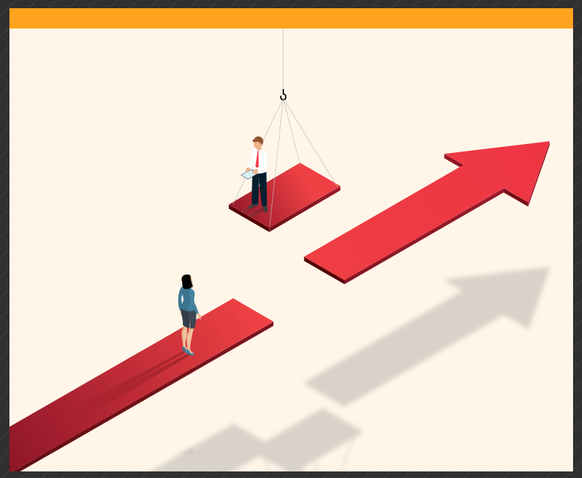
CreativePro magazine

Accessibility SuperGuide

+ Accessible PDF & EPUB Workflows
+ Writing Alt Text
+ Accessible Type & Color
+ Accessible Forms & Charts
+ Automating Accessibility Checks

Special Sampler

IN THIS GUIDE

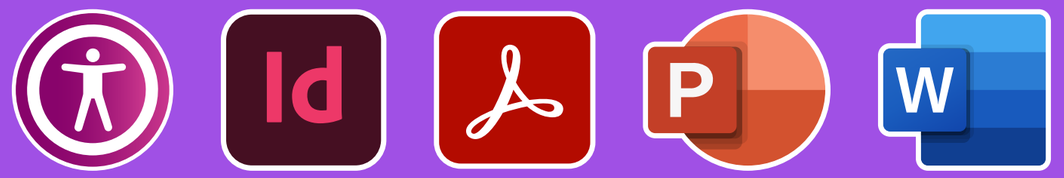


4 Ten Steps to Creating Accessible PDFs

The step-by-step process for exporting InDesign files to accessible PDFs

CreativePro

Design + Accessibility Summit



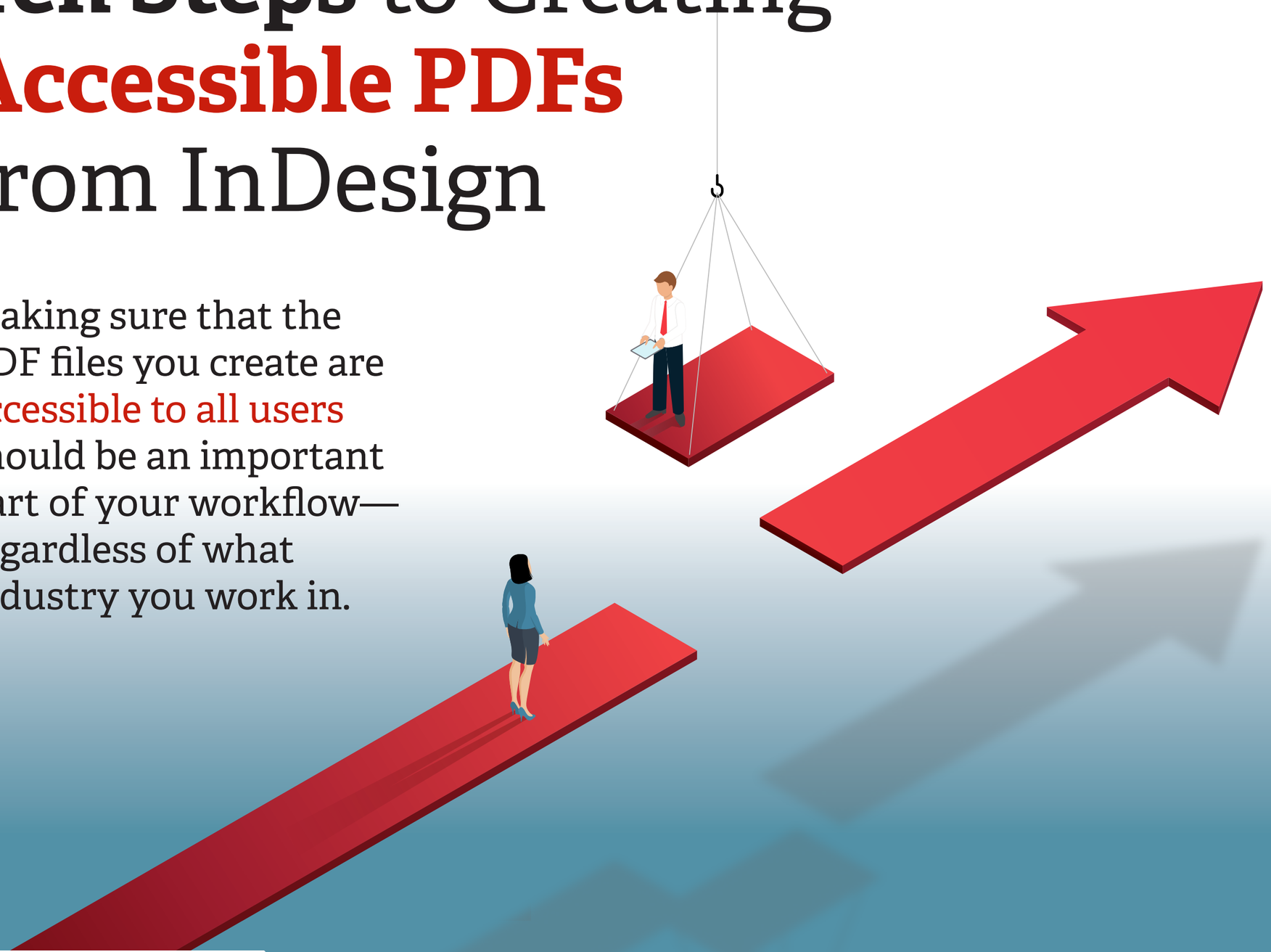
AN ONLINE EVENT

Learn more at CreativePro.com/Events

Ten Steps to Creating **Accessible PDFs** from InDesign

BY Chad Chelius

Making sure that the PDF files you create are accessible to all users should be an important part of your workflow—regardless of what industry you work in.



10 Steps to Accessible PDF

When I mention the term "accessibility" to most InDesign users, I typically get a blank stare. As designers, we're visual people and we focus on the appearance of what we create because that's what we've all been taught to do. But appearance isn't everything. We also need to consider how people with low-vision, blindness, or mobility impairments can consume our content. Those folks often use assistive software commonly referred to as "screen readers" that read content (usually a PDF or website) aloud. In an era where even household appliances can understand and respond to human speech, this may sound simple enough. But unfortunately, not all PDF files are created equal. Having actual live text in the document is a very important factor in making a file readable, but there are several other factors that also come into play.

To understand what it's like for a non-sighted user to access the content of a PDF, let's use the analogy of a printed newspaper. Imagine you open a newspaper in the morning looking for the news, only to discover that it contains nothing but blank pages. How would that make you feel? Or imagine opening the newspaper and there's text on the page, but no headlines to tell you where one article ends and the next one begins. Your only option would be to read the newspaper from beginning to end. These examples are exactly what an impaired user may experience when a PDF file is not made to be accessible.

When we talk about making documents accessible, we use the term "remediation" to describe the process of generating a PDF file that can be read by non-sighted users. That makes you the "remediator." There are two primary methods of remediation. The first is to take an existing PDF file and modify it in Acrobat or another PDF editor so that it can be read aloud by assistive software. This method, however, is the time-consuming, arduous, and expensive way of making an accessible document.

The second—and much better—way is to start with the source application and add properties that facilitate the creation of an accessible PDF file. Not all source applications give you this control, but applications like InDesign, Microsoft Word, and Microsoft PowerPoint allow us to do it efficiently.

In this article, I'll be focusing on the steps for creating accessible PDFs from InDesign. Although we've been able to accomplish this task to some degree in InDesign for years

(CS5.5 was the first version to offer accessibility features), the InDesign team has continued to improve things with each release. So before trying to make your documents accessible, be sure you're using the most recent version of InDesign that fits your workflow.

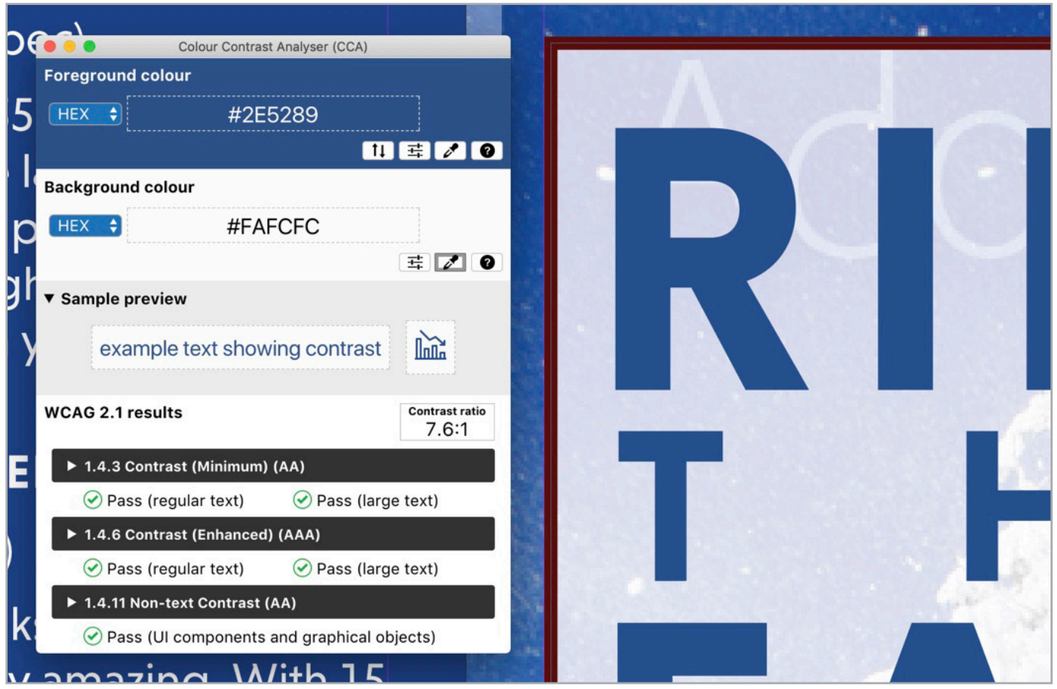
Step 1: Review Your Document

I know it sounds obvious, but just as you wouldn't dive in and start designing a project until you understand the goals and objectives, you need to get familiar with the content before you can start adding accessibility properties to your InDesign file. Evaluate things such as how objects are ordered on the page. How many heading levels are in the project? Are there tables? Are they complex or built in an inefficient way? Which elements require special attention? Once you have an understanding of what content is in the document, you can get to work adding the appropriate properties to the file and address any issues that you've encountered.

Step 2: Check Color Contrast

As designers, we tend to naturally use colors that create sufficient contrast between text and what's behind it on the page, but this is an important factor to consider for low-vision and even colorblind users. To check color contrast of a document, I like to use a product called the Colour Contrast Analyser (CCA) by The Paciello Group. This free tool allows you to pick the color of your text as well as the background color behind the text and reports whether the text has sufficient contrast at various sizes based on the WCAG 2.1 requirements (**FIGURE 1**). WCAG is the Web Content Accessibility Guidelines—one standard used to determine if web and PDF content is accessible. If your text color does not have sufficient contrast against the background color, consider making a change in the InDesign file. It's much easier than addressing contrast problems after the PDF file has been created.

FIGURE 1. The Colour Contrast Analyser allows you to check the contrast of the text to the background color behind it.



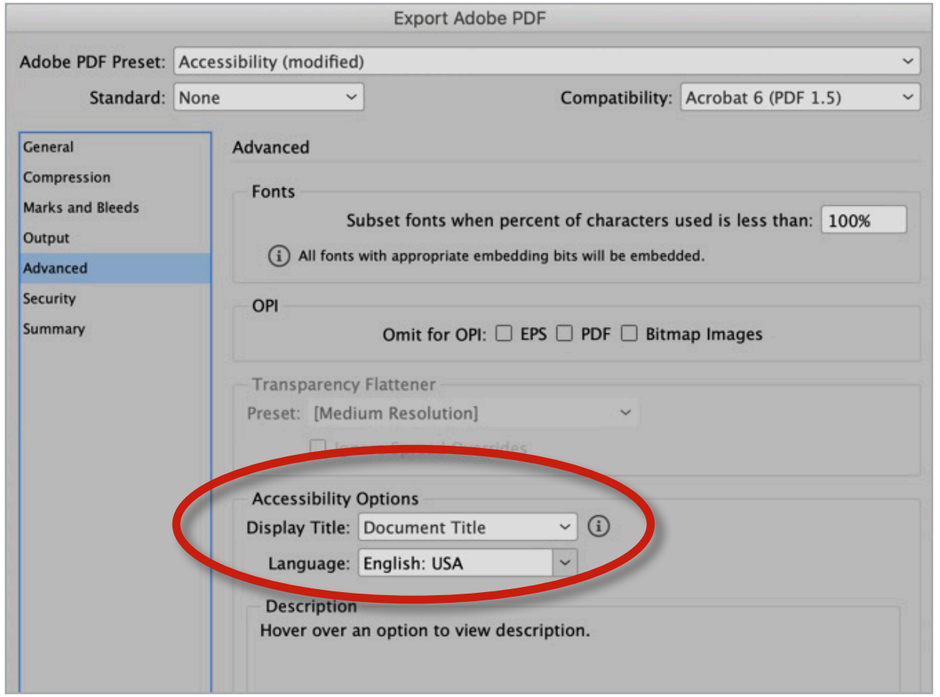
Step 3: Add Metadata

When it comes to metadata, the Title and Language properties are always required for accessibility. Metadata can be added to InDesign documents either in Adobe Bridge (in the appropriate field of the Metadata panel) or by choosing **File > File Info** from within InDesign when your document is open.

Although the Title and Language fields are the only required metadata elements, I encourage document remediators (remember, that's you) to add Author, Subject, and Keyword information to the file as well. This is especially important for public-facing documents that will be posted online. As a side benefit, adding additional metadata will assist with SEO, so Google and other search engines have a more complete picture of what's in the document.

Adding the document Title is a multi-step process. First, you need to add the title to the Title field (in InDesign or Bridge). Then, you also must set the title to be displayed in Acrobat when the PDF file is opened. We all have files with obscure names like "flyer_fall_2019_R9a_final.pdf." This name is just as useless to a non-sighted user as it is to a sighted user.

FIGURE 2. Defining the display title and language in the Export Adobe PDF dialog box



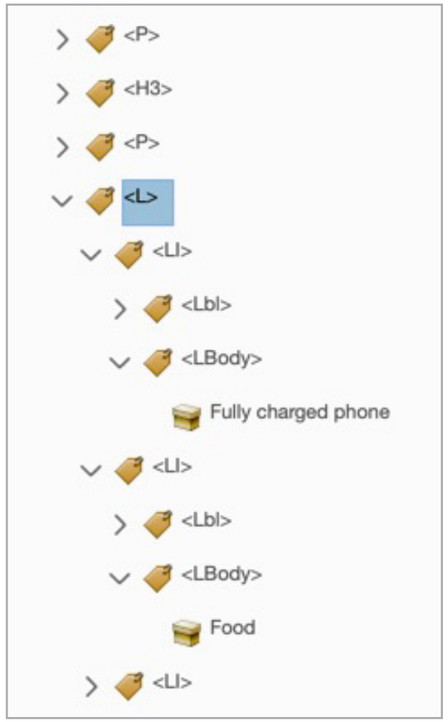
You can do that in Acrobat by using **File > Properties**. But, again, it's better to set it up in InDesign during the export process: to display the title and define the language of the document, you can click on the Advanced category (in the Export PDF dialog box) and choose Document Title from the Display Title drop-down menu (**FIGURE 2**). Then choose the language from the Language drop-down menu.

Step 4: Use Real Bulleted and Numbered Lists

Lists in an accessible PDF file are treated in a specific way to let the user know that a sequence of paragraphs belongs together as a list. Sighted users identify these elements automatically when they see the bullet characters and numbers at the start of each

paragraph. Fortunately, ensuring that bulleted and numbered lists are structured properly in the exported PDF is actually quite simple to do in InDesign (and you're probably doing it already). Just make sure all your lists are properly formatted with automatic bullets/numbering from the paragraph style. Or, at a minimum, make sure that you use the bulleted or numbered list formatting buttons to format the respective text. Just avoid manually typed numbers and bullets. This way, your lists will be structured properly in the all-important Tags panel in Acrobat (more about that later). (FIGURE 3)

FIGURE 3. When using the bulleted and numbered list feature in InDesign, the tags in the exported PDF file are properly structured.

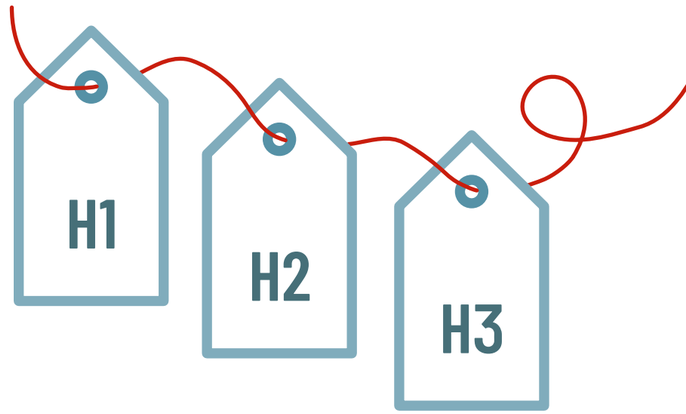


Step 5: Use Headings for Hierarchy

Headings are one of the ways we add hierarchy and structure to the text in the documents that we design. As designers, we do this naturally. We usually make the most important text the biggest, slightly less important text we still make big but a bit smaller than the really important stuff. It's just logical, and gives visual readers the opportunity to glance over a document to find information quickly. This same benefit should be provided for non-sighted users and is accomplished by applying appropriate heading tags to the headings in a PDF document.

Accessible PDFs provide for 6 heading levels from 1 to 6, with H1 being the most important headings in a document and H6 being the least important headings in a document. Assistive software allows users to employ a keyboard shortcut to read all of the headings in a document, just as a sighted user scans the headings in a document to find the topics they're interested in.

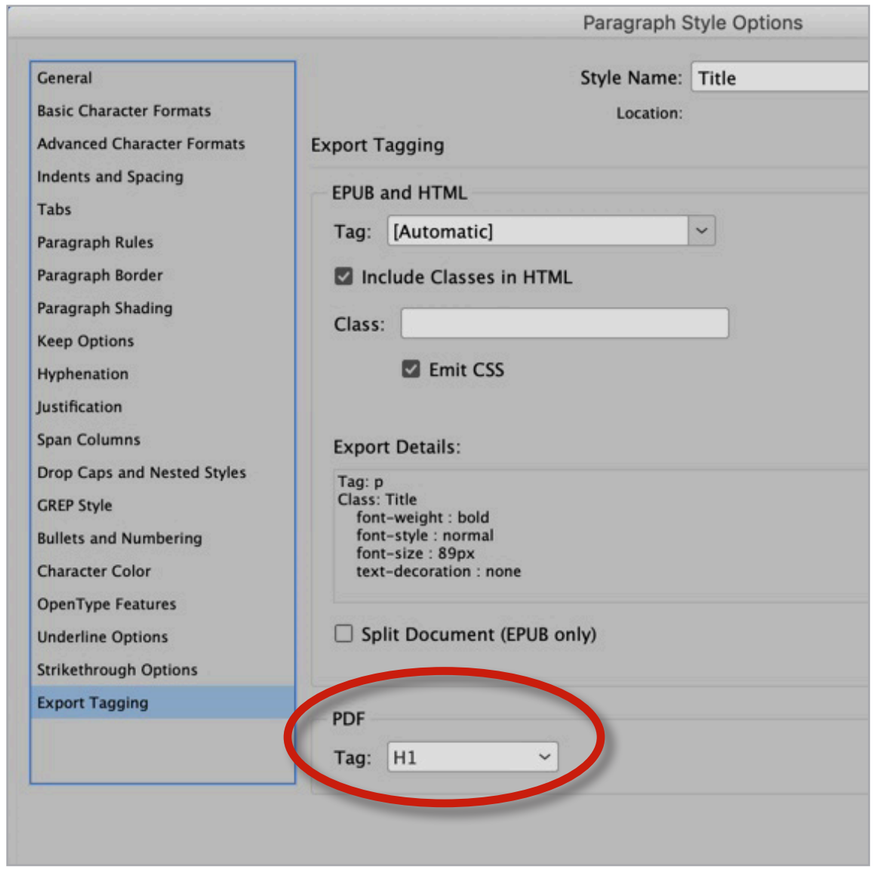
Proper tagging of headings is accomplished first by ensuring



that you've applied paragraph styles to all of the headings in your InDesign document. Once that's done, you can map the appropriate PDF heading level in that style. Start by opening the Paragraph Style Options dialog box for the heading style that you're working with. On the left side of the dialog box, you'll see a category called Export Tagging. Click that category to display the options; at the bottom of the dialog box you'll see a section called PDF. In the Tag drop-down menu, you can

choose the appropriate PDF tag that you'd like applied in the exported PDF file for each heading (**FIGURE 4**).

FIGURE 4. Defining the PDF export tag in the Paragraph Style Options dialog box



Step 6: Apply Alt Text to Images and Graphics

We've all heard many times the old adage that a picture is worth a thousand words, which is true for a sighted user. But pictures are a challenge for accessibility, of course. Because of this, you need to add alternate (or "Alt") text that can be read aloud by assistive software to describe images in context.

I stress the phrase "in context" because it's not enough to say "person standing in a room" for an image. That doesn't provide the reader with much information. A more appropriate alternate text description might be "a woman with long dark hair and glasses, wearing business attire standing at the window in her sun-lit office." I tell people to close their eyes and think about what the alternate text is saying. Based on the alternate text, can you envision what the photograph looks like? If not, you should rewrite it.

As designers of visual projects, it's quite common for us to add graphics and images that add to the visual aesthetics of the project. However, the reality is that sometimes those elements are purely decorative. They only make the project more visually appealing, and there's no value in describing those images to a non-sighted user. Such elements can be "backgrounded" or "artifacted" (Acrobat uses both terms to describe the same thing), which basically means that those elements will not be read by assistive software.

Adding alternate text and artifacting an image are both accomplished in InDesign by selecting the image with the Selection tool and choosing **Object > Object Export Options**. Clicking on the Alt Text button allows you to choose the alternate text source.

Choose Custom to manually enter the alternate text that you'd like applied to the image. Or, you can pick from one of the XMP metadata fields of the image to extract the alternate text from the metadata field of the image to generate alternate text in a more automated fashion (**FIGURE 5**).

Clicking on the Tagged PDF tab at the top of the Object Export Options dialog box provides the option to choose what tag is applied to the selected object from the Apply Tag drop-down menu (**FIGURE 6**). The primary purpose of this section is to apply the Artifact tag when you don't want assistive software to read a particular image. Again, this is usually done when an image doesn't contribute to the reader's understanding of the content.

It's also worth mentioning that objects drawn in InDesign (or copied and pasted into InDesign as paths from another application such as Illustrator) will automatically be artifacted in an exported PDF file.

Step 7: Set the Reading Order

Probably one of the more confusing concepts for beginning PDF remediators is that of reading order. Not because it is difficult to understand per se, but because there are actually two different reading orders to contend with in an accessible PDF file. Furthermore, ironically, the order as shown in Adobe Acrobat's Reading Order panel (and Order Navigation pane) is *not* the most important reading

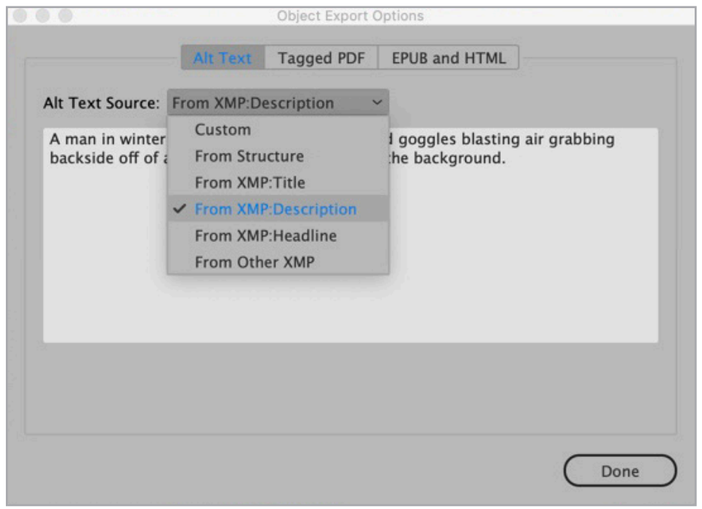


FIGURE 5. Adding alternate text to an image using the Object Export Options dialog box in InDesign

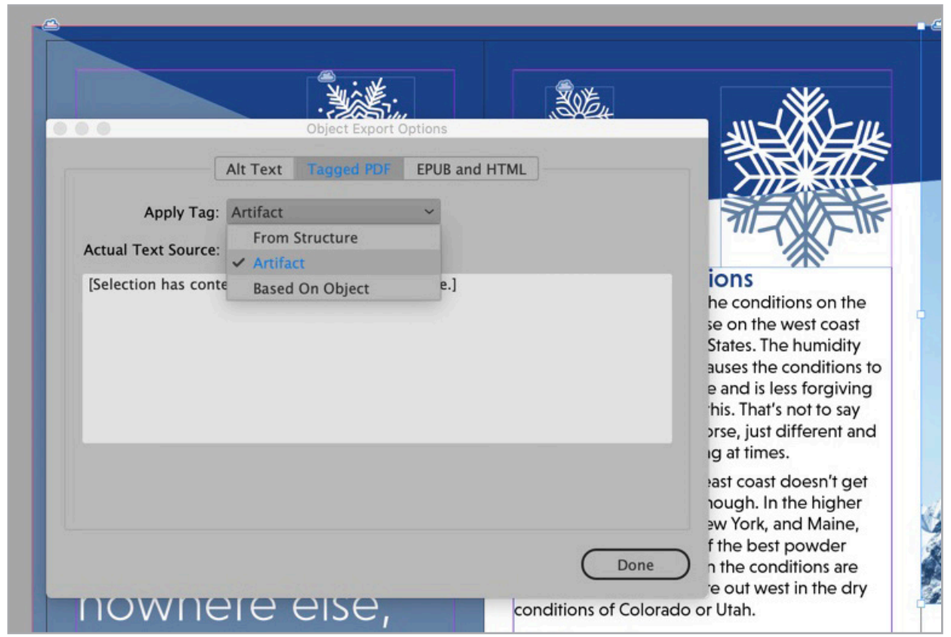


FIGURE 6. Applying the Artifact tag to objects that shouldn't be read by assistive software

order that you need to define. Let me explain.

Assistive software programs like JAWS (Job Access With Speech) or NVDA (Non Visual Desktop Access) take the *tag* order—as listed from top to bottom in the Acrobat Tags pane—as the order in which it reads objects in a document. As you can imagine, this order is incredibly important, since if it's not correct, the content may not make any sense to the user. For example, think about a document such as a tri-fold brochure. When it's laid out flat, you can't read this document left to right as you would most other documents. So, for a document like this, it's critical that you get that tag order correct.

How do you accomplish this in InDesign? By default, the tag order is defined in a linear manner, starting at the top left of each page in the document, moving across each page, and then down. For InDesign documents where this approach is not sufficient (again using a tri-fold brochure as an example), you need to use the Articles panel. There, you systematically determine the order that objects will appear in the Tags pane of the exported PDF file. Begin by selecting the first object in the document and using the Selection tool: drag it and drop it on top of the Articles panel. That object gets added to the Articles panel and becomes the first object that will appear in the Tags pane of the PDF. Continue adding objects to the Articles panel in the order that they should be read in the exported PDF (**FIGURE 7**). When you're done with that, there's one last feature that needs to be enabled in order for the Articles panel to do its job. Go to the panel menu, and choose Use for Reading Order in Tagged PDF (**FIGURE 8**). If you don't enable this option, it's as if you didn't add anything to the Articles panel at all.

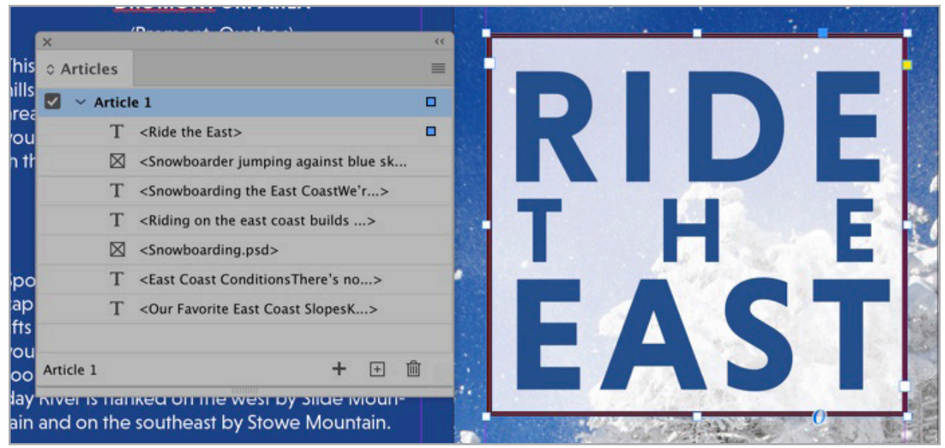


FIGURE 7. Adding objects to the Articles panel in InDesign

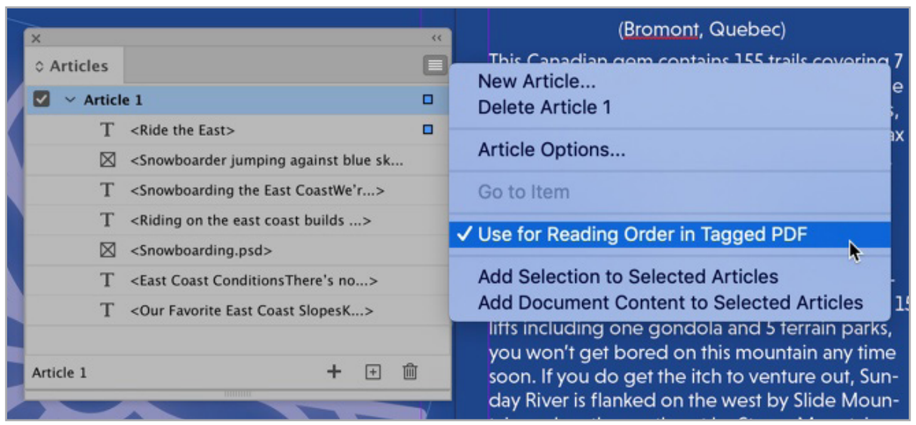


FIGURE 8. Enabling the Use for Reading Order in Tagged PDF in the Articles panel to ensure that the tags export in the proper order in the exported PDF file

However, just because you get the tag order correct doesn't mean you're done. You also need to contend with the Reading Order, as shown in Adobe Acrobat's Reading Order panel and Order navigation pane. I call this the "flow order." It is not used by screen readers such as JAWS and NVDA, but it is used by Adobe's Read Out Loud feature offered in **View > Read Out Loud** or by low-vision readers to reflow the text of a PDF file.

One way to get a sense of how reflow behaves is by opening a PDF file in Acrobat and choosing **View > Zoom > Reflow**. You'll see the content gets taken out of the layout in which it was designed and flows for an easier reading experience (**FIGURE 9**). I liken this to the read mode found in many web browsers.

I use the Reflow feature in Acrobat only to demonstrate the reflow behavior; low-vision users have more robust tools for reflowing the text that do a much better job. Unfortunately, it's nearly impossible to get perfect reading order straight out of InDesign. That's because InDesign uses the stacking order of the objects in the Layers panel to set the reading order of content in an exported PDF. For example, the bottommost object is first, then the object on top of that, and so on. For simple documents, you may be able to get away with dragging the objects up or down in the Layers panel (after clicking the triangle next to the layer name to expand it). But for most documents, this is impractical.

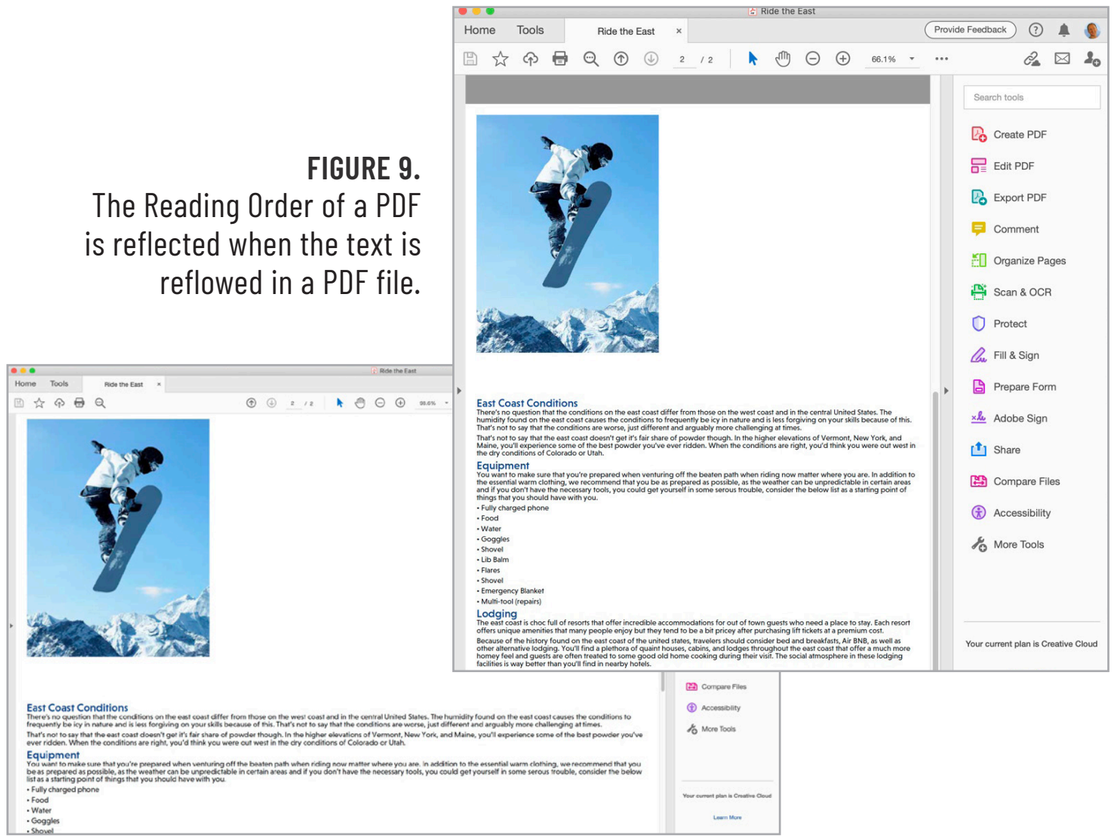


FIGURE 9.
The Reading Order of a PDF is reflected when the text is reflowed in a PDF file.

The result is that you'll almost always need to fix the reading order in Acrobat after the PDF has been exported.

Step 8: Anchor Graphics as Needed

Many designs contain elements that appear in the middle of flowing text. This is often accomplished by positioning elements at appropriate locations and applying text wrap to the object to cause them to push away the text around them. The challenge for us as PDF remediators is to

control where those items are read in the document. At times it may make sense to read the object at that exact location, but at other times you'll need to make a judgment call regarding where (or if) the object will be read. By default, the object will often appear at the bottom of the Acrobat Tags pane when exported to PDF.

When using the Articles panel, objects can only be inserted before or after another object. You can't insert an object in the middle of another object. A text story is a single object. So in order to force an image or graphic to be read at a specific location within a story, you need to anchor the object, or make the object inline within the text. By doing this, you can define exactly where in the tag order the object will be read.

There are challenges with this method, however—specifically, the fact that text wrap cannot affect the line on which the wrapping object sits. So you often need to choose to place the object at the end of the preceding paragraph, as opposed to the beginning of the paragraph where you want the object read.

Step 9: Use the Right PDF Export Settings

If you want to get an accessible PDF out of InDesign, you can't use the Print command. This may sound obvious, but I still run into a lot of users who create PDF files out of InDesign by choosing **File > Print**, and then selecting Save as PDF as their "printer." But using the Print command to create a PDF file from any application will yield a PDF file without any tags at all, so it's a nonstarter for accessibility.

To generate an accessible PDF file, you'll want to choose **File > Export**, and choose one of the PDF options from the format drop-down menu: either PDF (Print) or PDF (Interactive). Both choices allow you to generate an accessible PDF file, but I typically use the Print option unless I'm outputting a PDF that contains form fields, buttons, and other interactive content.

For creating an accessible PDF file, there are a few important options that you'll want to enable. All of the important

options for accessibility are found in the General and Advanced (**FIGURE 10**) sections of the Export Adobe PDF dialog box.

In the General Section:

- View PDF After Exporting
- Create Tagged PDF
- Bookmarks
- Hyperlinks

In the Advanced Section:

- Display Document Title
- Define Language

Step 10: Clean Up in Acrobat

When I'm creating accessible PDF files, I shoot for the 85/15 Rule: I try to do 85% of the accessibility work in InDesign (or any other source application) so that I only need to do the remaining 15% of cleanup in Acrobat. This is very much a general rule of thumb, and depending on the length and complexity of the document I may do better or worse. You'll never get to 100% efficiency because there's always some amount of cleanup to do in Acrobat after generating the PDF file. But try to keep that work to the bare minimum, because if you ever need to regenerate the PDF file for any reason, all of that additional cleanup that was done in Acrobat will need to be done again to the new PDF file.

The amount of cleanup that you'll need to do will also vary depending on the standard that you're shooting for. For the sake of this article, we're going to use the Acrobat Accessibility Checker, which generally checks against the WCAG standard.

To check your document, open the Accessibility tool found under Protect and Standardize in the More Tools menu in Acrobat, and click Accessibility Check to open the Accessibility Checker Options dialog box (**FIGURE 11, NEXT PAGE**).

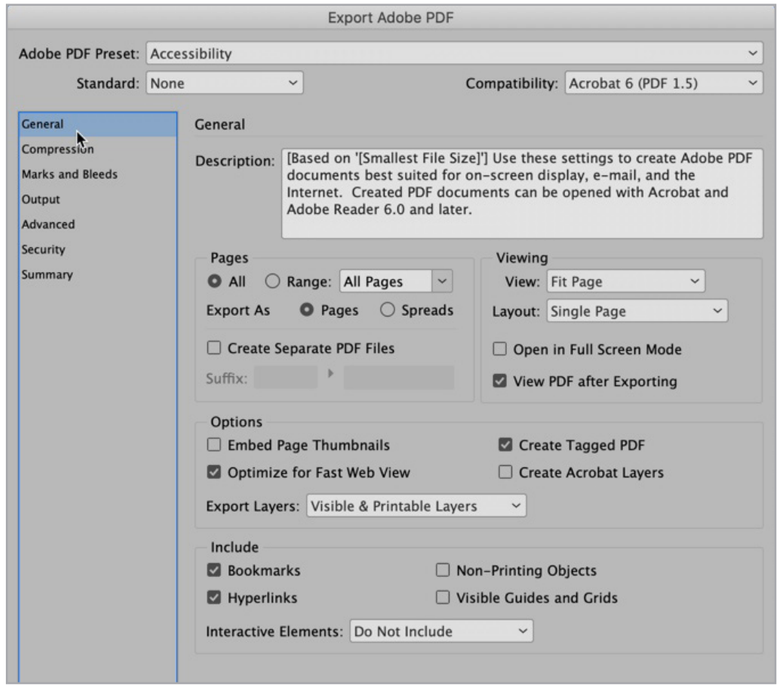
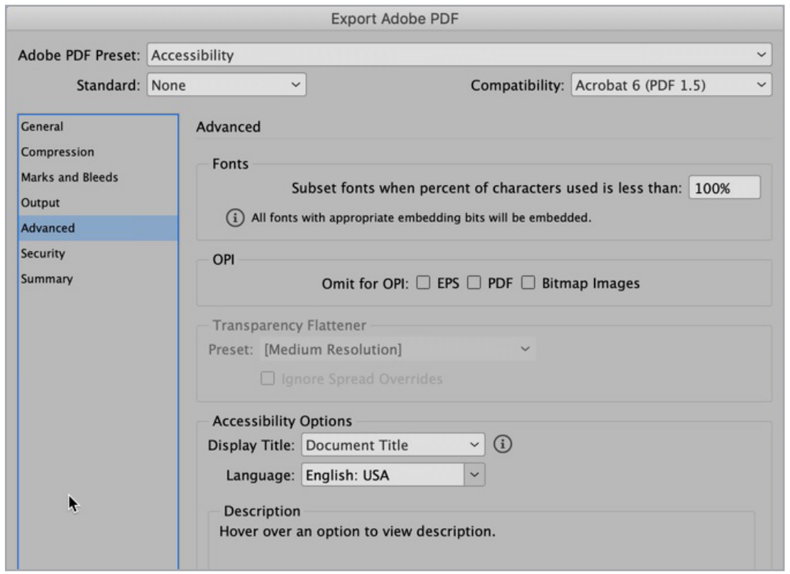


FIGURE 10. Accessibility options are found in the General and Advanced sections of export dialog box.



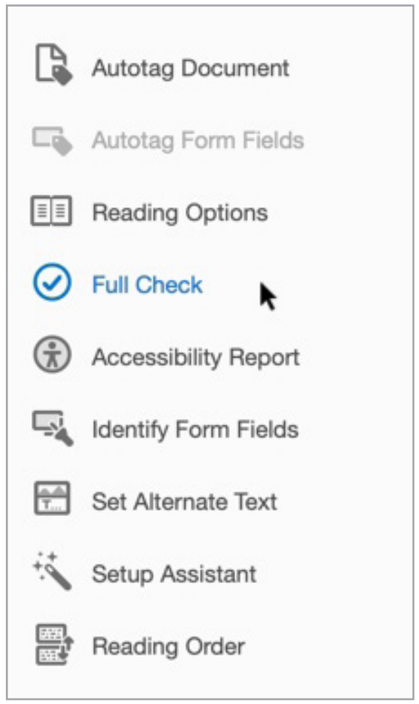


FIGURE 11. Choose Full Check from the Accessibility tool in Acrobat to run the Acrobat Accessibility Checker.

Leave all of the options in the dialog box enabled, and click the Start Checking button (**FIGURE 12**).

Acrobat will run the Accessibility checker and display a report in the navigation pane on the left side of the Acrobat interface. You'll see seven different categories in the report, and any issues that have been found will be shown in the report. The Document section will always show two issues (not errors) that basically tell you that reading order and color contrast can't be checked by Acrobat and need a human being to perform that step.

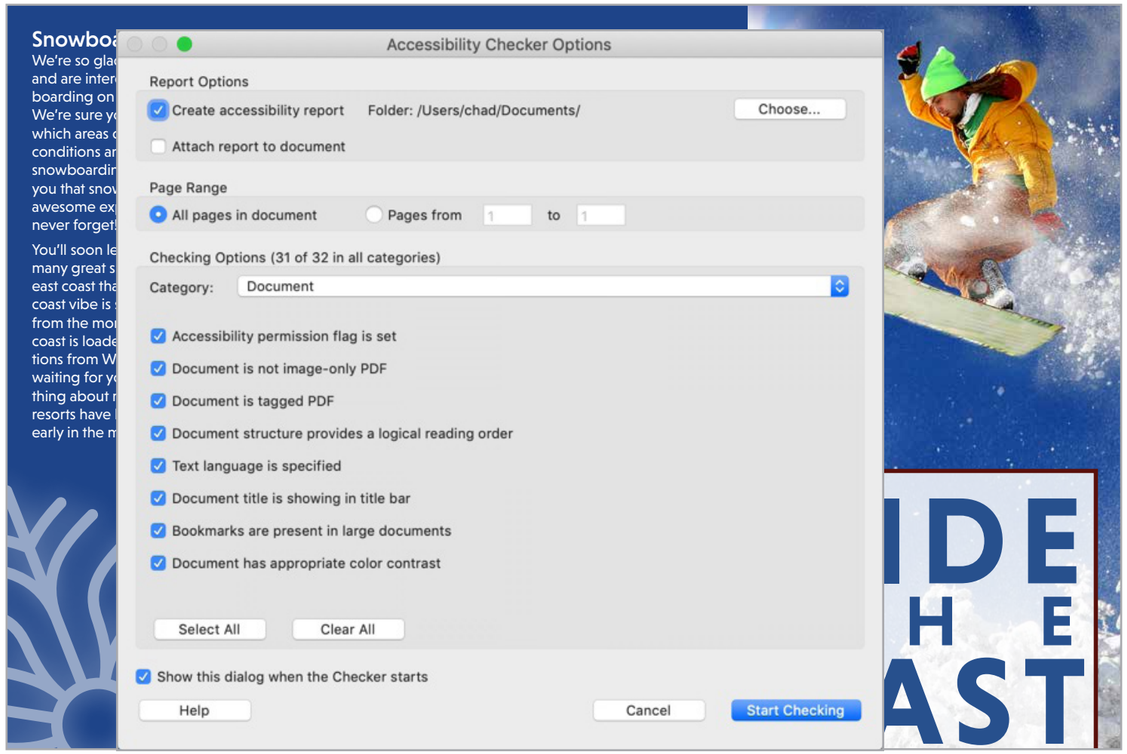


FIGURE 12. The Acrobat Accessibility Checker dialog box

Files exported from InDesign will always present an error for the Tab order. (Well, at least they do at the time of this writing. Perhaps Adobe will fix this in the future.) However, you can simply right-click on that error and choose Fix to remedy the problem. Other errors may require going back to InDesign to make adjustments and re-exporting the PDF file, and applying fixes in Acrobat for things like reading order, as discussed earlier.

Accomplishing Accessibility

The process of creating an accessible PDF file may seem a bit daunting after reading these steps, but in all honesty, most of what needs to be done to create an accessible PDF file comes down to creating clean files in InDesign and paying attention

to detail. As you remediate more files and gain more experience, you'll find yourself instinctively adding the necessary elements into the file that will aid in the overall process.

Furthermore, the founding principal of accessibility is "equal access for all." Ensuring that any user, whether sighted, non-sighted, low-vision, or mobility impaired, will be able to access the content in the file that you create is vitally important. It also shows our commitment as document designers to the idea that all users attempting to read our documents should be able to do so in a reasonable and convenient manner.

Chad Chelius *is a trainer, author, consultant, and Director of Training Solutions and Principal at Chax Training and Consulting. He resides in the Philadelphia area and has been using Adobe products for over 25 years. As an Adobe Certified Instructor, Accessible Document Specialist, and consultant he teaches and advises on all Adobe print and web products, specializing in InDesign and InCopy workflows, Illustrator, automation, and PDF accessibility using InDesign, Word, and Adobe Acrobat. He works with clients both large and small in and outside of the United States, helping them to solve design, workflow, and accessibility challenges using Adobe products.*

Building Accessible EPUBs with InDesign

BY Kevin Callahan

UPDATED
BY Laura Brady

Start with a strong foundation to reach more readers.

ACCESSIBILITY IS ON EVERYONE'S MIND these days and a part of everyone's job. Owing to current and incoming legislation, and to the moral imperative to publish more inclusively, content publishers need to start thinking about accessibility concerns right from typesetting and layout.

If you use InDesign to make print layouts that will become ebooks, this article is for you. InDesign is not designed to be an ebook creation tool. You can make ebooks with it, but you have to really work the InDesign gears to get it to create clean HTML. Left to its defaults, InDesign will export generic markup that will get in the way of accessibility.

An accessible EPUB includes many features that allow people to read what they want, when and where they want. This article lays a foundation for how to best set up InDesign for successful, accessible ebooks.

From Structure to Semantics

When thinking about accessibility, it helps to remember that we're really talking about a machine displaying content. Look at a print book. You turn a page and you recognize design elements that mean you're moving into a new chapter. But a reading device can't recognize those elements in the same way.

An e-reader doesn't glean information from your design or from the text itself. Instead, it has to be told, via special "tags" in the ebook, things like *this is a first level head* (h1), *this is a sidebar* (aside), *this is an illustration* (figure). Using these additional semantics makes your content meaningful to rendering engines and thus to readers with print disabilities.

FIGURE 1 shows an example of what accessible HTML5 semantic markup looks like, with tags inside angle brackets.

In this example, the `<section>` tag tells the machine we're beginning a new chunk

> **"STRUCTURE IS THE ELEMENTS YOU USE TO CRAFT YOUR EPUB CONTENT, AND SEMANTICS IS THE ADDITIONAL MEANING YOU CAN LAYER ON TOP OF THOSE STRUCTURES TO BETTER INDICATE WHAT THEY REPRESENT."**
>
> —Matt Garrish, *Accessible EPUB 3*

FIGURE 1. Structure applied to content with semantic markup

```
<section epub:type="bodymatter chapter">
  <h1>Chapter 1: Title</h1>
      <p>Text in here.</p>
      <section>
          <h2>Subhead</h2>
              <p>More text in here.</p>
                  <aside>
                      <p>Sidebar text.</p>
                  </aside>
      </section>
</section>
```

of text. Then, for clarity, the tag includes `epub:type="bodymatter chapter"` to say it's the start of the main text of the book, and it's a chapter. We read some text, and then a new, nested `<section>` begins. Inside that tag is an `<aside>`, which is like a sidebar.

If you're using the accessibility features on an e-reader, you might want to hear all of the chapter titles read aloud, so you could jump to a specific chapter. That's possible if the heading tag tag is applied only to chapter titles, such as `<h1>`. You can choose to skip sidebars tagged `<aside>` when reading the main text, hear a picture description before reading its caption, and so on. (These features aren't available in all e-readers today, but we have to aim for that possibility.)

How do we get that information into an ebook? Through structure and semantics.

Structure is the general outline of a book, as in `<section>` and `<h1>`. *Semantics* go further, assigning specific attributes to those generic elements: Frontmatter, Bodymatter, Backmatter, etc.

Semantics tells us whether a word is italicized because it's emphasized (`<em>`) or because it's a citation to a book title (`<cite>`), as well as whether a word is bolded for display purposes (`<b>`) or because it's a WARNING (`<strong>`).

The machine reads what we tell it to by our markup. The human touch is evident in every choice we make in bookmaking, even if it's only visible to a machine. And InDesign does a pretty decent job adding this information.

EPUB 2 OR EPUB 3?

Simply put, there is no reason nowadays to create an EPUB 2 file. Exporting to EPUB 2 would be like making a black-and-white TV. Sure, there are black-and-white shows and movies, but most are in color. And color sets will show the black-and-white material without any problem. And, it's important to note that EPUB 3 is more accessible by nature and by design. Every major reading system can display EPUB 3 files. EPUB 3 includes expanded navigation in the toc.xhtml, richer content tagging with epub:type, and stronger support for scripting, audio, and video. For older devices that can't read EPUB 3 features—like the TOC—InDesign includes fallbacks, like the EPUB 2-specific toc.ncx.

Mapping Styles

How does InDesign lay the foundation for an accessible EPUB? The answer lies with style mapping.

Just as you do for Word styles when importing text into InDesign, you *map* InDesign's paragraph, character, and object styles to HTML tags when you export to EPUB. For example, you can map the paragraph style Chapter_Title to `<h1>` and the basic Text paragraph style to `<p>`.

The quickest way to do this is to access the Edit All Export Tags command from any of the style menus. In this single dialog box, you can edit all the styles used in a file and enter corresponding HTML tags and CSS classes. (**FIGURE 2**).

If you let InDesign create the CSS document, it will create classes with names based on the paragraph, character, or object style names. If you develop your own class names, you can plug them into the Class column, and InDesign will use them. Being consistent about style names will make editing easier.

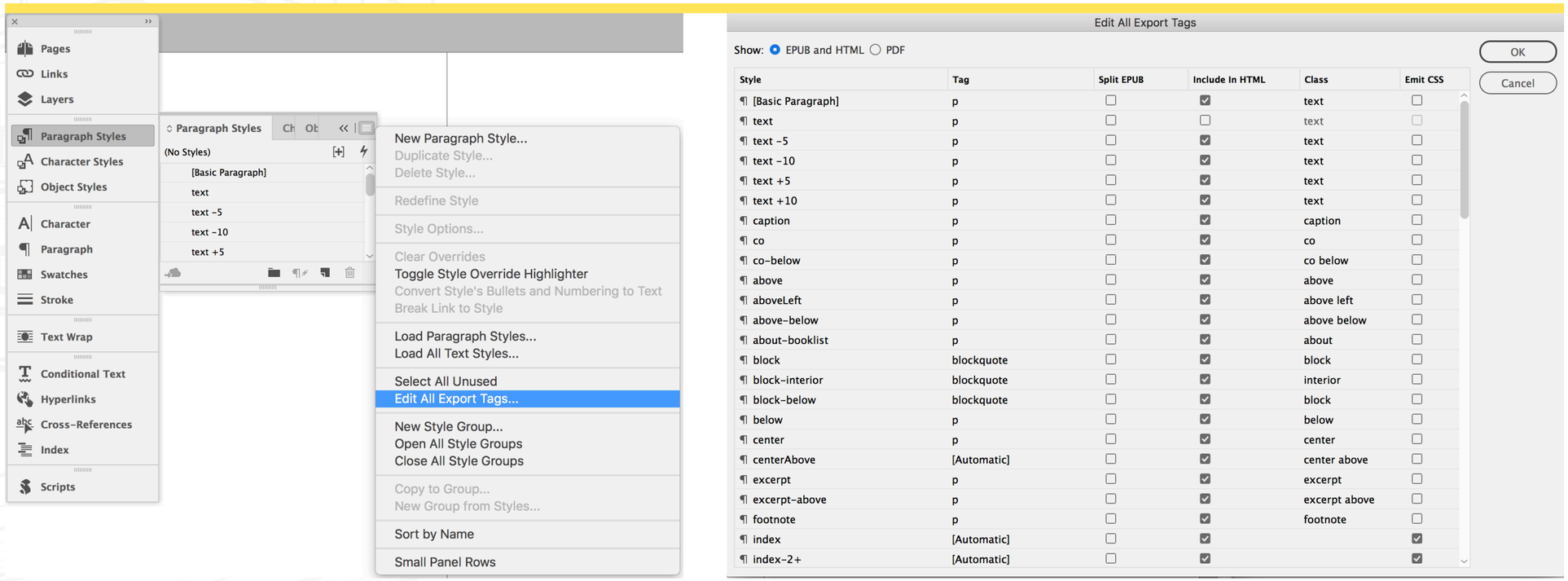


FIGURE 2. Access Edit All Export Tags from the panel menu in the top-right corner of any of the style panels (left). In the dialog box, you can enter CSS classes (right).

TIP: You can also designate a tag and a class from within an individual style's Style Options dialog box. If you do that, it's a good idea to double check in the Edit All Export Tags dialog box. In older versions of InDesign, the tags sometimes disappear from the style definition after you've clicked OK.

> "STYLE IS JUST A LAYER BETWEEN YOUR MARKUP AND THE DEVICE THAT RENDERS IT, NOT AN INTRINSIC QUALITY YOU CAN RELY ON TO SAY ANYTHING ABOUT YOUR CONTENT."
>
> —Matt Garrish, *Accessible EPUB 3*

Custom tags

Here's where real InDesign magic can happen. You don't have to stick with InDesign's suggested tags that appear in the Tag field's menu, like h1, h2, and so on. You can type in your own! For example, you can use object styles to create sections in your document. So, when you create an object style for an image, define its tag as `<figure>` (just enter the word *figure* in the field), which is the correct HTML5 tag for images. The same goes for a sidebar: Give it an object style called *sidebar* and define its export tag as `<aside>` (**FIGURE 3**).

Edit All Export Tags is the source of your EPUB super powers. Leverage these style mappings as much as possible. Once you've done that, you're closer to a semantically correct EPUB file.

FIGURE 3. Note how object styles have tags assigned. The "image" style is a figure in tag-speak, sidebar is aside, and so on. Also, each has its own class, so they will be easy to manipulate in the EPUB.

Edit All Export Tags

Show: EPUB and HTML / PDF

OK

Cancel

Style	Tag	Split EPUB	Include In HTML	Class	Emit CSS
A nobreak	span		☑	delete	☐
A super	sup		☑		☐
A table-number	span		☑	figure-number	☐
A ellipse	[Automatic]		☑		☑
A languages:Czech	[Automatic]		☑		☑
A languages:French	span		☑	french	☐
A com.rorohiko.pagestaker.style	[Automatic]		☑		☑
[Basic Graphics Frame]	[Automatic]		☑		☑
[Basic Text Frame]	[Automatic]		☑		☑
chapter	section		☑	chapter	☐
image	figure		☑	image	☐
sidebar	**aside**		☑	**sidebar**	☐

Keep Content and Design Separate

We've talked about structure and semantics. So where do design choices come in? The fact is that the aesthetics of a book—the typeface, text alignment, color—are meaningless for accessibility. But some design choices can make a book inaccessible. Suppose, for example, you color a table's row blue to convey meaning—common enough in print, but problematic for an EPUB.

E-readers don't derive any meaning from formatting like size and color. E-ink devices display only shades of gray. Text-to-speech software can't interpret color, and users with color blindness won't see the blue table row on a color tablet. So it's important when adapting a print project for digital use to keep an eye out for design elements that may need rethinking.

Using Object Styles for Semantic Markup

InDesign is "div" happy. It puts `<div>` elements around most everything you export to EPUB. **FIGURE 4** shows sample markup from an EPUB export. Unfortunately, the `<div>` tag has no semantic meaning. Because it doesn't mean anything, text-to-speech software won't know how it should render the content within the `<div>`. Instead, you should use `<section>`, the HTML5 tag. Earlier, we looked at how `<section>` and `epub:type` are used. You can assign both from within InDesign.

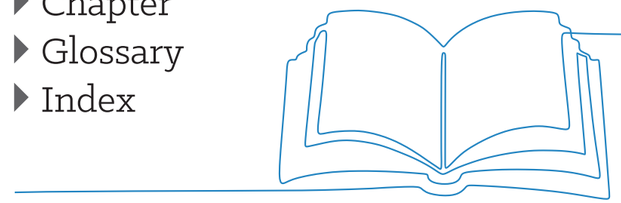
First, to assign the `<section>` tag, simply create an object style, and, in the Export Tagging area of the Style Options dialog box, enter *section* in the Tag field (**FIGURE 5**). Apply this object style to your text frame.

FIGURE 4. Typical code exported from InDesign, with a semantically meaningless element

```
<div id="_idContainer006">
    <h1>Chapter 1</h1>
        <p>…</p>
</div>
```

Next, consider `epub:type`. These designations tell reading software what part of the book you're looking at. Some example are:

- Cover
- Preface
- Chapter
- Glossary
- Index



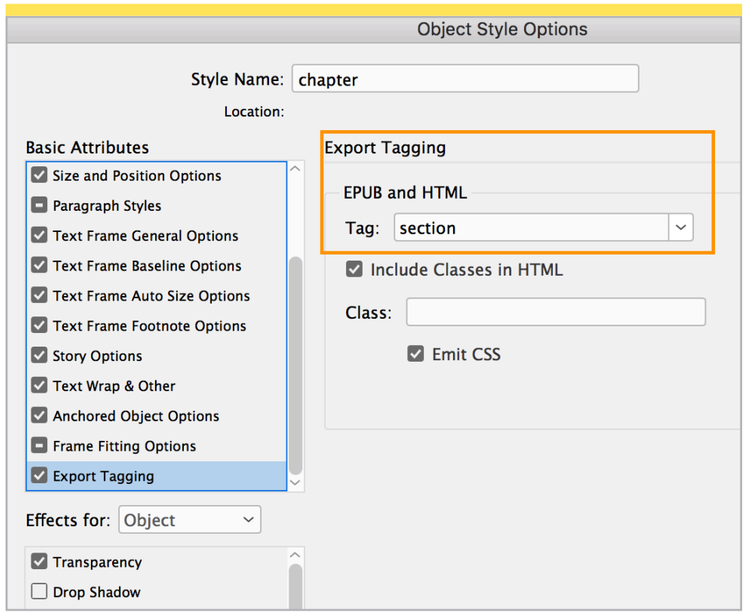


FIGURE 5. Create proper markup here in the Object Style Options dialog box.

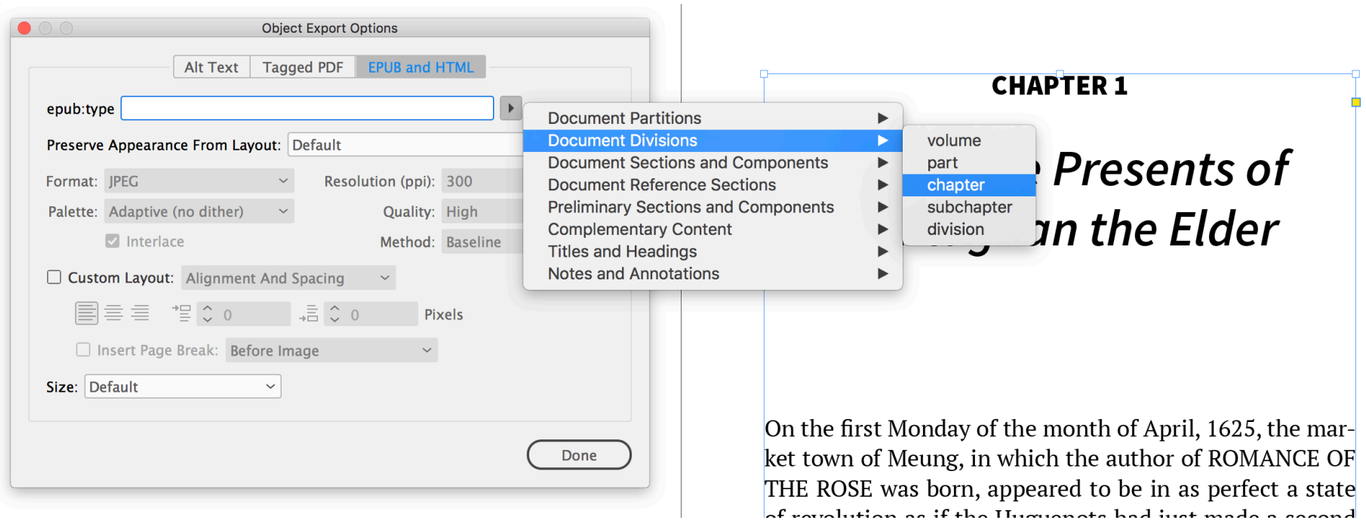
See the "Links and Resources" sidebar for links to more information on `epub:type` definitions. You can assign these to a text frame in the Object Export Options dialog box, and InDesign will include that markup in the exported EPUB (**FIGURE 6**).

Split the story to assign different object styles

There is one problem with applying tags via object styles: It only works on the first text frame of a story. If the dedication and preface are part of the same story, even if they're on different pages in the print edition, you cannot assign `epub:type` to both text frames; you'll need to break that thread.

InDesign has a couple of built-in scripts that can help. BreakFrame and SplitStory interrupt a thread and isolate text frames; unfortunately, they don't leave the rest of the thread intact. But you can use another script, StorySplitter, which interrupts a long thread, leaving styles and positioning intact on both sides of the break. You can download this script and read about similar ones in this post at CreativePro.

FIGURE 6. When you assign epub:type in the Object Export Options dialog box (top), InDesign will include that information in the code (bottom). Note section and epub:type:chapter. Note also id="_idContainer006". This is often a meaningless identification tag InDesign assigns. You can strip them out, but be careful: They are sometimes destinations for toc.xhtml listings.




```
1  <?xml version="1.0" encoding="UTF-8" standalone="no"?>
2  <!DOCTYPE html>
3  <html xmlns="http://www.w3.org/1999/xhtml" xmlns:epub="http://www.idpf.org/2007/ops">
4      <head>
5          <title>chapter 1</title>
6      </head>
7      <body id="chapter_1" lang="en-US" xml:lang="en-US">
8          <section id="_idContainer006" epub:type="chapter">
9              <h2 class="cn">Chapter 1</h2>
```

With StorySplitter, you go through a book that's threaded from front to back and you break it into chunks, such as by chapter. To each chunk, apply an object style that maps to `<section>` and the appropriate `epub:type`. (Note: When you activate the script, it sometimes jumps you to the last page of the thread; it's not an error. Just choose **Layout > Go Back** to return to the page you started at and continue.)

Navigation

Accessible EPUB3 includes rich navigation tools that give readers several ways into the text. They include the reading application's table of contents (usually found in its menu); lists of illustrations, tables, and landmarks; and a page list. **FIGURE 7, NEXT PAGE,** shows the navigation panel from Adobe Digital Editions with contents, landmarks, and page list.

The toc.xhtml is more straightforward than EPUB2's toc.ncx. All listings are simple ordered lists; there are none of the annoying navpoint and playorder numbers to keep track of. You should still include the toc.ncx

ARIA

The epub:type property is specific to ebook accessibility and, unfortunately, is essentially meaningless. The parallel web-based specification is ARIA: Accessible Rich Internet Application. ARIA has its own set of roles that can be placed alongside epub:type designations to future-proof content as ebook standards evolve. For example:

```
<section role="doc-chapter" epub:type="chapter">
    <h1>Chapter 1: Title</h1>
        <p>…</p>
</section>
```

You need to put these into place once your EPUB is exported, because InDesign doesn't include ARIA (yet) in its exported markup. Consider running global Find/Change routines after exporting the EPUB to add ARIA roles.

Find: `<section epub:type="chapter">`

Change to: `<section role="doc-chapter" epub:type="chapter">`

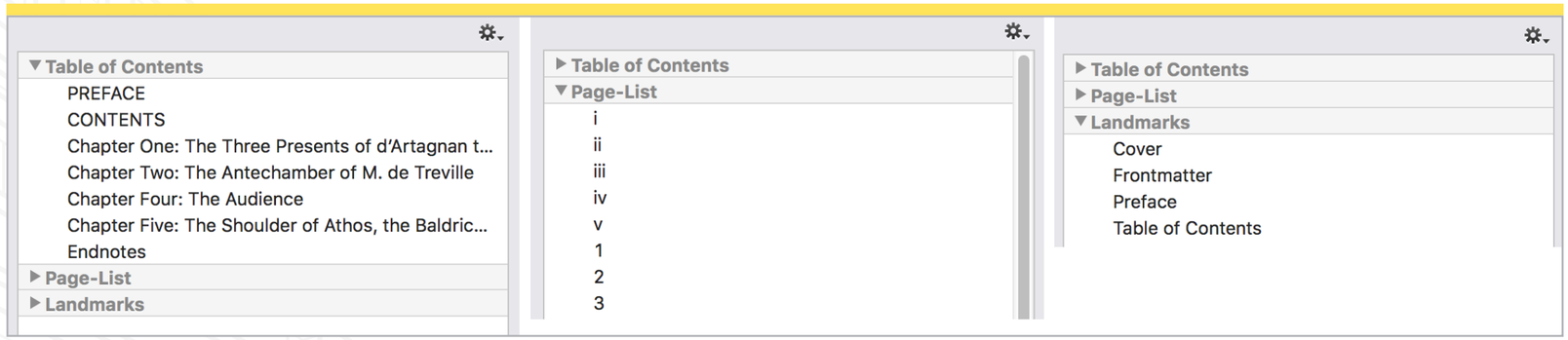


FIGURE 7. The navigation panel from Adobe Digital Editions, showing (from left) the TOC, Page-List, and Landmarks

in your EPUB for older devices; InDesign exports it for you.

Create the toc.xhtml using InDesign's Table of Contents (TOC) feature (**Layout > Table of Contents**). You don't need to insert the TOC in your book—just save your settings as a TOC style. Then, when you export to EPUB 3, choose Multi Level (TOC Style) and your style for the type of Navigation TOC, in the General section (**FIGURE 8**). This populates the e-reader's Contents menu.

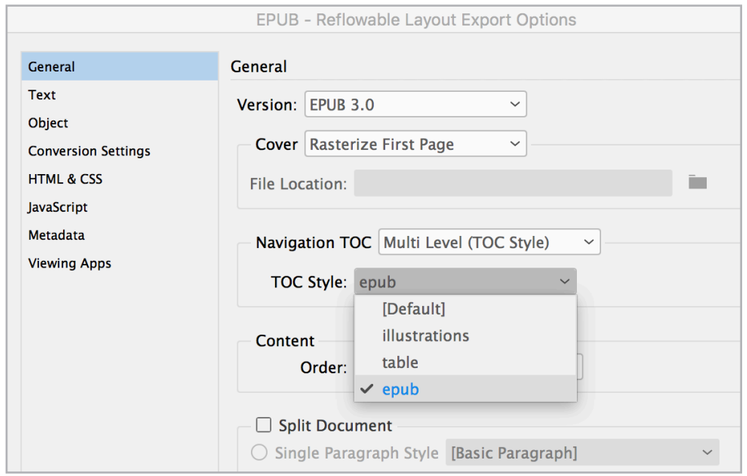
If you've added `epub:type` tags to parts of your book, like Frontmatter and Preface, InDesign will add them to the Landmarks in the TOC.

Page-List and listings of illustrations and tables add to accessibility by providing information about the contents to readers in various situations. Landmarks can let a reader using a small screen quickly find the index instead of scrolling through many screens or slogging through a long table of contents.

Logical reading order

When reading a book that has tables, footnotes, images, or sidebars, you usually read the text first, and then go to the sidebar or table for additional information. The HTML5 `<figure>` (for images and tables) and `<aside>` (sidebars and footnotes) tags signify elements that are not part of the main text flow. Someone using assistive technology may be able to delay reading

FIGURE 8. Choose the TOC style you want. InDesign creates the toc.xhtml from this.



anything marked <aside> or <figure>, depending on the software they're using.

Images

Inserting images involves applying structure and including alt text (alternative text). One is simple to address; the other may require a team effort.

Structure

FIGURE 9 shows the basic structure for an image with a caption in HTML5. If only InDesign would export like this! You'll have to do a combination of style mapping and post-export Find/Change to accomplish it. Here are the steps:

1. Assign an object style to your image, and map it to generate the <figure> tag (**FIGURE 9**).
2. Assign a paragraph style to your caption and map it to <figcaption> (**FIGURE 10**).
3. Anchor your image in position, either before or after the caption.
4. When you export, you'll see something like the markup in **FIGURE 11**.

FIGURE 9. HTML5 <figure> element representing an image

```
<figure>
  <img src="photo.jpg"
   alt="sunny day and big
   ocean waves" />
    <figcaption>John and Mary
    on the beach.</figcaption>
</figure>
```

FIGURE 10. Map your caption paragraph style to <figcaption> (top) and your image object style to <figure> (bottom).

¶ caption	**figcaption**	☐	☐	**caption**	☐
image	**figure**		☑	**image**	☐

FIGURE 11. Notice all the tags. As we've described, they don't have any semantic meaning. You can leave them in place or remove them. You do have to move the </figure> from line 21 to after line 23, though, so it closes after the </figcaption> (line 24). Note that if InDesign creates the CSS, it may be using some of those classes inside the tags to size or position the image. So be careful when deleting them; see if the class is in your exported CSS.

```
16 ▼  <div class="_idGenObjectLayout-2">
17 ▼      <div id="_idContainer009" class="Basic-Text-Frame">
18 ▼          <div class="_idGenObjectLayout-1">
19 ▼              <figure id="_idContainer008" class="image">
20                    <img class="_idGenObjectAttribute-1" src="image/IMG_7098.png" alt="
                      path leading to the castle keep" />
21                </figure>
22            </div>
23            <figcaption><a id="_idTextAnchor003"></a>Castle of the d&amp;apos;Artagnans.
              </figcaption>
24        </div>
25    </div>
26
```

Alt text

Alternative text sits inside the `<img>` tag and is read aloud when text-to-speech reading is enabled. It should provide information not found in the caption or surrounding text.

For example, the caption for a photo of a couple at the seaside might mention John and Mary on the beach, while the alt text describes the weather and surf conditions.

If descriptions in the caption and elsewhere in the text are enough to convey the meaning of the image, you can artifact the image in InDesign which will leave the Alt tag empty: `alt =""` in the HTML. Or, you can reference the text discussion: If the text mentions Figure 1, and the caption describes it in full, just put `"Figure 1"` in the alt text so text-to-speech listeners will know that they're moving past the image.

How to add alt text

You can add alt text in a couple of places. In the Object Export Options dialog box, you can select Custom from the Alt Text Source menu and enter the text (**FIGURE 12**) or select From Structure, which pulls from the image's metadata if it has any.

TOOLS FOR EDITING EPUBS AFTER EXPORT

Use a tool like ePub Zip/Unzip or eCan-Crusher to open an EPUB, which unzips the file and makes all parts available for editing in text editors like Dreamweaver, BB Edit, or Notepad++. Editors like Oxygen or Sigil allow editing from within the unzipped EPUB. It also validates your EPUB as you're working.

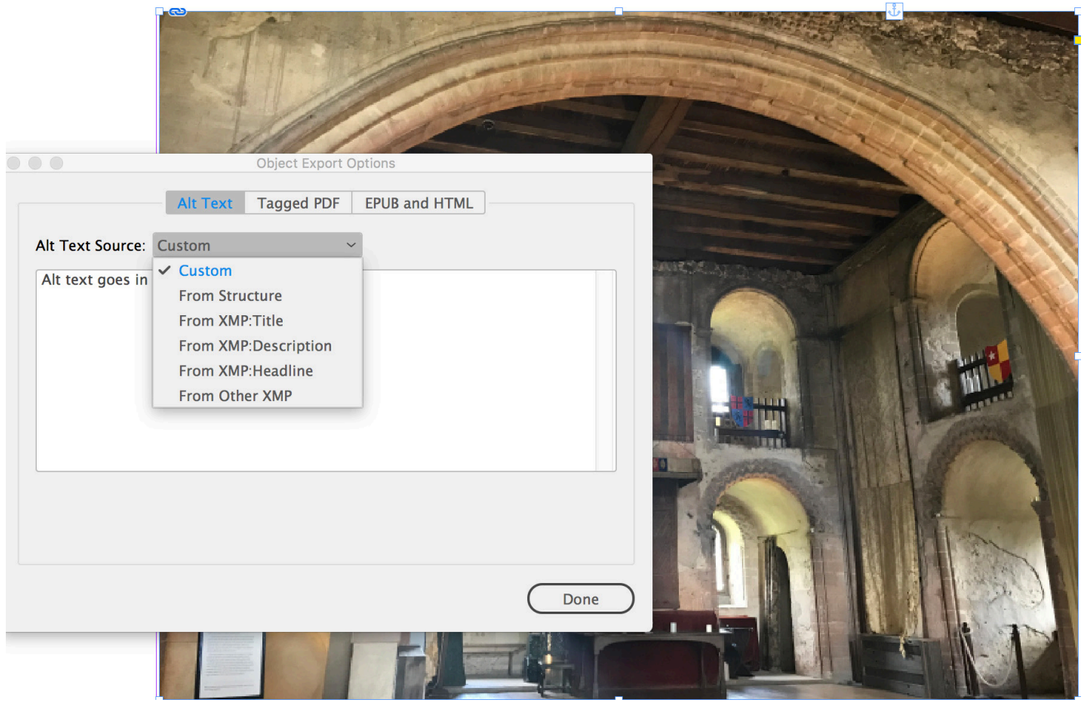


FIGURE 12. Enter your own custom Alt Text here, or select another choice if your workflow allows.

You can also add the Alt Text in the HTML after export; just type it between the quote marks: `alt="sunny day and big ocean waves"`.

Tables

Tables have always been problematic in ebooks. Multi-column, complicated material doesn't fit on a small smartphone screen. Some reading systems have trouble parsing the code. This led developers to present complex tables as images. But that has problems, too.

Any text inserted as an image is not accessible. Text-to-speech readers can't get inside an image. You can add alt text to describe the table, but that probably won't work for complicated tables. So what's to be done? There are some solutions.

- Rewrite simple tables so they are flowing text. Turn column headers into subheads, for example.
- For larger tables, use InDesign's table tools. These will export to EPUB with the necessary HTML markup, especially if you map InDesign's styles to HTML tags. You'll need to go in and clean it up after export, but the exported markup is usable.
- Be careful to designate header rows as header rows. The will create a `<thead>` line which will get read accurately with the columns items by TTS technology.

FIGURE 13 shows the basic accessible HTML5 markup for a table.

FIGURE 13. HTML5 table code

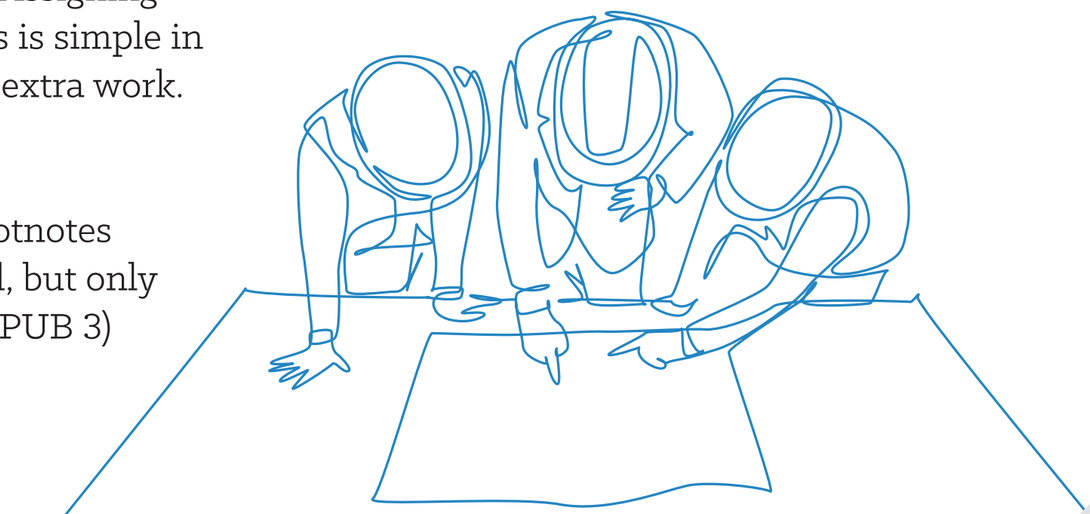
```
<table>
  <caption>What I Need for Winter Weather</caption>
      <thead><tr>
          <th>Clothes</th>
          <th>Tools</th>
          <th>Household</th>
      </tr></thead>
      <tr>
          <td>Boots</td>
          <td>Ice Scraper</td>
          <td>Blanket</td>
      </tr>
</table>
```

Sidebars, Footnotes, and Endnotes

Like images and tables, sidebars, footnotes, and endnotes often give extra information that can be read after the main text. That's why they take the `<aside>` tag. Assigning this tag to footnotes and sidebars is simple in InDesign. Endnotes need a little extra work.

Footnotes

InDesign assigns `<aside>` to footnotes created with the Footnotes panel, but only if you choose Inside a Pop-up (EPUB 3)



from the Footnotes Placement menu in the EPUB Reflowable Layout Export Options dialog box (**FIGURE 14**). Otherwise, you'll have to add your own `<aside>` tags manually, by editing the ebook afterwards.

Sidebars

So long as the sidebar is in a non-threaded text box, simply assign it an object style with the export style mapped to `<aside>`. That's it (**FIGURE 15**).

Endnotes

When exported, endnotes don't take on any of the necessary semantics. As for footnotes, you'll need to take endnotes as `<aside>` and ensure they carry an `epub:type` of `rearnote`. The text reference needs an `epub:type="noteref"`. After exporting, you can use Find/Change to adjust the HTML as shown in **FIGURE 16, NEXT PAGE**.

Lists

Numbered and bulleted lists have their own markup in HTML5. InDesign exports those for you if you've set them up as lists.

FIGURE 14. The EPUB Reflowable Layout Export Options dialog box showing Inside a Pop-up (left) and the exported HTML showing the <aside> tag, along with the epub:type markup (right)

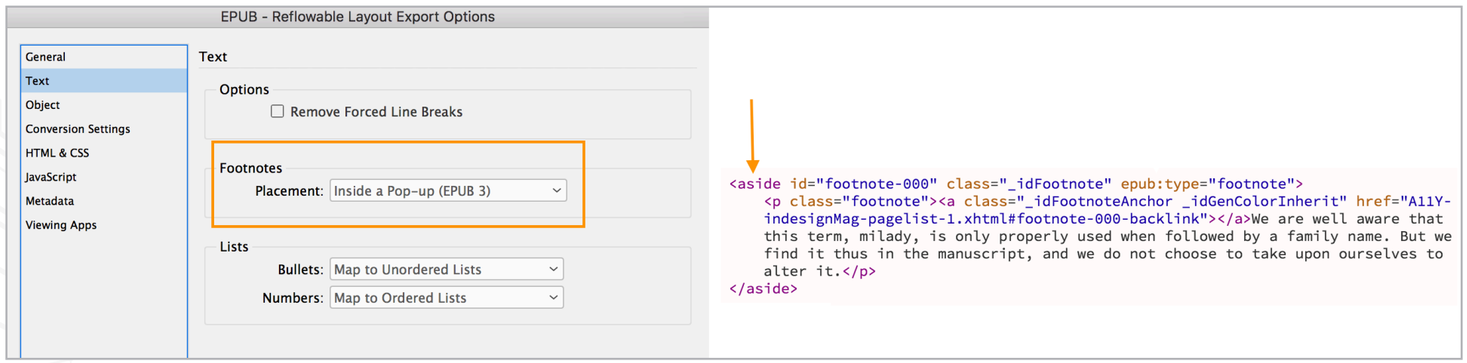
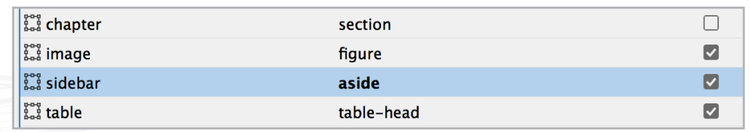


FIGURE 15. Assign an unthreaded text frame containing the sidebar an object style that will export as an <aside> (right) and map the sidebar object style to export as <aside> (bottom).



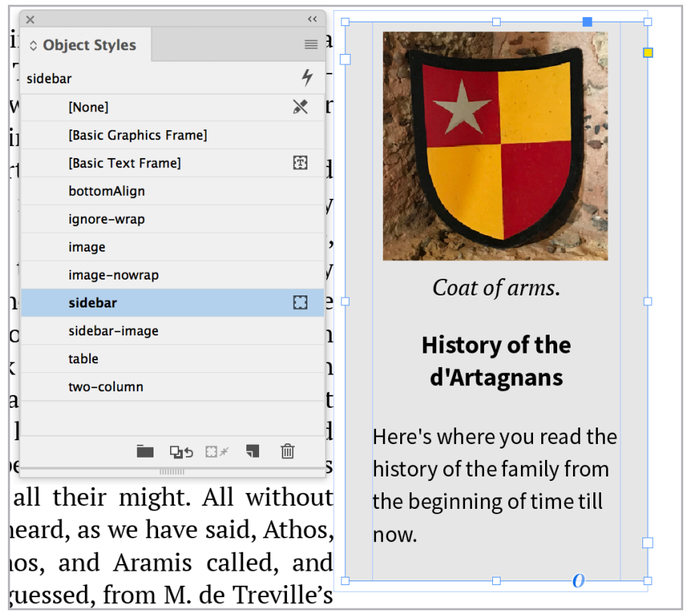


FIGURE 16. HTML5 code for an endnote

```
<p>Body text with endnote reference.
   <a epub:type="noteref" href="#en01">1</a></p>
  <aside id="en01" epub:type="rearnote">
    <p>Endnote text.</p>
  </aside>
```

In the HTML, a numbered list becomes an ordered list (`<ol>`), and a bulleted list is an unordered list (`<ul>`). When you export, you choose how to map them (**FIGURE 17**).

It is important to maintain lists as lists in the HTML, because assistive technology alerts the reader that they have encountered a new list item. Converting those to paragraphs will get in the way of the text-to-speech reading experience.

Language Shifts

Marking a change from the dominant language in an ebook helps a reading system pronounce the word correctly. If your language is set to English: Canadian and your book includes some words in French, apply a character style to the French words.

First, create a new character style (call it *french*). The only attribute you need to apply is in Advanced Character Formats. Set the Language option to French (**FIGURE 18**).

FIGURE 17. The default settings for converting lists as you export

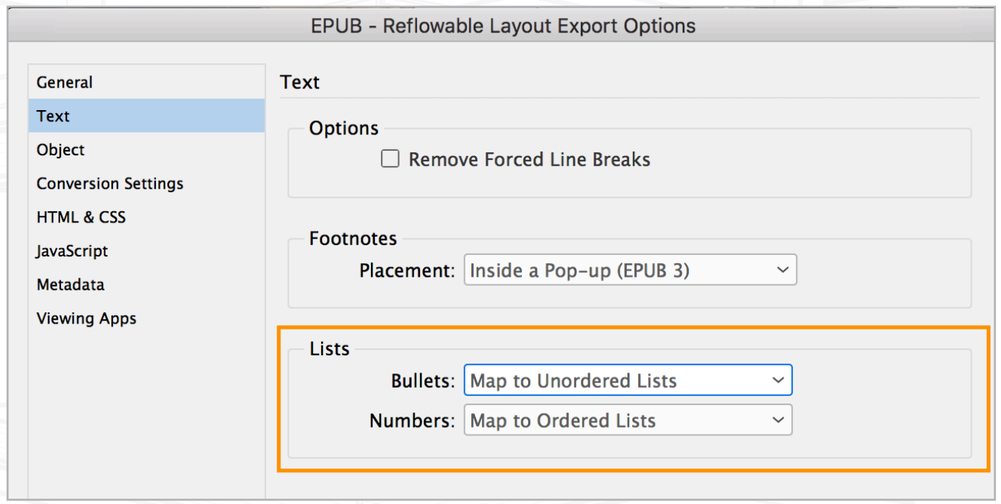
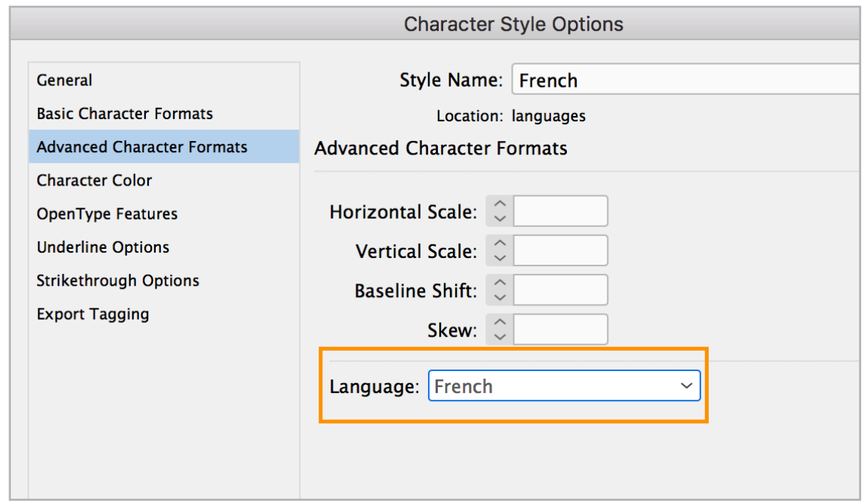


FIGURE 18. Define language shift in a character style. InDesign will wrap it in the correct markup.



On export, InDesign will wrap the French-styled words in a that includes the language attribute (**FIGURE 19**).

You might have to rely on your editor to provide a list of words or support them to do the tagging in the manuscript.

FIGURE 19. The includes the French language designation.

```
<p class="co">On the first Monday of the month of April, 1625, the market town of Meung, in which the
author of ROMANCE OF THE ROSE was born, appeared to be in as perfect a state of revolution as if the
Huguenots had just made a second <span class="french" lang="fr-FR">La Rochelle</span> of it. Many
citizens, seeing the women flying toward the High Street, leaving their children crying at the open
doors, hastened to don the cuirass, and supporting their somewhat uncertain courage with a musket or a
partisan, directed their steps toward the hostelry of the Jolly Miller, before which was gathered,
increasing every minute, a compact group, vociferous and full of curiosity.</p>
```

Color and Contrast

Some readers have difficulty distinguishing colors when there's not enough contrast between foreground and background. Think about type treatments where thin pale blue text sits over a medium-blue band. Combinations that work well in print may be illegible in ebooks, particularly in e-ink (grayscale) e-readers.

This is an issue if you convert your type treatment to a JPEG or PNG, for a highly designed title page, say. If the text is a very thin sans-serif rendered in 40% black, for example, chances are it will be illegible on an e-ink screen.

Test your ebooks on e-ink devices, and create an alternative color scheme if needed. Take values directly from InDesign, and input to your CSS as needed. See the "Links and Resources" sidebar for links to online color checkers, where you can see how accessible your combinations will be.

Emphasis

All italics are not , nor are all bolds . InDesign provides only those choices when you're mapping styles to CSS. But remember: You can input anything you want in the mapping panel. Ask your editor or author to help assign the correct tag.

Italic

There are three types of italic in accessible markup.

- That bump is *huge!* (for emphasis)
- *Voila!* That's how they serve them on board the *Titanic*. (<i> for foreign words, proper names)
- I read all about ocean-going desserts in *Cruise Food: A Guide to Onboard Cuisine*. (<cite> for titles)

Bold

- **Desserts on your cruise.** Ice cream cones are favorite desserts. (Use `<b>`, because the bold adds no meaning.)
- **WARNING!** Eating ice cream too fast makes you want more! (Use `<strong>` when the bold needs to be emphasized.)

Accessibility Testing

Adding the features described here will help bring your EPUBs to a baseline of accessible content, but there's lots more you can do. Since there are so many layers, it's helpful to have an evaluation.

Ace by DAISY Accessibility Checker

Ace by DAISY has quickly become an indispensable tool for ebooks developers. It will analyze your ebook, pointing out places where you might have missed semantics or an image description, in addition to pointing you to resources to help explain what is missing.

Global Certified Accessible

Benetech—a pioneer in making ebooks accessible—recently launched Global Certified Accessible. This program provides publishers with detailed feedback on their EPUB 3 files, with suggestions for improvements.

Reference Tools

Check out the two sites below for more information, code samples, and explainers on the bits and pieces of accessible publishing.

The Daisy Knowledge Base

The Daisy Knowledge Base is a high-level knowledge repository for all things accessible EPUB. It is dynamic, nicely indexed, and very thorough. It has code samples that developers can borrow, and links out to more detailed information in the spec.

Accessible Publishing Learning Network

APLN.ca is a different kind of knowledge base. It aims to be a plain language repository on things related to accessible publishing in general: ebooks, of course, but also

audiobooks, websites, digital marketing, image descriptions, and metadata.

A Rewarding Effort

Maybe someday InDesign will be smart enough that making accessible ebooks will be nearly automatic. But in the meantime, use InDesign thoughtfully and carefully will help you produce an ebook that everyone can enjoy. Taking the time to understand and implement accessibility best practices pays off richly in the end. By following the steps outlined in this article you can produce well-structured content that will be easier to publish in any electronic format or media that may come down the pike.

Kevin Callahan *is an ebook developer, writer, and speaker based in New York City. He has presented sessions at PePcon, the InDesign Conference, Digital Book World, and ebookcraft. You can find him at BNGOBooks.com.*

Laura Brady *is an ebook developer and accessible publishing consultant. She runs APLN, is on the board of the Accessible Books Consortium, and is a general accessibility busybody. You can connect with her on LinkedIn, Facebook, and Twitter.*

LINKS AND RESOURCES

Here's where you can find out more information about the tools and techniques mentioned in this article.

EPUB Secrets: Operated by the publishers of *CreativePro Magazine* and edited by Laura Brady, epubsecrets.com is an invaluable resource for all things EPUB, including accessibility.

For a list of things that should be cleaned up in an InDesign EPUB Export, see this Accessible Publishing Learning Network post.

Other Resources:

- ACE by DAISY Accessibility Checker
- Accessible EPUB 3, by Matt Garrish
- ARIA
- Benetech
- BISG Quick Start Guide to Accessible Publishing
- Colour Contrast Check
- Contrast Ratio
- IDPF EPUB accessibility standards
- ePub Zip/Unzip
- Epub:type guidelines
- FlightDeck
- Oxygen
- StorySplitter script
- PageStaker script (for creating a PageList)

Some of the scripts mentioned in this article are free, and some cost a little. Please consider donating to the free scripters if you use their scripts.

Third-Party Accessibility Tools and Services

BY Chad Chelius

Filling in the gaps in an accessible document workflow

Third-Party Accessibility Tools and Services

Anyone who builds accessible documents for a living will tell you that the process is far from straightforward. The part you do in InDesign is like pouring the foundation of a house: It's a critical first step, but there's a lot more work ahead (in the form of what's called "PDF remediation") before the job is done. For this reason, third-party tools and services are available to ease the process.

Full disclosure: Some of the products I'll be talking about in this article are from AbleDocs, which is a company I worked for in the past. I'm not in sales, however, and this article isn't an ad. I was using AbleDocs' products long before I started working for them, and my high regard for them was one of the reasons I joined the team. I tried to make this article less of an evaluation and more about the capabilities of each product so keep that in mind as you read on.

For new and seasoned PDF remediators, the go-to tool for making PDF documents accessible is Adobe Acrobat Pro. Naturally, a document's source application is of utmost importance in the PDF remediation process—that is, you want to try to create as accessible a PDF as possible to start with—but we typically end up toiling in Adobe Acrobat to make the PDF file fully compliant. And as powerful of a tool as Acrobat is, it can't do it all. Or to state it more accurately, Acrobat can't do it all *well*. Ask anyone who has exported a PDF from InDesign and then had to move all of the <table> tags out of the containing <p> tags, and they'll tell you it's no fun.

So, we often need to reach for other tools to help us along the way. This is not to put down Acrobat at all, but I consider it just one tool in a larger remediation workflow. I like to remind folks that it's called a *workflow* because the content moves through several applications to achieve the desired result. In truth, it's not very different from other workflows where we need a page layout tool, an image editing tool, and an illustration tool to get the job done.

Meeting Compliance

Let's start off with the topic of meeting compliance. In the world of accessibility, there are several different targets or standards that quantify whether or not a document is considered accessible. Those standards include but may not be limited to:

- WCAG 2.0/2.1: The W3C (World Wide Web Consortium) developed the Web Content Accessibility Guidelines that establish requirements on a web page or a PDF document that are required to make a document accessible. WCAG is arguably one of the most widely used standards and is also referenced in some areas of the PDF/UA standard.
- PDF/UA: PDF Universal Accessibility is a complete definition of a set of requirements for universally accessible PDF documents. This format applies clear specifications for both assistive technologies as well as PDF reading software.

- Section 508: This is the U.S. Government's accessibility program. Many of the requirements map to the WCAG standard.
- HHS: The U.S. Department of Health and Human Services. The HHS developed their own standard that they use for establishing what makes a document accessible.

Determining if a document meets a particular standard can be a challenge. It's difficult for a human being to look at a document and verify its compliance. For this reason, we as PDF remediators rely on software to help us determine if a document meets a particular standard. That's not to say that you shouldn't check your document using assistive technology that an impaired user would use to read a document, but checkers make quick work of identifying obvious problems in a file. Here are three that you need to know about.

Acrobat Accessibility Checker

Arguably the most convenient checker available is the Acrobat Accessibility Checker. Since it's built into Adobe Acrobat, you can simply open up any PDF file and run the Acrobat Accessibility Checker to check the compliance of a document. The Acrobat Checker is found within Adobe Acrobat by choosing the Accessibility Tool and clicking the Accessibility Check button. The Acrobat Checker will notify you of such common problems as out of order heading structure, figures with no alternative text, tables with missing header rows, and a missing document title—all of which are important items that need to be fixed regardless of which standard you are aiming for. The problem, however, is that the Acrobat Accessibility Checker only loosely checks to the WCAG standard and does not officially check against *any* recognized standard. It gets you to a certain point, but then falls short (**FIGURE 1**).

PAC 2021 Checker

The PAC 2021 Checker (PDF Accessibility Checker) is distributed by the PDF/UA Foundation and is available at no charge. As it's a standalone application, you can have it running in the background and just drop a PDF on

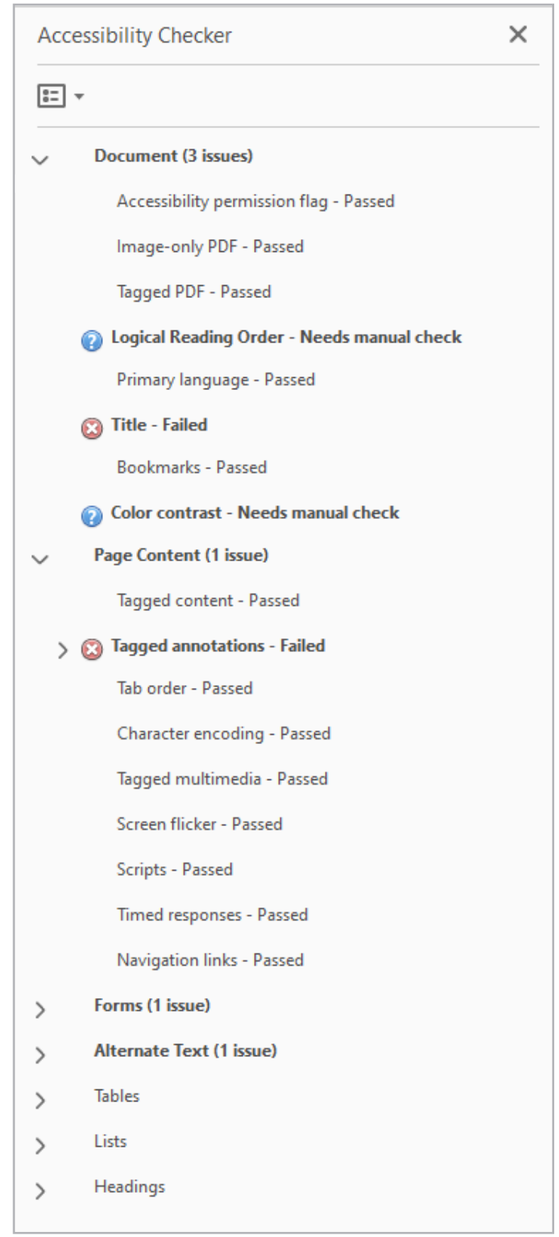


FIGURE 1. The Adobe Acrobat Accessibility Checker checks for basic problems in a file.

the PAC 2021 Checker to check compliance as you work. The PAC 2021 Checker checks for PDF/UA and WCAG compliance and provides you with a robust list of checkpoints (including color contrast) that it uses to determine if a document is compliant. In my testing, I found the PAC 2021 Checker to be quite fast and easy to use.

The PAC 2021 Checker also provides a Results in Detail option that allows you to click on an error and identify which object in the document is causing that error. You can also view the logical tag structure of the document as well as view a Screen Reader Preview of the document, which is a great way to visually evaluate document structure and order of elements. You can also generate a detailed report providing evidence to clients that the PDF file does in fact meet the PDF/UA standard. Although the PAC 2021 Checker is currently available only for Windows, a macOS version is on its way (**FIGURE 2**).

Commonlook Validator

Commonlook offers a robust, free PDF validator for checking compliance of PDF files. The Commonlook Validator is available free of charge, runs as a plug-in within Adobe Acrobat, and pretty much takes over the Acrobat interface when invoked. I didn't find this to be a problem as it allowed me to focus strictly on the compliance of the file.

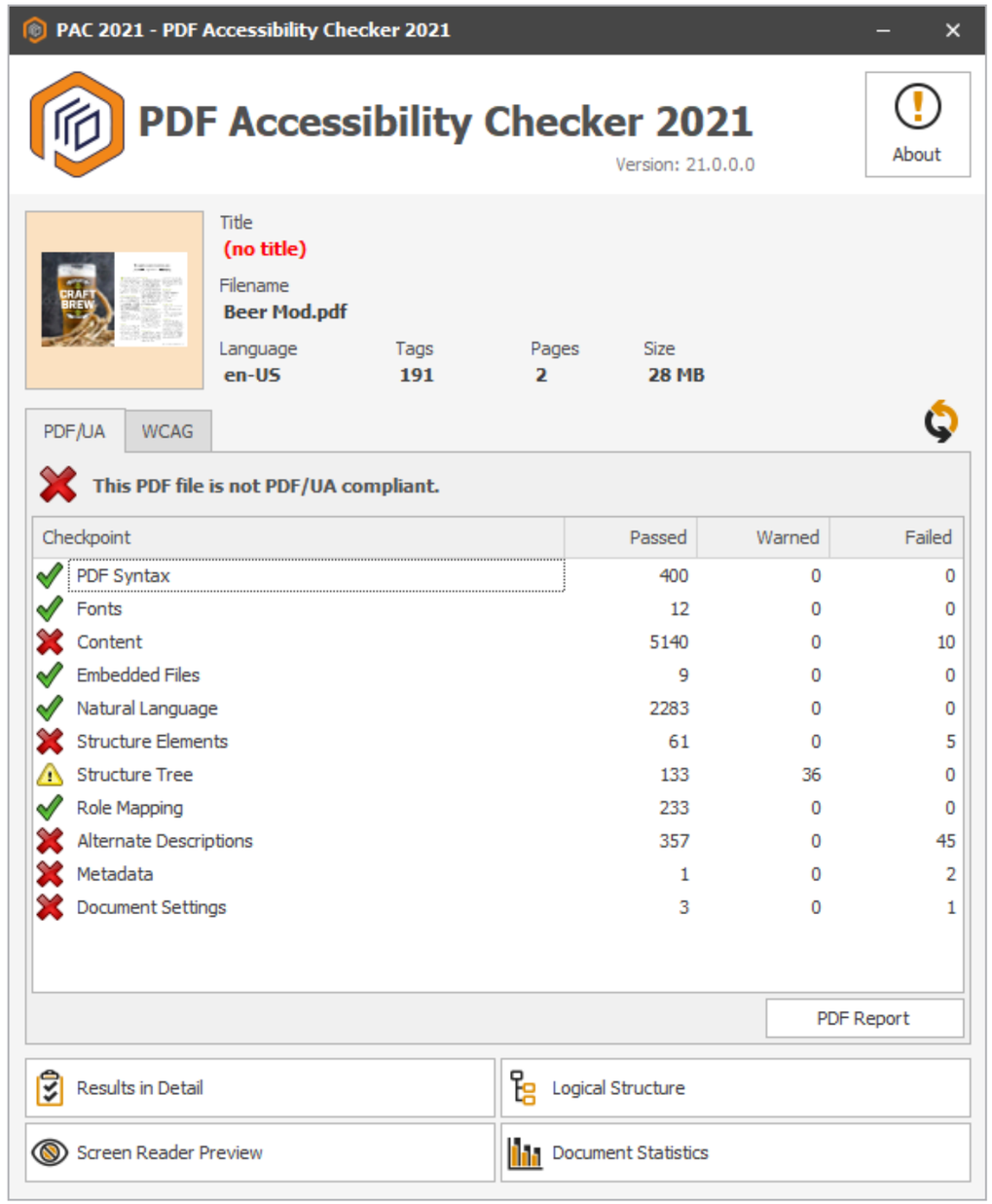


FIGURE 2. The PAC 2021 Checker verifies compliance to the PDF/UA and WCAG standards.

One frustration I did experience was that it was very slow to load (as in several minutes), and the only apparent way to close the Validator is to quit Adobe Acrobat.

However, one of the things that I really like about the Commonlook Validator is its ability to verify compliance to a number of specific standards including:

- Section 508 (2001)
- WCAG 2.0 AA
- PDF/UA 1.0
- HHS
- HHS (2018 Regulations)
- WCAG 2.1 AA

The ability to verify compliance to all of these standards makes the Commonlook Validator a great tool for anyone who remediates files for a variety of clients and therefore must meet a variety of standards (**FIGURE 3**). The report generated by the Commonlook Validator allows you to filter results by warnings, failures, and what is passing, which makes it easy to narrow down the results to what really matters.

The Commonlook Validator is available for Windows only, so if you're a Mac user, you'll need to set up a Windows partition running an emulator in order to use this product.

Checking Color Contrast

Anyone who has used the Acrobat Accessibility Checker will immediately note that in the Document category of the report is a warning that says "Color Contrast-Needs manual check." This alert concerns a lot of people, especially new users, because they interpret it as an error. The reality is that it's not really an error at all. It's a notification indicating that the Acrobat Checker isn't smart enough to check color contrast of text against its background, and therefore needs someone smarter (like you) to check the color contrast. That's all well and good, but the next question, of course, is, "How do I do that?"

It's a good question, and the answer is not as straightforward as you might hope. First let's dig into what that error is actually saying. WCAG Success Criterion 1.4.3 states that any text or images of text need to have a contrast ratio between the text and its background of 4.5:1. To check the color contrast of

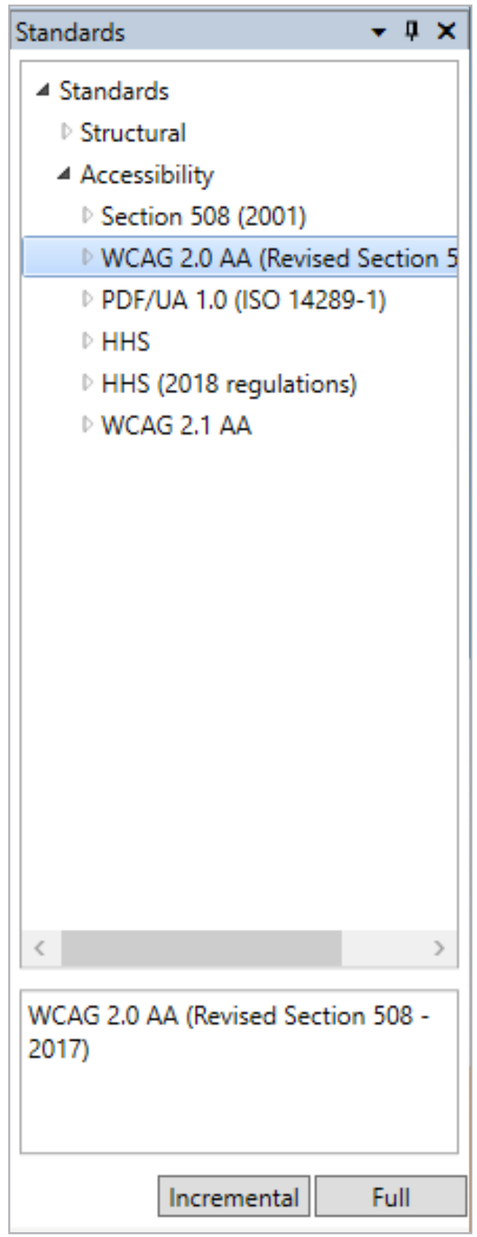
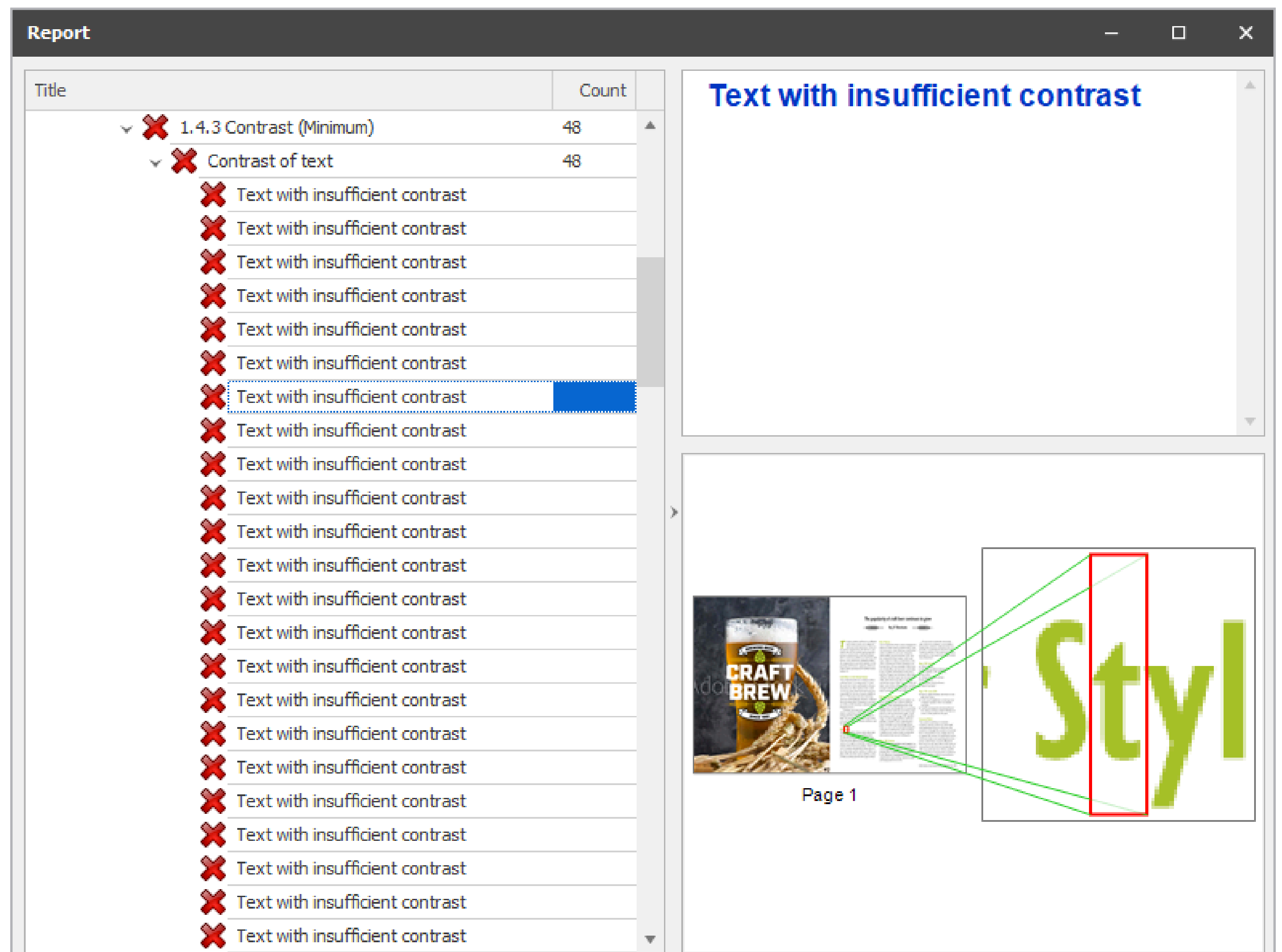


FIGURE 3. The Commonlook Validator can verify compliance to a number of standards.

elements on a page, you need to reach for some third-party tools.

In her article in this issue, Colleen Gratzer goes into detail on contrast checkers like WebAIM, Adobe Color, and TGPi's Colour Contrast Analyser. Unfortunately, most of them require you to manually sample colors to evaluate the contrast, which can be time-consuming. A more automated solution just became available in the form of the PAC 2021 Checker. Released in early August 2021, it contains a built-in checker that automatically evaluates color contrast between any text elements and the background. So you can simply run the entire file through the PAC 2021 Checker and see the results. Personally, I've incorporated this product into my workflow since I think it will save me a huge amount of time when I no longer have to manually sample colors to evaluate the contrast (**FIGURE 4**).

FIGURE 4. The PAC 2021 Checker automatically evaluates color contrast of text elements and displays the results in the report.



Verifying Structure and Order

One of the essential PDF remediation tasks is a process that those of us in the business refer to as "walking the tags tree." This refers to clicking the tags in the Tags pane of Adobe Acrobat (or another tool) and verifying that the order in which the tags appear in the tags tree is in fact the order in which we want our content to be read, as well as verifying that content is tagged using the proper tag type (*i.e.*, H1, H2, H3, and so on). This is crucial work because most popular assistive software applications (like JAWS and NVDA) use the tag order as the order in which content is read.

Walking the tags tree is not particularly difficult, but it can be time-consuming in a long document. Fortunately, there are

some third-party applications that can ease the burden.

pdfGoHTML

pdfGoHTML is a plug-in for Adobe Acrobat and a fantastic tool for evaluating content, order, and structure in a PDF file. This free plug-in (available for both macOS and Windows) is a must-have tool to have in your remediation arsenal. The plug-in renders the PDF file as a web page, which might seem strange at first. But this approach makes it easy to evaluate the tag order because everything is rendered in the same order as the content appears in the Tags pane. In addition, pdfGoHTML offers multiple views of the content, including a Low-Vision view, Dyslexia view, and my personal favorite, Structure Tags view. In the latter view, the tag assigned to the content is listed next to the content, making it easy to not only evaluate the order of content, but also how that content is tagged (**FIGURE 5**).

PAC 2021 Checker

We've discussed the PAC checker earlier in this article, but I bring it up again here because it includes a Screen Reader Preview that renders the PDF file in a continuous view

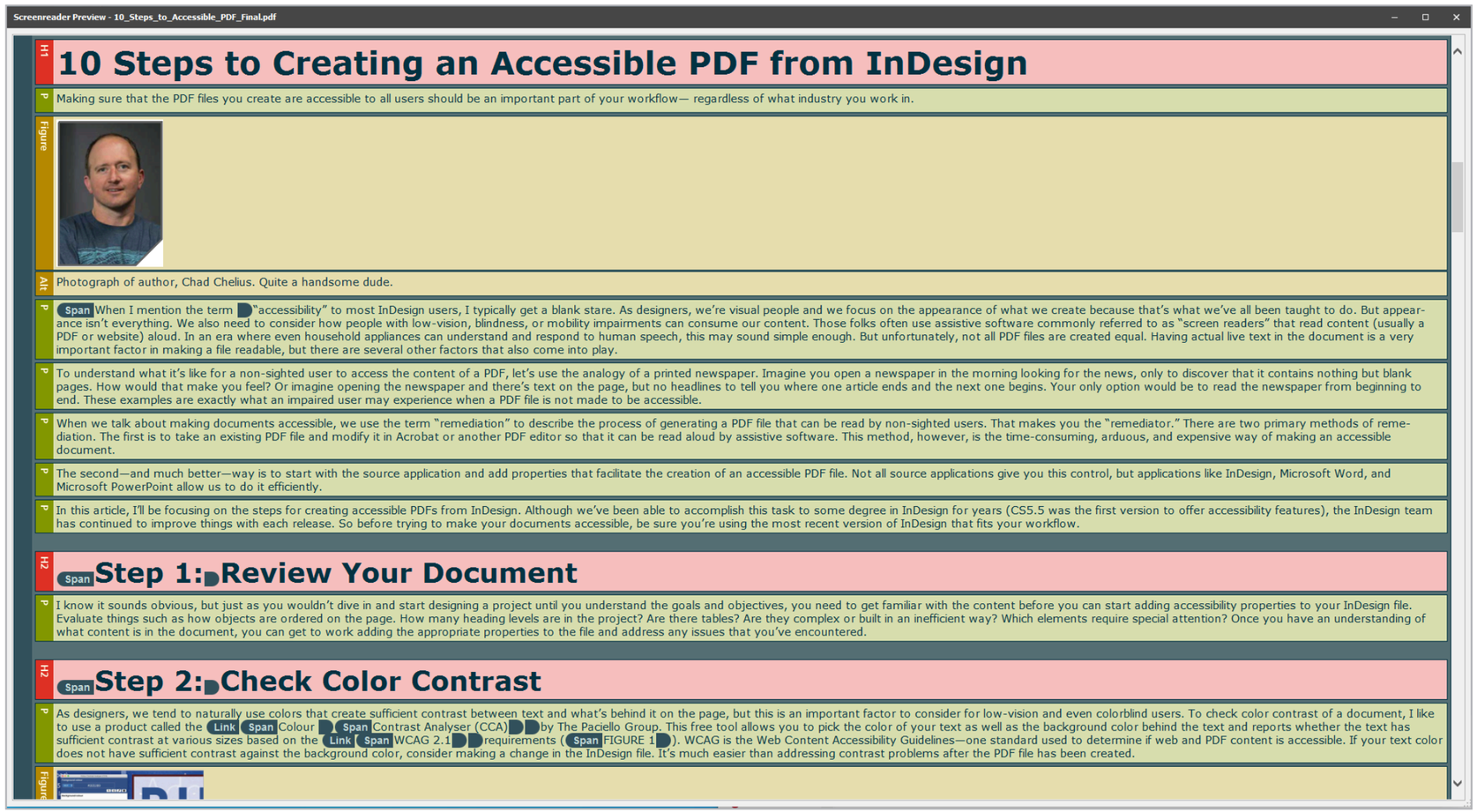
FIGURE 5. The Structure Tags view available in the web rendering of the pdfGoHTML plug-in makes it very easy to evaluate order and structure of content in a PDF file.

(very similar to the Structure Tags view in pdfGoHTML) that allows you to check the order of content and how that content is tagged. The location of this tool is convenient; after you're done checking documents for compliance, you can quickly jump to Screen Reader Preview to evaluate the document further (**FIGURE 6**).

More Robust Tools

Folks who remediate files know that Adobe Acrobat can do pretty much everything we need in order to make a file compliant. Doing it *efficiently,* however, is not always Acrobat's strong suit. For this reason, depending on the volume and complexity of your files, you may find yourself searching for a better mousetrap. So, let's look at some tools offered by third-party vendors that can significantly improve the PDF remediation experience for you.

FIGURE 6. The PAC 2021 checker contains a Screen Reader Preview for evaluating order of content and how it is tagged.



Commonlook

As a major player in the PDF remediation space, Commonlook offers two products that can make the PDF remediation process a lot easier: Commonlook PDF and Commonlook Office.

Arguably one of the most robust PDF remediation tools available on the market, Commonlook PDF is designed for a wide variety of needs, including remediating a PDF that was created without any regard for accessibility. Commonlook PDF is an Acrobat plug-in, so it leverages the Acrobat interface while you're using it. Anyone who remediates PDF files knows the mess of a poorly structured (or untagged) file, and Commonlook PDF gives you tools to fix these problems.

Where Adobe Acrobat is limited to manual adjustments, Commonlook PDF not only allows you to search and replace tags, but it also allows you to contextually find tags and change their structure. For example, find all <table> tags within any <p> tags and remove the containing <p> tag. You can also linearize tables with the click of a button or define a proper list from a range of text, and even regenerate a table of contents. One feature that I like very much is the ability to select a tag or group of tags and move them up or down in the tags tree using a keyboard shortcut. Overall, Commonlook PDF adds automation to the PDF remediation process, which saves a lot of time and effort (**FIGURE 7**).

The second tool, Commonlook Office, is available as a plug-in for both Microsoft Word and Microsoft PowerPoint. This tool adds functionality so that when a document is exported from

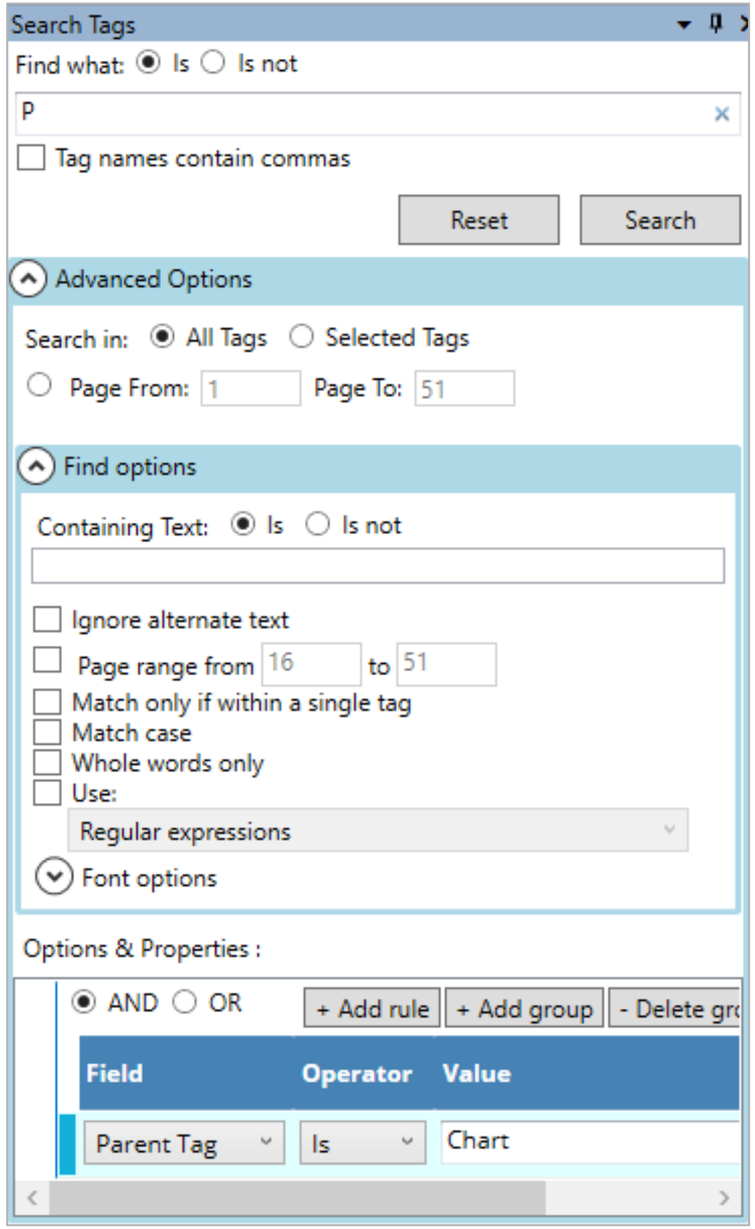


FIGURE 7. Contextually finding and replacing tags is one of Commonlook's strengths.

those apps to PDF, the result is a fully compliant PDF file with little to no additional intervention. Anyone who regularly produces PDFs from Word or PowerPoint files will appreciate the amount of cleanup work that this plug-in reduces.

AbleDocs

Another major player in the PDF remediation space, AbleDocs offers a number of products that ease the process of making accessible PDF files. AbleDocs takes a different philosophical approach to PDF remediation. One of their fundamental goals is to make inaccessible content obsolete. Rather than remediating existing PDF files, they want you to always start with a source file from Word or InDesign. That being said, they realize that it's not always possible and have developed two products to aid in the PDF remediation process. Let's look at them.

axesPDF**:** A stand-alone application, axesPDF allows you to open existing PDF files to fix common errors encountered in the PDF remediation process. Built into axesPDF is the PAC checker. With it, you can check a document against the PDF/UA standard. Errors are listed, and double-clicking on one will take you to the offending object in the document and in the tags tree. Using axesPDF you can easily change the tag name, change the role map of objects, change the placement attribute, as well as adjust the scope of table header cells among many other things.

That's all great stuff—but wait, there's more! axesPDF allows you to quickly artifact all non-tagged content, globally rename tags, and set the contents key for link annotations. Some of the more powerful features of axesPDF include the ability to create destinations from headings, captions, tables, and other elements as well as the ability to edit the Unicode mapping for fonts. One of the features that I really enjoy is the Table Editor, which gives you the ability to define row and column headers, define the scope for the header cells, and associate header cells with data cells (**FIGURE 8**). I found this to be a huge timesaver for documents containing a lot of tables and even complex tables as well.

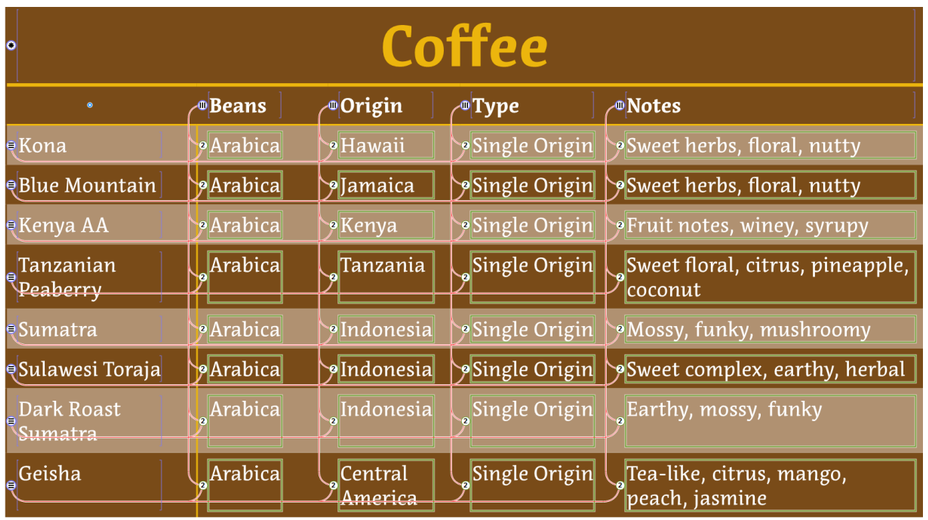
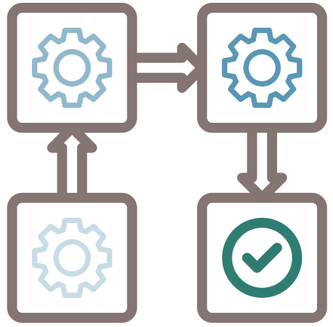


FIGURE 8. axesPDF contains a powerful table editor that makes it easy to accomplish tasks like associating header cells to data cells.

axesWord**:** Similar to Commonlook Office, axesWord is a plug-in for Microsoft Word that gives you the ability to export to a PDF file while alleviating a lot of Word's shortcomings

related to accessibility. Among the plug-in's strengths are the ability to define continuing numbered or bulleted lists with other paragraphs between list items. You can easily (and properly) define column and row headers in complex tables. When tables are used inappropriately in Word (*i.e.*, using table cells like text boxes to lay out a page), you can linearize those tables so they output as content instead of a table. axesWord gives you role-mapping capabilities so you can easily define the role for various styles used in a document which is much easier than doing it natively in Word.

What I really appreciate about axesWord is what it does automatically without any intervention on my part. For example, if extra returns were used in the Word document to add vertical spacing between paragraphs, axesWord is smart enough to tag them as artifacts. Also, path elements from tables which normally require post-export intervention, are simply artifacted automatically on export. Essentially, both axesWord and Commonlook Office take what Word does natively, and just make it better.



InDesign Plug-in: MadeToTag

While far from perfect, InDesign still ranks as the best source application that can be used to generate accessible PDF files. Tables, footnotes, endnotes, table of contents, hyperlinks, and tags are all handled exceptionally well. But while there are many things that I praise about the accessible PDF output from InDesign, I have just as many complaints. (I'll spare you the full list.) For this reason, it's worth looking into some third-party tools that can make accessible PDF output from InDesign even better.

MadeToTag is an InDesign plug-in by axaio software GMBH. I wrote a review of MadeToTag in Issue #62 of where I was impressed by both the features and the interface. After installing MadeToTag, you'll find a new panel in the Window menu of InDesign. MadeToTag addresses a lot of the shortcomings of InDesign, including the following abilities:

- Define scope and row headers in a table
- Create links in footnotes
- Locate common problems such as faux lists, empty paragraphs, and figures with no alt text
- Locate text with insufficient contrast
- Set the language of selected text
- PDF optimization

That is by no means a complete list. I've been using MadeToTag for a while now and I can tell you that many files that I now export out of InDesign are immediately PDF/UA compliant without any further intervention after export. If you use InDesign as the source for accessible PDF files, I highly recommend that you give MadeToTag a try.

PDF Remediation Services

Sometimes it's worth weighing the effort and time required to remediate a PDF file versus the cost. I often work with folks who

are overwhelmed by the idea that they need to remediate a 200-page PDF file. There are times when it may not be worth the effort to do these projects yourself. For those situations, there are many companies that offer PDF remediation services at a reasonable cost. In addition to AbleDocs and Commonlook whom I've mentioned earlier in this article, there are many large and small organizations who offer PDF remediation services at a fair price and a reasonable turnaround time. I don't want to dive into pricing or details for those services in this article; my only goal here is for you to know that this option exists. For projects that simply require too much effort on your part, I encourage you to research some of these companies as an alternative to remediating PDF files yourself.

Making a Workflow that Works

The reality is that PDF remediation isn't quick, easy, or straightforward. It requires skill, knowledge, and experience on the part of the person who is remediating the PDF file. So, anything you can use to streamline the process can significantly change your workflow for the better. The tools outlined in this article are ones that I recommend, and many of them I use on a daily basis. I encourage you to explore some of these tools and decide for yourself which ones can best help you in the quest to make accessible PDFs.

Chad Chelius *is a trainer, author, consultant, and Director of Training Solutions and Principal at Chax Training and Consulting. He resides in the Philadelphia area and has been using Adobe products for over 25 years. As an Adobe Certified Instructor, Accessible Document Specialist, and consultant he teaches and advises on all Adobe print and web products, specializing in InDesign and InCopy workflows, Illustrator, automation, and PDF accessibility using InDesign, Word, and Adobe Acrobat. He works with clients both large and small in and outside of the United States, helping them to solve design, workflow, and accessibility challenges using Adobe products.*

CreativePro

Become a Member

Get year-round access to resources that will keep you current and strengthen your skills

For just $6.50/month (billed annually), you'll get access to valuable benefits, including:

- ***CreativePro Magazine***
- **How-To Articles**
- **Downloadables**
- **Discounts**
- **Webinars**
- **Top Tips Ebooks**

Sign up now at CreativePro.com/become-a-member