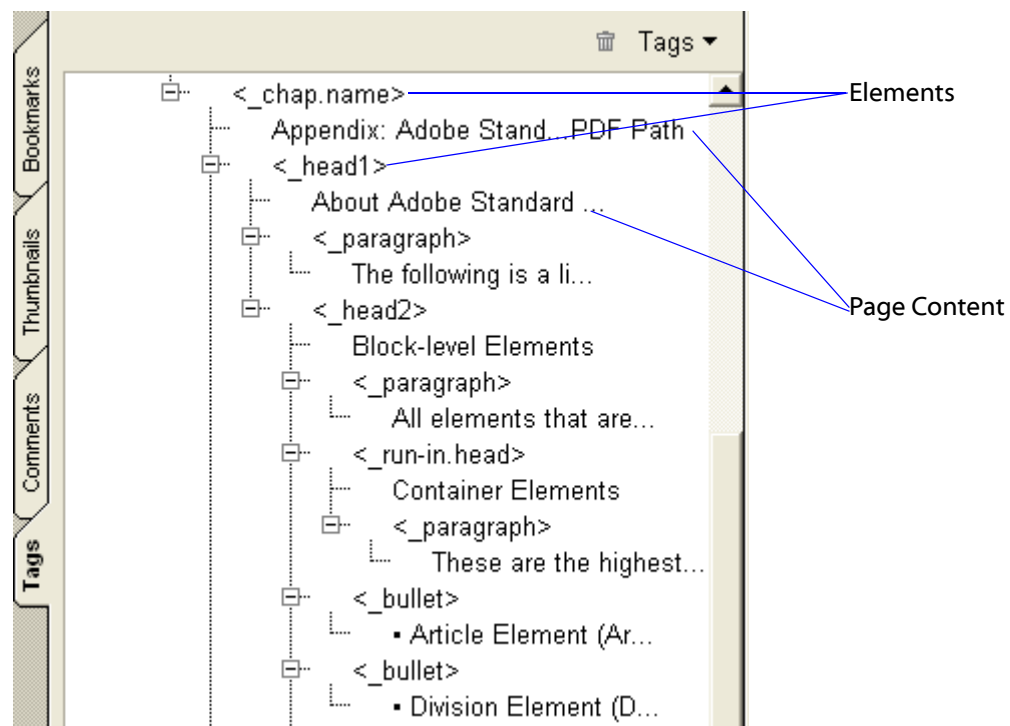


Fixing Content Problems

Instructor Notes: The following paragraphs and screen shots comprise a somewhat lengthy overview of selecting types of elements in a PDF document and creating new PDF Tags for them. In the exercise that follows, students will use the sample file [MktPlanSolution.pdf](#). Have them open that file now, so they can following along with these explanations.

You may find there are situations where the actual document content does not match what has been processed into the PDF tags. This can occur, for example, when a watermark has been picked up and incorporated into the text. There may be other circumstances where text that should be processed specially, like a URL or cross reference, is treated like regular text. You may decide that certain charts or tables require extended explanations to convey content that a sighted audience does not need.

You can use the Tags Palette to create new tag elements to address all of these situations. Adding new elements to the structure tree is a two-step process. First, you must create the appropriate element and insert it into the correct place in the structure tree. Second, you must tie specific page content to the new element. If you neglect to perform this last step, a screen reader ignores the new element.



Study the existing tags in your document to get a feel for what is being used to represent paragraphs, headings, lists, sections, and so on. For consistency, use the same tag types that are currently in your document. If you make a mistake or are not sure what tags to use, do not worry. Screen readers currently do not use the tag type when they process a document. For more information on the types of

tags that are available, refer to the appendix in *Advanced Techniques for Creating Accessible Adobe PDF Files*.

In this section, you will first learn about different tools that you can use to select page content in Acrobat. Once you are familiar with these techniques, you will learn the step-by-step procedure to add new elements to your Tagged PDF document.

Selecting Text

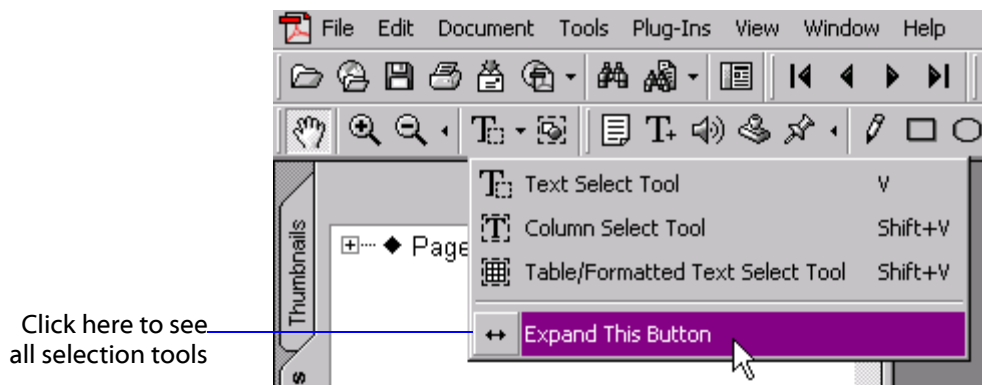
To create a child element from specific content in your PDF document, you must be able to select the desired text. Acrobat provides three tools that you can use to select text: the *Text Select* tool, *Column Select* tool, and the *Touchup Text* tool.

- The Text Select tool is useful for selecting small amounts of text, including discrete words. It is the most convenient tool to use. You can use the Text Select and Column Select tools interchangeably, although there are situations when the Text Select tool works where the Column Select tool does not.
- The Column Select tool only selects text within the bounding box that you sweep with the mouse. It is good for selecting large amounts of text or selecting chunks of text from a multi-column layout. The Column Select tool needs to be used with care. When it works, it is the most convenient of the selection tools. However, if the page content order is not correct, you may think that the Column Select tool has picked up text that it has not really selected. You can detect a problem by slowly selecting text with the column tool. If Acrobat highlights the selected text in an odd way, for example, line 1 highlights, followed by line 3 and then line 2, then the Column Select tool is not working properly. Always verify the text order of selected text by doing a save as **Text (Accessible)**.
- The Touchup Text tool can also be used to select text, although most users do not use it as frequently. Of the three available tools, it is the most reliable but you can only use it to select, at most, one line at a time. Reserve its use for situations where the Text Select tool and Column Select tool are not picking up the text you want.

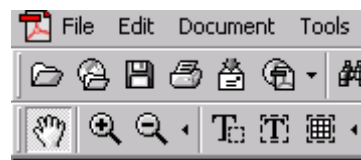
By default, only the Text Select tool is shown in the Acrobat toolbar. In order to expand the selection toolbar, click the down arrow next to the Text Select tool icon




and select **Expand This Button** from the popup menu.



The expanded toolbar is shown below. The table/formatted Text Select tool is used for copying tables in order to paste them into other applications. You cannot use it to tag tables.



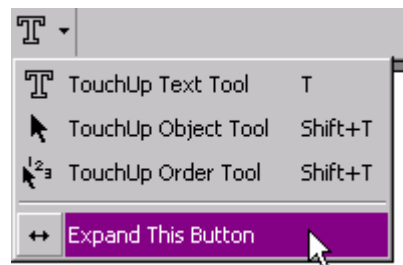
Selecting Graphics

Normally, you use the Touchup Object tool  to edit a graphic in your PDF document. The Touchup Object tool also provides a convenient means of selecting multiple objects that you wish to tag with a Figure element. Using the mouse, you can either left-click a specific object or sweep across the objects you wish to select, holding down the left mouse button. Acrobat highlights all the graphic objects found in the area.



Often, what appears to be a single graphic is a composite of several objects. To be on the safe side, sweep with the mouse to select the object.

By default, the touchup toolbar shows the last selected touchup tool. You can expand this by clicking on the down arrow next to the current touchup tool icon and selecting **Expand This Button** from the popup menu.



Creating a New Child Element

[Instructor Notes: This is a hypothetical example intended as an overview of the procedure. A real example follows in the next exercise.](#)

Once you have selected the text or graphic for which you want to create a PDF Tag in your document, you would then use the following general procedure to create a new Tag element for it in the Tags Palette. Assume for this hypothetical example that there is graphic on one of the pages of the document that lacks a PDF Tag and needs one.

To create a new element:

1. Open the tagged PDF document in Acrobat. Locate the page in the document that contains the graphic for which you want to create a new PDF Tag. Open the Tags Palette.

2. Decide where the new Tag should be placed in the PDF Tag hierarchy:

Select an element in the document which should come just before the graphic in the logical flow. If that is a text element, the Text Select tool would probably be the most appropriate one to use.

Open the Tags drop down menu and select **Find Element From Selection** to find that element's Tag position in the Tags Palette. Refer to ["Finding the Associated Element for Document Content" on page 9](#) for more information on how to find a Tag element associated with particular page content.

3. In some cases, you may find that there is already a PDF Tag associated with a particular document content, but that it is not the type you want, or that it is not appropriate for your purposes. If that is the case, delete the Tag element by selecting it in the structure tree and pressing the Delete key.

In general, the Tags you are interested in, and the ones that are critical for accessibility purposes, contain either text or graphics with alternative text. A number of other Tags may have been generated in the process of creating the PDF file which are superfluous for accessibility purposes. For the most part, these can be removed from the Tags Palette if you wish to do so. For example, there is a <Shape> Tag just above the <TextBox> Tags you worked with in the exercise [Fixing Logical Read Order Problems](#), which has several PDF_Path child elements. If you turn on Associated Content Highlighting, you will see that these Tags refer to the graphic frames around the text boxes, and that the <Shape> Tag itself refers to a group of three of these boxes. These Tags lend nothing to the accessibility of the document, and, unless you wish to add alternate text to them, can be deleted without affecting accessibility.

4. Find the Tag that is the parent for the new Tag you wish to create. The parent Tag may be considerably further up in the Tag hierarchy from the position of the new Tag. Typically, the parent Tag for the new Tag will be the same as for the adjacent element you selected in Step 2, above.

Click the element that will be your new element's parent Tag.

5. Open the Tags drop down menu, or select the parent Tag and right-click on it to bring up the Tags menu. Select **New Child Element**.
6. From the popup menu that appears, select the Tag element type you wish to insert.

For more information on the types of elements that are available, refer to the appendix in *Advanced Techniques for Creating Accessible Adobe PDF Files*.

The new Tag element becomes the *last* child under the parent Tag. If you do not see the new Tag there, you may have inadvertently placed it as a child element under the wrong parent.

7. Move the new Tag to the location in the Tags hierarchy you identified in Step 2, above.

The new Tag you just created is a placeholder. You must now tie the element to the actual page content.

8. Click the new Tag to select it. This is important, because the selected Tag will become the parent for the document content you are about to specify.
9. Using the appropriate selection tool, select the desired text or graphic. For this hypothetical example, you would use the Touchup Object tool.
10. From the **Tags menu**, select **Create Child Element From Selection**.
11. **Save** the PDF file.

Modifying the Tag Type of an Element

There may be situations where page content has been associated with the wrong element type. When this happens, you do not need to delete and recreate the element. You can simply modify the element's type.

To modify the type of an element:

1. Open the tagged PDF document in Acrobat, bring up the Tags palette, and locate the element that you need to change.
2. Click the element that you want to modify.
3. Open the Tags menu and select **Element Properties**.
4. In the **Type** field in the **Element Properties** dialog box, delete the existing Tag element type name and enter a new one.

For more information on the types of elements that are available, refer to the appendix in *Advanced Techniques for Creating Accessible Adobe PDF Files*.

5. **Save** your PDF file.



Incremental changes to a PDF file are added to the end of the file. To compress your PDF file, and make it more efficient, you should do a Save As (as PDF) to the same file name that you have been using.

Use your discretion to decide whether a graphic should be accessible. Graphics that are not important for page content are also referred to as *artifacts*. These can be ignored entirely if they do not have an associated PDF Tag. If you want a graphic to be accessible, always create a **Figure** Tag for it. After you have created a Figure Tag, make sure you edit the Tag's properties and enter alternate text for it.



Exercise: Creating New Tag Elements

In order to do this exercise, you must have Acrobat 5.05 installed on your machine. You must also have downloaded and unzipped the file **TagsPalette.zip** from the Accessibility training Web site.

In this exercise you will work with a partially Tagged PDF file. The file was originally created from an application that could not create Tagged PDF. You will practice tagging both text and graphic content. You can find the solution for this exercise in **MktPlanSolution.pdf**.

At the end of the exercise you will be able to:

- Create heading, paragraph, and figure elements
- Use the Column Select tool and Text Select tool to select text
- Use the Touchup Object tool to select graphics
- Add alternative text to a Figure element by editing the element's properties
- Verify that elements that you created are accessible

Tagging Text

1. Open **MktPlan.pdf** in Acrobat.
2. Open the Tags palette, expand the structure tree, and look over the current state of the document.

Use **Turn On Associated Content Highlighting** from the Tags menu to get a feel for how much of the document has been tagged.

On page 1, most of the content has been tagged except for the last heading and paragraph. You will add a Heading 2 and Paragraph element to the structure tree so this content will be accessible.

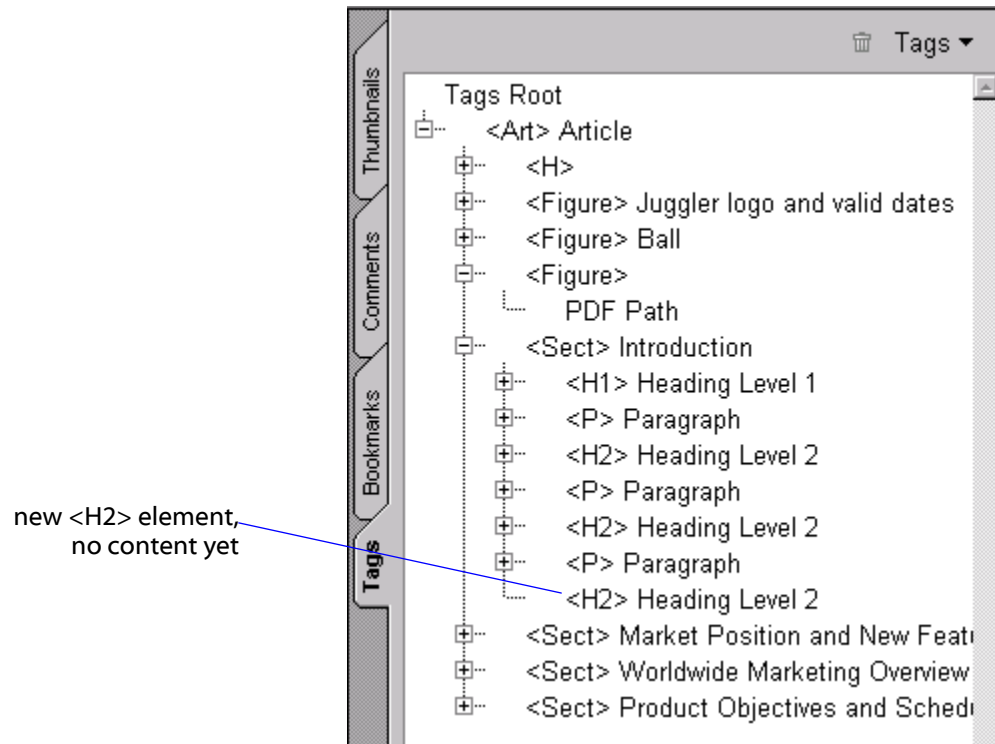
3. In the Tags palette, find **<Sect> Introduction**.


This is the parent element.

4. Click **<Sect> Introduction** to highlight it.
5. Click on the **Tags** drop down menu and select **New Child Element > Heading Level 2 Element**.

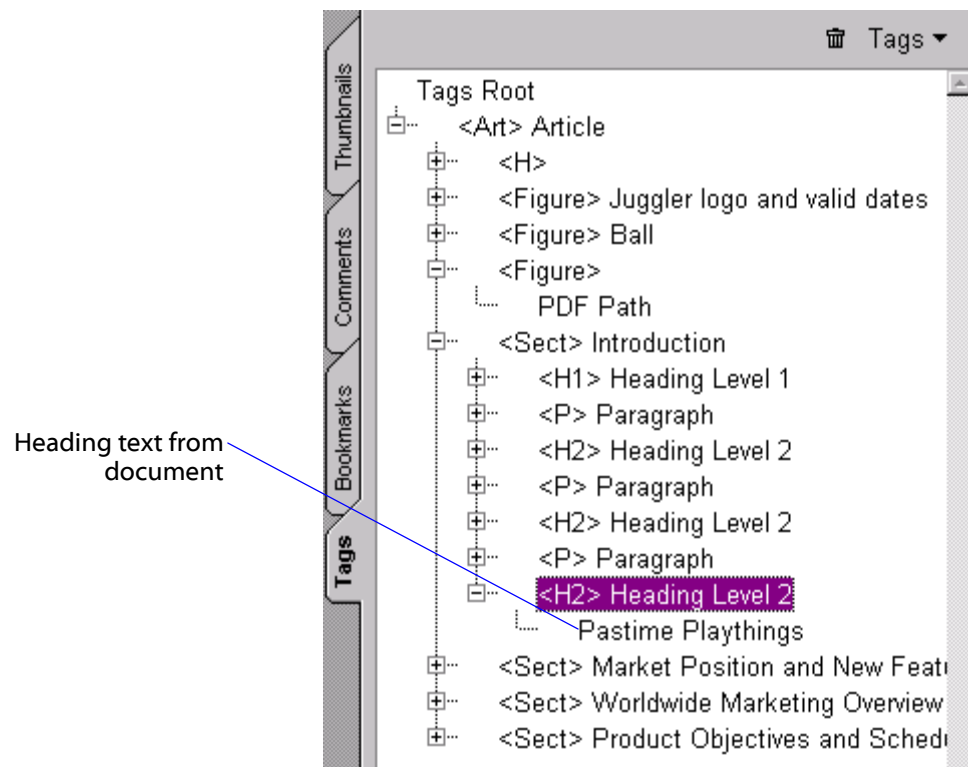
Acrobat will insert an empty **<H2>** element as the last child element under **<Sect> Introduction**.

Instructor Notes: When a New Child Element is created, it is always placed as the last child in the hierarchy under the tag element that was selected at the time the new child was created. This may be a little confusing at first. Often, the newly created tag will then have to be moved into the correct position in the logical order.



6. Click the new **<H2>** element to select it.
7. In the Acrobat toolbar, click the **Column Select** tool .
8. Move the mouse cursor over to the document pane.
Notice that the cursor is now an I-bar with a bounding box around it.
9. Hold down the left mouse button as you select the text heading **Pastime Playthings**.
10. Release the mouse button to complete the selection.
11. Click on the **Tags** drop down menu and select **Create Child Element From Selection**.

The structure tree should appear as below.




12. Now add a Tag for the paragraph below the Heading:

Click **<Sect> Introduction**.

13. **Right-click** on the **<Sect>** Tag to bring up the Tags menu and select **New Child Element > Paragraph Element**.

Acrobat will insert an empty **<P>** element as the last child element under **<Sect> Introduction**.

14. Click the new **<P>** element to select it.

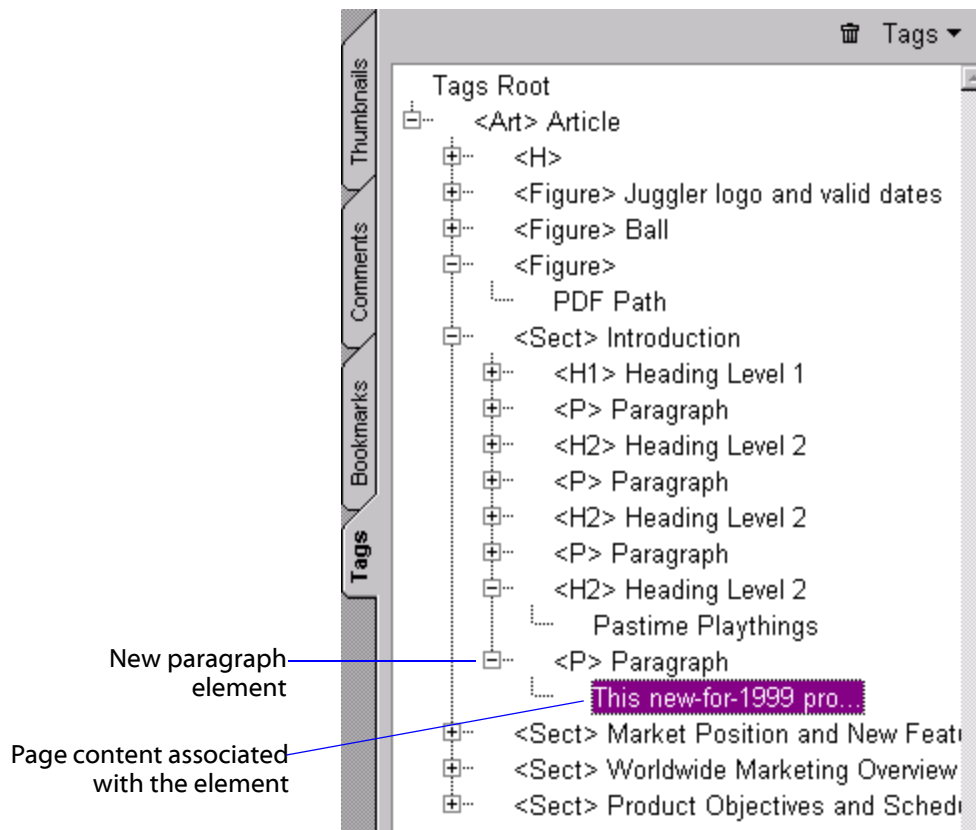
15. In the Acrobat toolbar, click the Text Select tool .

16. Move the mouse into the document pane and select the last paragraph on page 1.

17. Open the Tags menu and select **Create Child Element From Selection**.

18. Save your changes.

The structure tree should appear as below.



19. To test your changes, turn on content association and step through the elements under **<Sect> Introduction** to see if the content is in the right order.

20. **Save** the file as **Text (Accessible)**.

21. Open the text file with an editor and make sure the heading and associated paragraph for Pastime Playthings now appears at the end and in their entirety.

If the text does not appear in the file, you may not have created a child element based on the text in the document.

Note that if the new Heading and paragraph had not been the last elements on the first page, you would also have had to move the Tags to their correct position under the **<Sect> Introduction** Tag


Tagging Graphics

Instructor Notes: Students will not be tagging everything in the sample document, so there will be gaps in tag structure. The pie chart on page 3, for example, is not tagged

1. Open **MktPlan.pdf** in Acrobat.

On page 1, there is a graphic of a green car at the bottom of the page. You will create a figure element for this car.

2. In the Tags palette, find **<Sect> Introduction**.

3. Create a new **<Figure>** element under **<Sect> Introduction**:
Right-click on **<Sect> Introduction** and select **New Child Element > Figure Element**.
4. Select the **Touchup Object** tool . The green car is a single object. **Left-click** it to select it.
5. **Right-click** on the **<Figure>** Tag you just created, and select **Create Child Element From Selection**.
6. Add **Alternate Text** to the graphic by right-clicking the **<Figure>** element and selecting **Element Properties** from the popup menu.
7. In the **Alternate Text** field in the **Element Properties** dialog, enter the text *Sample vintage vehicle*.
8. At the bottom of page 3, there is a bar chart labelled "Worldwide Projected Revenue Goals For 1999 - 2001" showing differences in revenue projections for Jugler Toys versus its competitors for the years 1999 through 2001.

Add **Alternate Text** for the barchart figure, describing the percentages for each company.
9. Test your changes by saving the file as **Text (Accessible)**.

Acrobat saves the alternate text for the figure in the file. If you had a **<Figure>** element with no alternate text, it would be invisible to the screen reader. If you wish, you can use a screen reader to process the PDF file.

Extra Credit

[Instructor Notes: Use this exercise if you have time, or give it to the fast ones in the group. Step by step instructions are included for creating a tag structure for Lists, but students are on their own in regard to creating the table tag structure.](#)

MktPlan.pdf also contains a bulleted list and a table on page 4. The Acrobat Distiller will correctly convert lists and tables from the original authoring tool, for example, MS Word, InDesign, or FrameMaker, to Tagged PDF. **MktPlan.pdf**, however, was not originally created as a Tagged PDF document. As a consequence there are no Tags in the Tags Palette for those content elements and they will not be accessible by a screen reader.

In Tagged PDF, lists and tables are elements represented by the Tags, **<L>** and **<Table>**. They themselves are comprised of child elements **** and **<TR>**, which in turn are comprised of other child elements, **<LBody>** and **<TH>** or **<TD>**. Most screen readers recognize the tags that comprise lists and tables, and note to the user that a list or table is being read.

You may recognize the similarity of these tags to HTML. Each item in a list, **<L>**, should have its own list item, ****. Since an item in a list could potentially be comprised of several paragraphs, each paragraph should be tagged with a list body, or **<LBody>** element. The **<Table>** element should contain a **<TR>** element

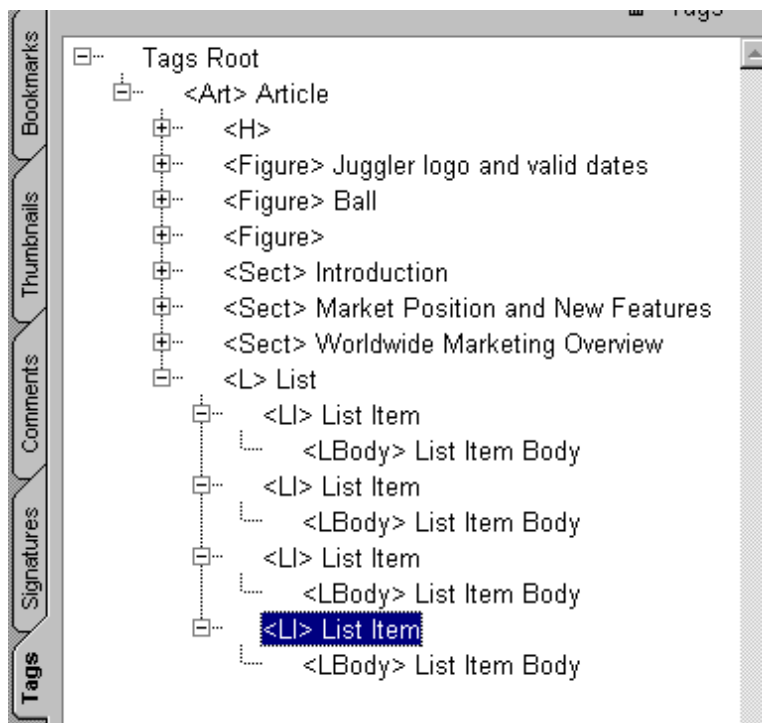
for each row, including the heading. Within the **<TR>** element, there should be a **<TD>** (data cell) or **<TH>** (heading cell) element for each cell in that row.

In this supplemental exercise, use the following two step procedure to add tag structure to the list and table on page 4:

1. Determine what elements are needed in the Tag structure to represent the content of the list and table, and add those tag elements to the Tags Palette.
2. Select the content elements in the document that correspond to the tag elements in the Tags Palette, and add them as child elements of their respective Tags.

Start by analyzing the list on page 4. It needs a **List** tag, and four **List Item** tags each containing a **List Item Body** tag. The List element is going to be a child element of the **<Art>** Article tag. It should be placed at the bottom of the hierarchy, which will happen anyway, by default, because the last tagged content in the document is the graph on page 3.

1. **Right click** on the **<Art>** Article tag and select **New Child Element > List Element**. The **<L>** List tag should appear below the last **<Sect>** tag and at the same level in the tag structure.
2. **Right click** on the **<L>** List tag and select **New Child Element > List Item Element**. The **** List Item tag should appear below the **<L>** List tag and one level deeper in the tag structure.
3. **Right click** on the **** List Item tag and select **New Child Element > List Item Body Element**. The **<LBody>** List Item Body tag should appear below the **** tag and one level deeper.
4. **Repeat** steps 1 through 3 for the other three bullet items in the list on page 3. When finished, the new Tag hierarchy should look like this:



Instructor Notes: The trick in getting the placement right in the tag hierarchy is to always right click on the parent tag, or select it, before selecting New Child Element: the new element is always created as the child of the selected tag. If students make a mistake in the placement of the tags, they can move them to the correct position or delete them and try again.

Next, you need to specify what actual content elements in the document are represented by the new tags to the Tags Palette. In the Tags Palette, the last, or most deeply nested tag in a parent-child structure is the one that is associated with the content in the document. So in this case, the <LBody> List Item Body tags are the ones you are going to connect to each of the bulleted items in the list on page 4.

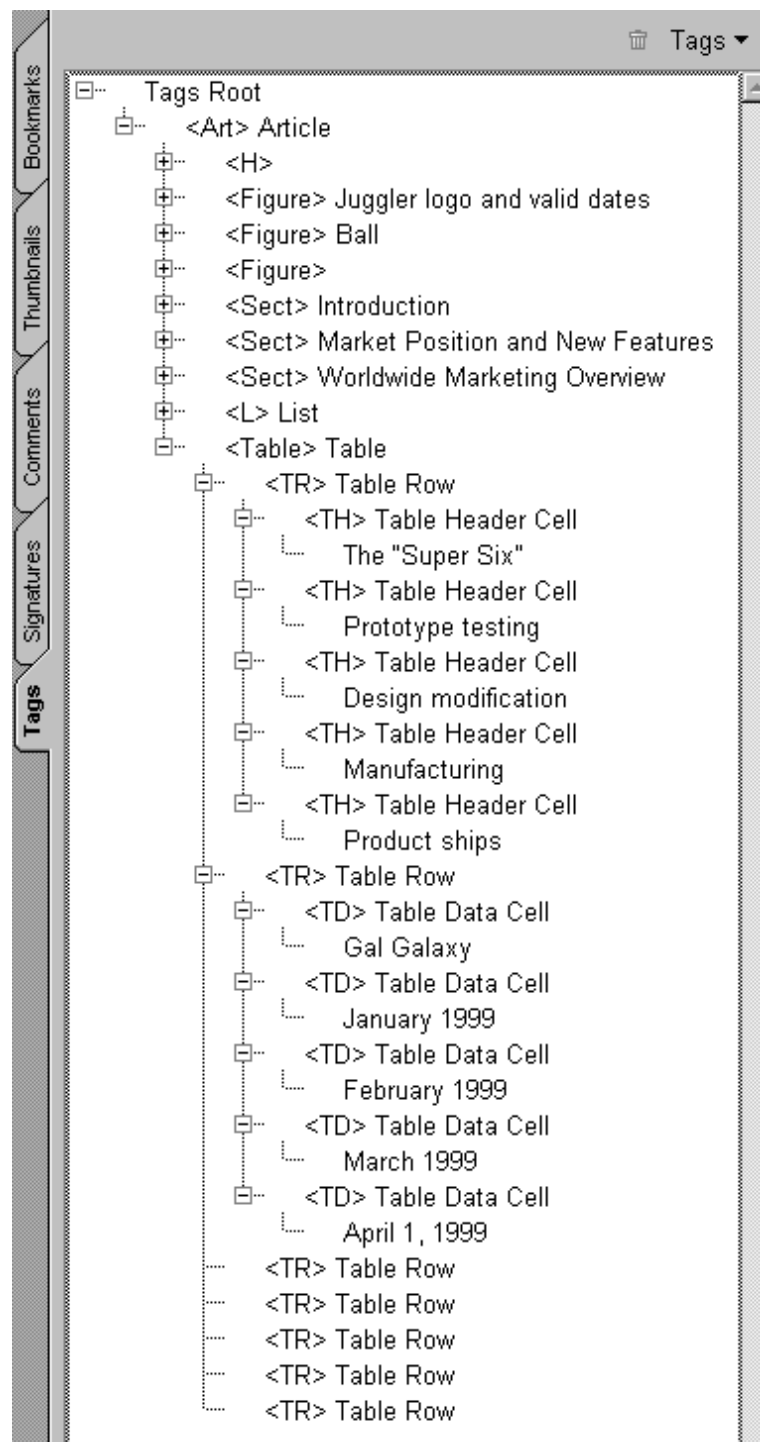
5. Go to the list on page 4. You can use either the **Text Select** tool or **Column Select Tool** to select the entire text of the first bulleted item.
6. **Right click** on the first **<LBody> List Item Body** tag in the List structure and select **Create Child Element From Selection**. If you click on the plus sign that now appears to the left of the <LBody> tag, you will see that the text from the first bulleted item in the list on page 4 has been added to the tag structure as a child of the first <LBody> tag.
7. **Repeat** Steps 5 and 6 for the other three bulleted items in the list.
8. **Save** the file as **Text (Accessible)** and check to see that the list items are now included. Alternately, use a screen reader to see if the list items are read, and secondarily, to see if the bulleted items are identified by the screen reader as items in a list.

Now you are on your own. Using the procedures you learned above for creating PDF Tag structure for the list on page 4, create a PDF Tag structure for the table at the bottom of page 4.

- Create the tag structure for the table in the Tags Palette.
- Select the text in the table's header and body cells and associate it with the appropriate tags in the Tags Palette. Selecting the text in the table cells can sometimes be a bit tricky. Use the Touch Up Text Tool, which seems to work best in terms of being the most controllable when selecting text in just one particular cell.

Even using the Touch Up Text Tool, you may have trouble selecting the "Manufacturing" cell heading because it actually exists in its own text box behind the text box that contains the other cell headings. A trick is to skip the "Manufacturing" heading and go on to select "Product ships" and associate that with the last <TH> tag. Then, with "Product ships" still selected, press the right arrow key twice, which will cause the cursor to jump back to the "Manufacturing" text box, from which you can select it and associate it with the correct <TH> tag.

- When complete, the <Table> tag hierarchy should look like the following screen shot, in which the first header row and the first data row tags have been expanded:





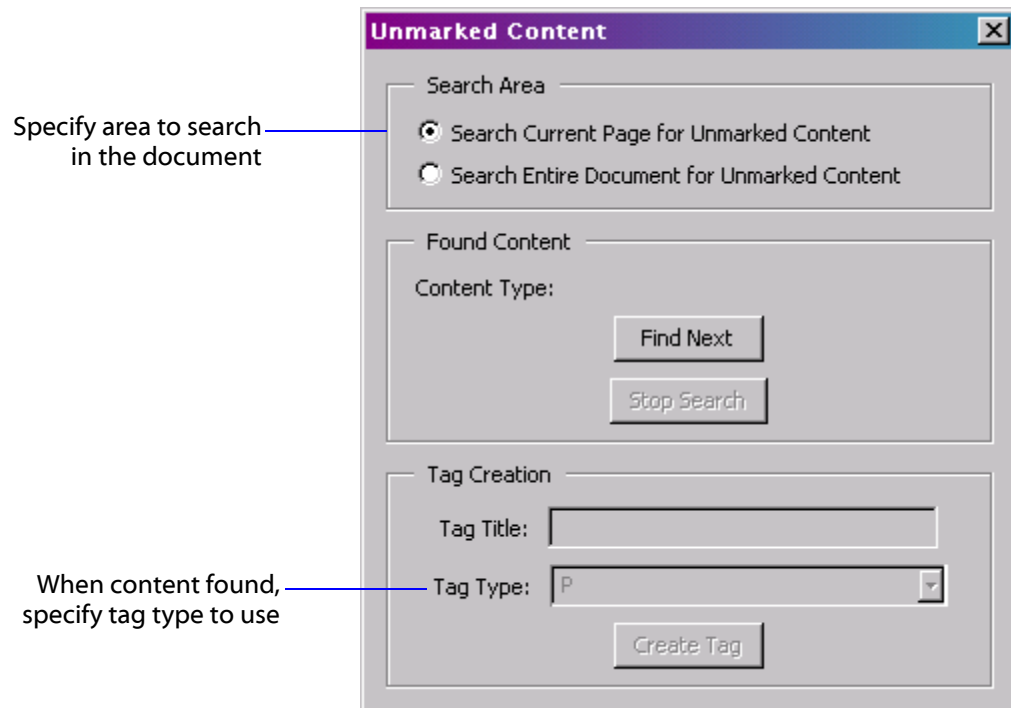
Converting Missing Content

Occasionally, the conversion process to Tagged PDF neglects to create corresponding tag elements for some of the page content. If page content has no associated tag element, it will not be accessible to a visually-impaired person. It is not necessary to convert every single piece of text and graphic. Some text, like page headers and footers, may not provide any significant information to a visually-impaired person. Sometimes graphics are used to add color and visual appeal to a document. These document elements, which are referred to as *artifacts*, do not need corresponding tag elements if they add nothing to the intended meaning of the document.

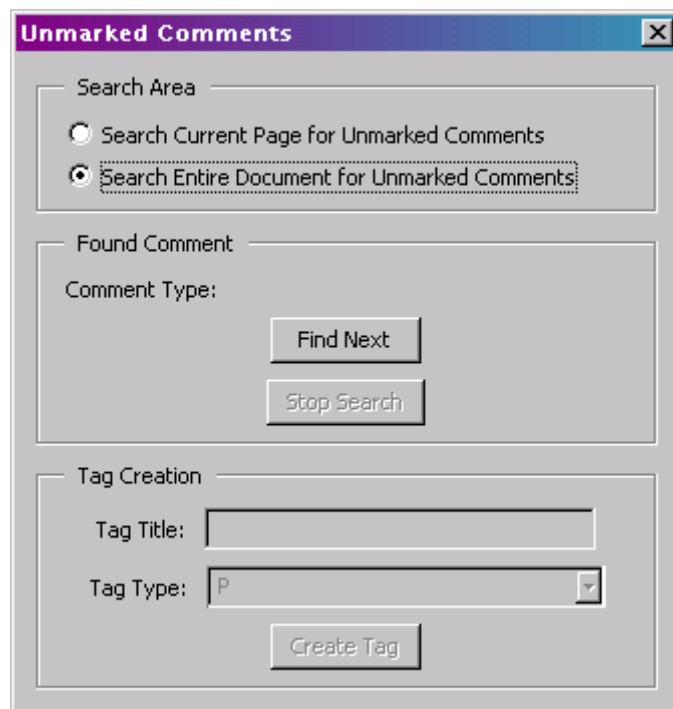
Although the Accessibility Checker can locate untagged page components, you can also use two Tags palette commands to dynamically create a tag element for document content that is not in the structure tree. The commands are **Find Unmarked Content** and **Find Unmarked Comments**. Both of these commands allow you to create a tag of your own specification for the found object. You can also elect to skip creating a tag. When Acrobat finds an object that has no associated tag, it highlights the object with a border. You can use the document pane to examine the object and decide whether you want to tag it. If you do tag it, you need to specify the type of tag to use and, optionally, a tag title. The tag title allows you to easily figure out to what the tag corresponds without having to use content association.

Acrobat automatically inserts a dynamically created tag as the last child element of the currently selected tag. Therefore, to avoid moving tags around, you should determine which content is missing tags and where the surrounding tags are located in the structure tree. Once you know where the missing content's element should go, select the parent tag in the palette pane and then use one of the **Find...** commands to search just the current page.

Use **Find Unmarked Content** to search for text, lists, tables, and so on that were not converted. If you have composite graphic objects that were not tagged, attempts to create a tag for each individual graphic object. You have some options for how to approach composite graphics. You may choose to use the **Find Unmarked Content** command in spite of the fact that it will create unnecessary tags for individual graphic objects, since the hard work is in finding and selecting the correct content to begin with. You can then move the content to the right place in the Tags hierarchy and delete the empty tags. You can choose to rearrange the structure tree to organize the individual objects, or you may decide to use manual tagging for composite graphics.



Due to PDF implementation details, links, such as cross references within a document, and form fields are considered to be comments or *annotations*. Use **Find Unmarked Comments** to search for links, annotations, or form fields that are not present in the structure tree.



Be aware that link elements differ from annotation and form field elements in that links have page content counterparts. Consequently, using **Find Unmarked Comments** to locate untagged links only accomplishes half the job. You must still manually tie the new link element back to the appropriate page content to make it accessible to a screen reader. If the link is embedded in a paragraph, you also have to manually split the paragraph into two separate paragraph tags, one coming before the text of the link, and one following the text of the link.

You need to use a screen reader to verify that a link is accessible. A screen reader uses audio cues to let a user know that the text it is reading is a link that can take the user to a different part of the document or to a Web site. When Acrobat saves a file as **Text (Accessible)**, links are not treated any differently than regular text, so using Text (Accessible) is not a useful way to test whether links are active in the logical structure.



Exercise: Converting Missing Content to Tags

In order to do this exercise, you must have Acrobat 5.05 installed on your machine. You must also have downloaded and unzipped the file **TagsPalette.zip** from the Accessibility training Web site.

In this exercise you will work with a Tagged PDF file with cross references that are not distinct tag elements. You can find the solution for this exercise in **UserGuideSolution.pdf**.

Finding Unmarked Comments

Instructor Notes: Reinforce making a copy of the exercise file in Step one, otherwise someone won't do it. They will need a copy of the original in order to compare the original with the modified version.

1. Make a copy of **UserGuideStart.pdf**. You will need to refer to the unmodified file later.
2. In Acrobat, open the copy of **UserGuideStart.pdf**.
3. Open the Tags palette.
4. First, locate the untagged cross references and figure out where they should belong in the structure tree.

Select **Tags > Find Unmarked Comments**. Select **Search Entire Document for Unmarked Comments**.

5. When Acrobat finds an unmarked comment, the **Tag Title** and **Tag Type** fields will become available for selection. You may need to scroll in the document pane to find the highlighted link. Make note of the page number and general location of the cross references, but do not create tags for them yet. There are only two unmarked links, on page 9, *Update Student Information* and the numeral 5 in *page 5*.
6. Use the **Text Select Tool** to select any part of the line containing the link *Update Student Information*. Then use **Tags > Find Element From Selection** to determine approximately where in the Tag hierarchy the cross reference links should be inserted.
7. In the structure tree, select the element that will be the parent of the cross reference links.

Hint: It should be an **<LBody>** element that corresponds to the text for the third numbered list item on page 9.

8. Once more, select **Tags > Find Unmarked Comments**. You may have to position the cursor at the top of the document before using the Find command again. This time you will create the link elements.
9. Enter the following information into the **Unmarked Comments** dialog

- **Tag Title:** *Update Student Info*
- **Tag Type:** Open the drop down menu and select **Link**.

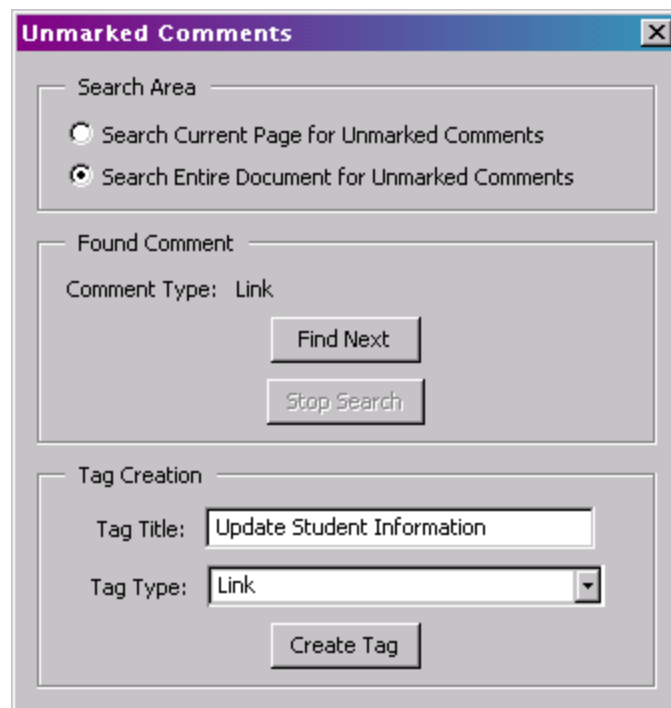
10. Click **Create Tag**.

11. Click **Find Next** to continue your search.

By default, the tag title and tag type are not cleared when Acrobat finds the next unmarked link.

When Acrobat finds the second untagged link, enter the following information:

- **Tag Title:** 5
- **Tag Type:** Open the drop down menu and select **Link**.



12. Save the file as **Text (Accessible)**.

13. Save the original `UserGuideStart.pdf` as **Text (Accessible)**.

Do the files look different? You should notice that the files are identical. If you use a screen reader to process the PDF file you just edited, it does not treat the links any differently than the other text in the document. What is going on?

You may remember that links have to be tied to specific page content. This is the missing piece of the puzzle. In addition, you probably did not modify the original paragraph that included the two links to separate the regular text from the links.

Fixing Missing Comments

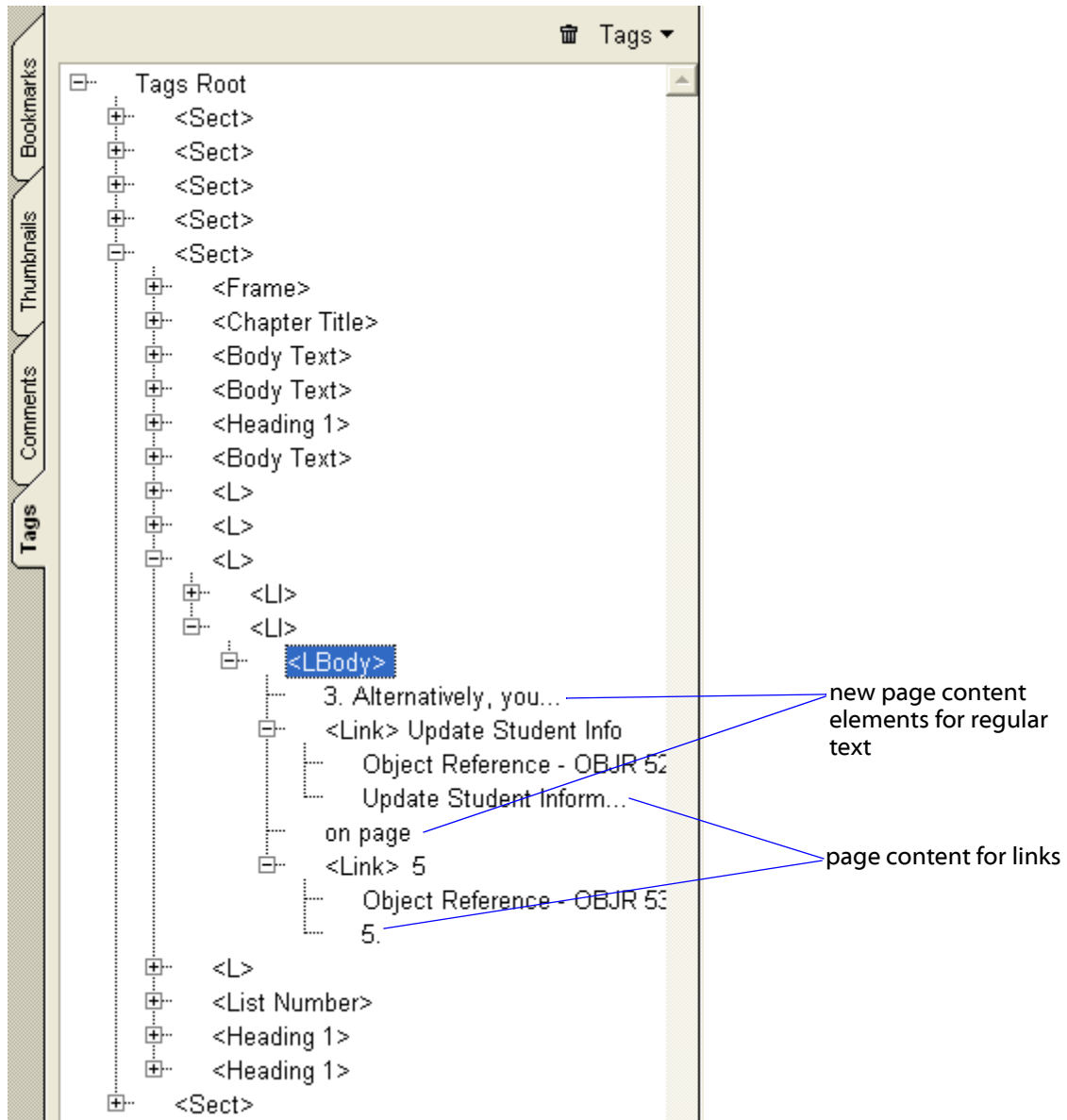
In this step, you will create a compound child structure for the <LBody> tag of the paragraph that contains the two links. You will create a child element for the first part of the paragraph up to the first link and for the part of the paragraph between the two links. You will add the text of the links as child elements under their respective <Link> tags, and then arrange them in the correct order under the <LBody> tag for the paragraph.

1. Use one of the methods you've learned to locate the **<LBody>** parent tag of the two Link elements you just added to the Tags Palette. Click on that **<LBody>** tag to select it.
2. Using the **Text Select Tool**, highlight all the text in list item three, including the list number "3.", and everything else that comes before "Update Student Information".
3. **Right click** on the **<LBody>** tag and select **Create Child Element From Selection**.
4. **Move** the new child element into the correct place, just below the <LBody> tag.
5. Use the **Text Select Tool** to select the text "on page" between the two links, then **repeat** Step 3.
6. Move the new child element into place between the two links.
7. Notice that the text of the links now exist as separate child elements of the **<LBody>** tag. You want them to be child elements of the **<Link>** tags, however. **Move** them into the correct place using the mouse.

Alternately, you could select the text for each link in the document pane, then select the appropriate **<Link>** tag in the Tags Palette, and then select **Create Child Item From Selection**. However, simply moving the tags is easier. You may want to delete any empty tags that appear in the modified paragraph structure, although empty tags are harmless.

Using whatever method you choose, rearrange the Tag structure until it looks like the one shown in the screen shot below.

8. Test your changes using a screen reader. The links should be correctly identified.





Summary

In this module, you learned how to modify the Tag structure characteristics of a PDF file for accessibility purposes. You were introduced to a number of Acrobat commands and utilities available for the post processing modification of a PDF document's logical structure. You were reminded that it is always preferable to control the PDF structure in the original authoring tool, prior to generating the PDF file, if possible. This is because modifications to the PDF Tags Palette cannot be saved and reapplied if the PDF document is re-generated from the original source. Not all authoring tools generate structured PDF however, in which case the Acrobat tools you have been introduced to in this module will be essential for developing an accessible PDF document.

The topics covered in this module were are follows:

Logical Read Order and Page Content Order

Logical read order, logical structure, and logical flow refer to the order in which contents of a document will be accessed by devices like screen readers. The logical flow is determined by the hierarchical structure of the tags in the Acrobat Tags Palette. The logical flow should conform to the sequence in which the document was intended to be read. Altering the logical flow by altering the structure of the Tags Palette does not affect the physical appearance of the document.

Page content order, or text reflow order, refers to the sequence in which the contents of a PDF document are presented when they are reflowed into different display formats. A display format, in this case, might be a handheld PDA. More relevant to accessibility issues, the format might be greatly magnified text within a narrow viewing window, used by someone with low visual acuity. In both cases, the reflow order determines the sequence in which the document content elements are displayed, and the reflow feature allows the contents of the document to be correctly formatted and wrapped to the dimensions of the display window. Reflow order is not necessarily the same as the visual order of elements in the document, nor is it necessarily the same as the logical order of the document. Both logical order and reflow order should be attended to when creating accessible documents.

Fixing Content Problems in the Tags Palette

You learned how to add structure elements to the PDF Tags Palette using the commands in the Tags menu. You also learned to use various Acrobat Selection tools for selecting text and graphics in a document, and how to use Tags menu command to associated document content with new and existing tags in the Tags Palette. You learned how to move elements in the Tags Palette into their appropriate places.

Finding and Converting Missing Content

You learned how to use the Find Unmarked Content and Find Unmarked Comments commands from the Tags menu to find text, tables, graphics, and

cross reference links that were not tagged. You used the Find Unmarked Comments command to find two untagged cross reference links in the document, and you used various previously acquired skills to add a functioning tag structure for the links in the Tags Palette.