

The Fundamentals of an Accessible PDF File

Chad Chelius

www.accessibilityunraveled.com

chad@accessibilityunraveled.com

[@chadchelius](#), [linkedin.com/in/chadchelius](https://www.linkedin.com/in/chadchelius)

The 80/20 Rule

In a perfect world, every PDF we exported from our source applications, would be 100% compliant and ready to go! The truth however, is that those source applications currently can't do everything (although it's on my wishlist). Not only are there limitations of those source applications, but poor choices during the design process by the document author, can often cause elements to output incorrectly or at the very least, in an undesirable way. For this reason, I try to work within the 80/20 rule. This means that I try to do as much as possible in the source applications before I export to a PDF file, I'm about 80% there, but then I'll need to do another 20% of work directly inside of Acrobat or another PDF remediation tool to get the file fully compliant. Now I say 80/20 but depending on the complexity or simplicity of the file that could change to 90/10 or even 70/30. It will vary for sure but the approach is the same.

When I start the remediation process in Acrobat after I export a PDF file, I follow a general approach so I can begin ticking off the list of items that need to be completed. This of course can vary depending on the file but overall my approach is pretty much the same each time. Sometimes however I can skip some of the items because they don't apply to the file that I'm working on. Regardless my approach is the same and I adapt that approach depending on the needs of the file.

Walking the Tags Tree

Walking the tags tree doesn't require any cardio fitness and it won't count towards your daily steps. All it requires is your knowledge of the document and your pointer finger on the keyboard. Walking the tags tree is achieved by highlighting the first tag in the document and using the down and up arrow keys on your keyboard. As you move up or down the tags tree, you'll see the object associated with that tag highlighted on the page. This allows you to see how an object is tagged as well as the order that the objects appear in the tags tree. This order is important as it is used by the most common assistive technologies in use today (JAWS and NVDA) and ensures that the content will be read in the appropriate order. Walking the tags tree serves several purposes:

- To make sure that content is being read in the appropriate order
- To identify elements that are tagged incorrectly
- To make sure that content is tagged correctly
- To clean up the tags structure when necessary

Although walking the tags tree can be a bit arduous in longer documents, it's the only way to ensure the integrity of the content in a document. You'll notice that when running the Acrobat Accessibility Checker, you always see a notification that the Order needs a manual check. That's because only we as human beings can ensure that the content in the document is being read in the appropriate order. Even the best AI still can't do this for us.



Walking the tags tree in Acrobat.

Reorganizing Content

Sometimes content is created with the visual appearance in mind. But when it comes to accessibility, this doesn't always lend itself to being read in a proper or logical way. In some cases, you may find that you need to reorganize and/or restructure content to facilitate reading that content using assistive software.

A good example is when content is structured in a way that it looks like a table but isn't structured as a table. You may find that structuring the content into a table will be much more logical from a screen-reader perspective.

Also, you could find the reverse. We as designers are notorious for using tables simply to align and position elements on a page. For accessibility, tables are meant to read data. If the content of the table isn't data or the table doesn't have a logical header, it will fail validation in all standards. In cases such as these, we often need to linearize the table. Effectively pull the content out of the table, delete the table itself, and have the content read in a logical order.



Reorganizing content in the Tags pane

Reading Order

The Reading Order in a document is one of the more confusing aspects of PDF remediation. It's often used interchangeably to describe the tag order in a document and often misinterpreted as the same. The reason that we have two reading orders cannot be explained here but if you're interested in knowing why, check out [this ChaxChat podcast episode](#) to learn the reason behind it.

Regardless of the reasons as to why there are two different orders to contend with in a PDF, both need to be defined when remediating a PDF. Why you ask? Well WCAG success criterion 1.3.2 states that documents need to be read by all technologies. Due to the fact that different assistive technologies use different methods of interpreting content order, both Tag order and Reading Order need to be set.

To be clear, InDesign gives you all the control you need to set the Reading Order of an exported PDF file. By using the Layers panel in InDesign and organizing content from bottom to top, you will achieve proper reading order in the exported PDF file. Occasionally challenges can occur doing this in InDesign so some cleanup may be required after the PDF file is generated.

Logic would tell you that to fix the Reading Order in a PDF file, you would use the Order panel in Adobe Acrobat. The harsh reality however is that adjusting content in the Order panel can absolutely annihilate your tag structure. So after painstakingly walking the tags tree, you could find your tag structure a complete mess!

So what do do? Interestingly enough, you can adjust the Reading Order of a document using the Content pane. Using this technique allows you to define the Reading Order appropriately without affecting the tag structure in any way. A technique that I find useful is to open the Reading Order panel and leaving it open as I make adjustments in the Content pane. Doing this allows me to see the number order of each item and the update that occurs

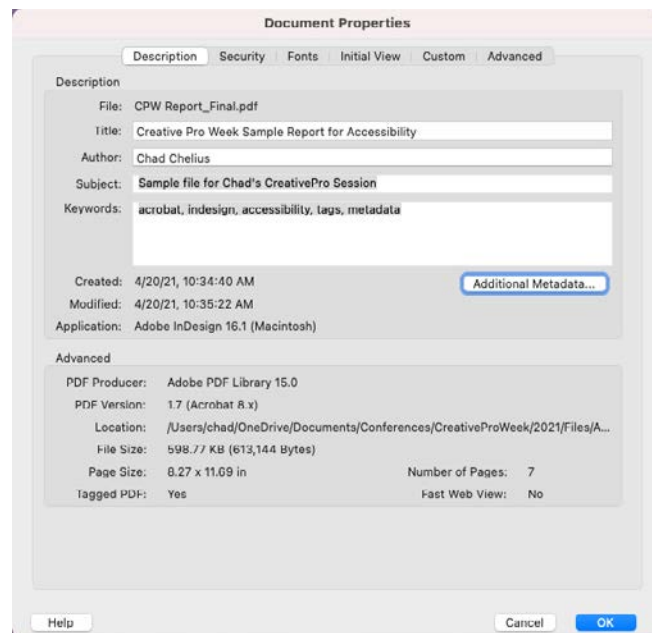
to make changes in the Content pane. Without this you fly blind as you're making changes but with the Reading Order panel active, you can see the numbers and make adjustments easily.



Adjusting the Reading Order using the Content Pane.

Metadata

You're probably aware that every file requires Title metadata that is displayed and read when the file is opened as opposed to the file name. Although technically the file is the only required metadata, because these files are almost always posted online, it's helpful to also add an Author, Subject, and Keywords to the file to aid in Search Engine Optimization to help users find your file when using a search engine. The metadata should have been added in the source file prior to export, but it's something I always check when I open the PDF file. Often the metadata is incomplete and requires me to add in in Acrobat. The metadata in Acrobat can be found in the Properties dialog box which can be accessed by choosing File > Properties or by pressing Cmd+D (Mac) or Ctrl+D (Windows) on your keyboard.



Viewing metadata in the Properties dialog box.

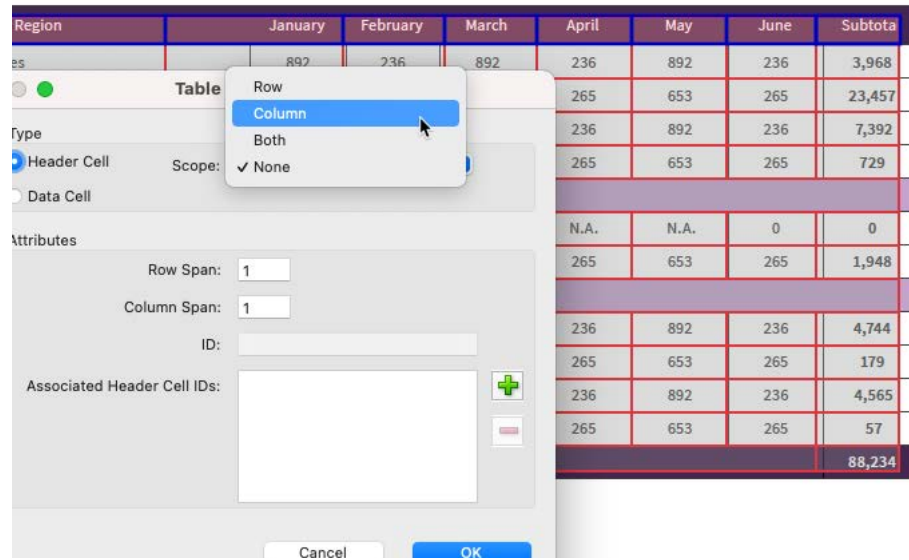
Addressing Table Issues

When it comes to tables, there's a lot of opportunity for fixing problems. Tables are often created incorrectly or misused in a document, making the content inaccessible for someone using assistive software. The cause of this can range anywhere from user error to the source application's inability to properly output a table. Some of the more common issues that need to be corrected when remediated a PDF file are:

- Tables with missing headers
- Tables with missing row headers
- Merged cells incorrectly defined
- column headers positioned mid way down in a table
- Scope not defined for header cells (PDF/UA requirement)

Some source applications are better than others when it comes to outputting tables properly. Adobe InDesign does quite well with merged cells as the ColSpan or RowSpan is properly defined. This is a huge time-saver compared to other applications. InDesign cannot however define row headers in a table which means you'll need to define them in the PDF file after export. InDesign also places tables within a <P> tag within the Tag structure of a PDF file. Although some standards accept this, others (such as PDF/UA) flag this as an error because structurally, a table shouldn't live within a <P> tag but instead should reside on it's own.

3 Income by office 2020
on submitted reports.



Adjusting header cell properties using the Table Editor.

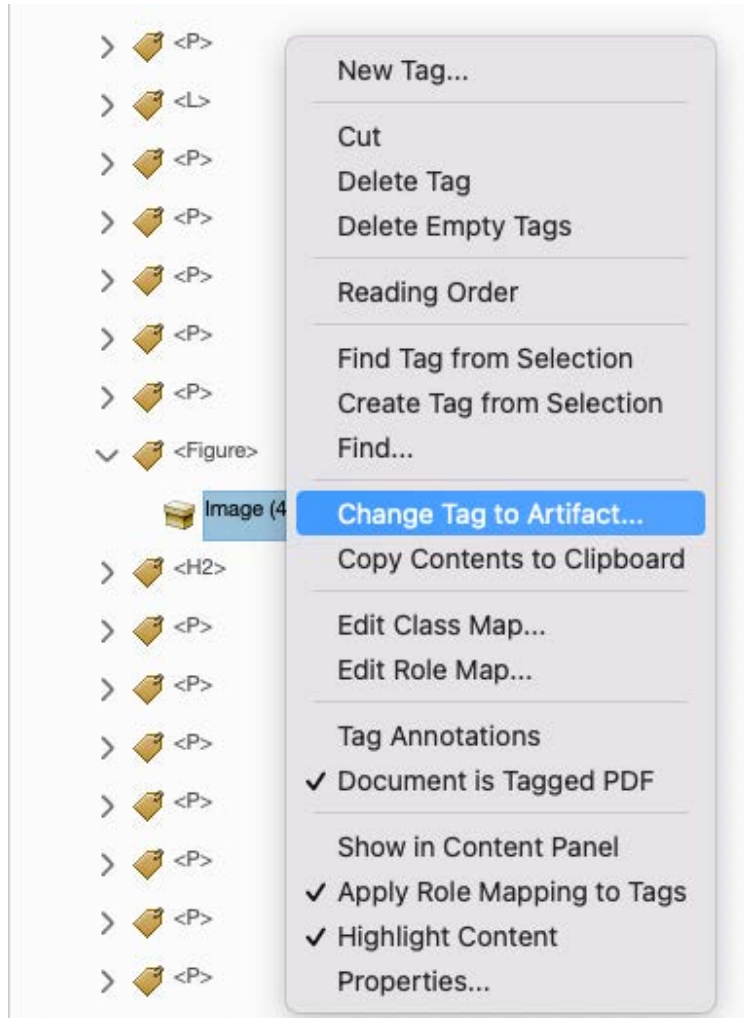
Figures

All figures in an accessible PDF files need to contain alternate text. The job of alternate text is to describe the figure (graph, graphic, image, photo) in its context. I try to imagine closing my eyes and if that alternate text was read aloud to me, could I visually paint a picture or envision the figure being described. Web AIM has a [good article](#) on writing alternate text to give you some guidance. The general idea for alternate text is to be as concise as possible while still accurately describing the context of the figure. Remember, alternate text has no tag structure so for users of assistive technology, they read all of the alt text or skip. There's no in-between.

You may find however that for designed documents, often figures and imagery are used specifically for aesthetic reasons and don't really have any value in the context of the project. In cases like these, you can Artifact the figure so that it's not read by assistive software.

Charts and graphs can often be particularly challenging to describe using alternate text. Simple pie charts aren't too bad but a graph showing values over time with several data points can be particularly challenging.

Another issue I encounter quite often is the captions for figures. Captions should be tagged as <Caption> and then nested within the <Figure> tag that it's associated with. Most source applications don't do this with the exception of InDesign and even then, unless you use the Caption feature in InDesign (Live Captions to be specific), it won't happen there either.



Artifacting a figure in the Tags pane.

Hyperlinks

Adobe Acrobat auto-detects implicit hyperlinks in a document. What I mean by that is that if you have text in a PDF file that reads www.creativeproweek.com, Acrobat detects that and makes that text a clickable URL. This is quite helpful and has caused many users to rely on this feature as something Acrobat will just do for them. The problem with this feature is that those auto-detected hyperlinks are not accessible. In order for a hyperlink to be accessible in a PDF file, a hyperlink needs to be tagged using a <Link> tag with an OBJR object within the <Link> tag. When tagged properly like this, assistive software will announce the link and provide the user with the ability to follow that link.

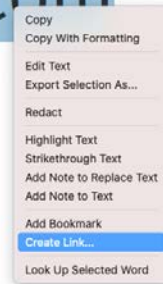
For this reason, it's important to define hyperlinks in the source application. InDesign, Word, and Powerpoint, all provide this feature and provided the link has been created in the source application, the PDF file will contain the necessary <Link> tag to make the link accessible.

Unfortunately, sometimes people forget to do this in the source application. Fortunately, Acrobat gives us the ability to create links in the exported PDF file albeit with somewhat inconsistent results.

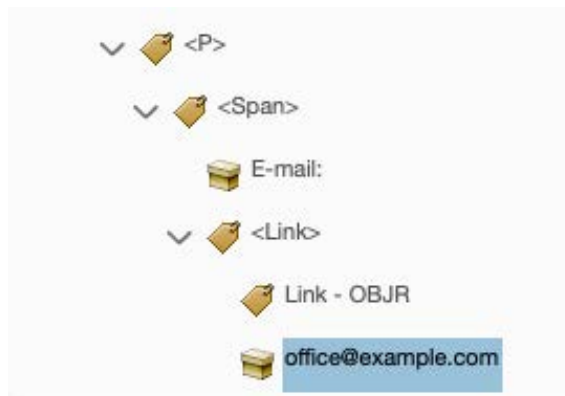
To create a link to text in a PDF file, choose the Selection tool in Acrobat and select the text in which you want a hyperlink. Now right-click on the selected text and choose Create Link. In the Create Link dialog box, choose "open a web page", then click next. Enter the url for the link in the Edit URL dialog box, and click OK.

Creating a hyperlink in Acrobat using the Edit tool and drawing rectangles around areas that you want to become a hyperlink in and of itself will not yield an accessible link. You can however choose Find from the Option button at the top of the Tags pane and choose Find. In the Find drop-down menu choose Unmarked Links and click Find. It will find any hyperlinks created by drawing a rectangle. Click the Tag Element button to properly tag.

office@example.com




Creating a link to text in Acrobat.



The result in the Tags pane.

The Preflight Panel

One of the most powerful features in Adobe Acrobat for addressing common Accessibility issues is the Preflight panel. Known primarily to print production professionals, the Preflight panel offers a number of accessibility fixes that can save you a lot of time and effort when meeting compliance especially for PDF/UA.

Start by opening the Print Production tool in Acrobat, then click on Preflight. In the Preflight panel, choose PDF Standards from the top drop-down menu, then click on the wrench icon . The Document, Document Info and Metadata, and Interactive Elements and Properties categories contain a lot of great fixes to address common accessibility problems that can take a lot of time to fix manually.

Some of the fixes that you might find useful include:

- Add unique ID to Note elements in tagging structure
- Correct language for list labels
- Create Bookmarks from Headings
- Embed fonts
- Mark all non-structure elements as artifact
- Merge Adjacent headings
- Remove empty group and block-level elements from tagging structure
- Set document language to language in tagging structure
- Set labels in unordered lists to generic bullet character
- Show document title in Title Window
- Set PDF/UA-1 entry
- Create Content entry for Link annotations

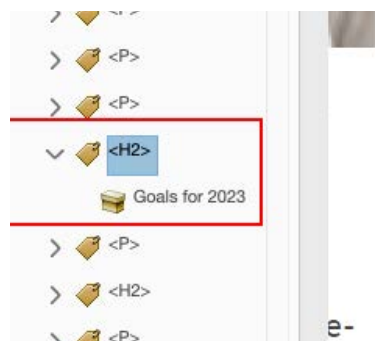
I've created a [custom preflight set](#) that runs a bunch of these at one shot. Feel free to download it for your own use.

Editing a PDF File

My approach to PDF remediation is to do as much work in the source application that will be generating the PDF file as possible to facilitate the creation of the accessible PDF file. This is the most efficient approach and it's repeatable. Meaning if you ever need to go back and make an edit to the source file, all of those properties defined in that source file are replicated each and every time you export to a PDF file. This saves time, money, and energy in the PDF remediation process.

That being said, there are times when you have to start with only the PDF file (avoid this approach whenever possible). The challenge here is that the cake is already baked at this point. Any issues you find in the document such as color contrast issues, typos, etc. can only be done in the PDF file if they can be done at all. Editing color in an existing PDF is painful at best.

Using the Edit tool in Adobe Acrobat is a sure way to break your file. Avoid it at all costs. You'll find that after using the Edit tool in Acrobat, your tag structure has become a hot mess. You might be able to get away with small edits if necessary but you'll definitely need to fix things in the Tag structure if you do.



The tags pane prior to making an edit using Acrobat's Edit tool.

lenim acil (MIN) atio iurem wisc
tionsequisim wis loremipsum ij
andreet dolutatueros.

Goals for 2023

Uipsusciduntaa to ese wissim u
si conullam em facinim feuguei



The tags pane after making a minor edit using Acrobat's Edit tool.

ny facinim; ute ibh etuerosto sustruc
na adigniscipisee os dio nonsecte
lenim acil (MIN) atio iurem wiscil wis
tionsequisim wis loremipsum ip nos
andreet dolutatueros.

Ojectives for 2025

Uipsusciduntaa to ese wissim utpat
si conullam em facinim feuguer el pi
dolorpercing ero exeriurero it rit fa-
cinim. Laore suscipi nis feuisi is lur
consequis magnis: eugiatet, zzriuste
nim tumsandionse exerostrud; iri-
ustrud velenibh; hendrer; adigniscip
lortie; nis dolorer atio tetummy.

Relying on Checkers

Accessibility checkers are useful tools for helping us to identify problems in our accessible PDF documents. There are a number of checkers available for you to use including the one built in to Adobe Acrobat.

I like to use the Acrobat Checker as a first line of defense. It's useful for letting me know obvious things that I've missed in a document (i.e. tables with no headers, heading tags out of order, missing alt text, tab order not defined), just to name a few. It's convenient and easily accessible directly within Adobe Acrobat.

The Acrobat Checker is very helpful but it's not comprehensive. The Acrobat Checker unfortunately is not fully checking your document to any recognized standard. We could argue that it's loosely checking to the WCAG standard, but certainly not fully.

For this reason, it's often necessary to look at other alternatives. One alternative is the free [Commonlook Validator](#) which can check your document to a variety of recognized standards and performs a much more comprehensive check than the Acrobat Checker.

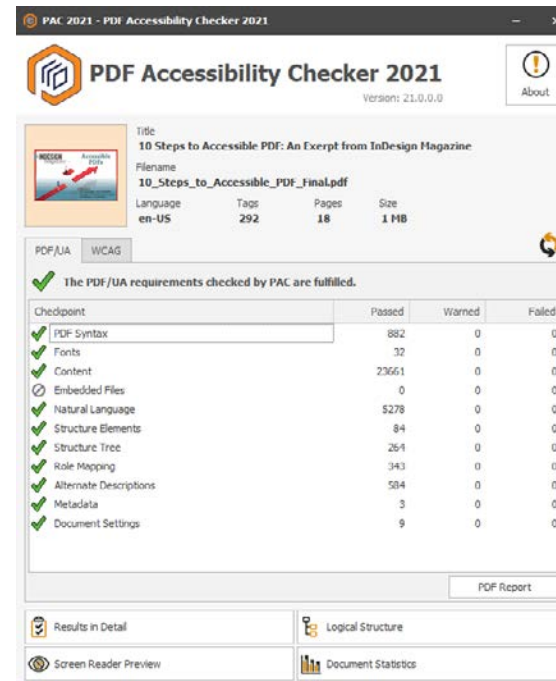
Another potential option is the [PAC 2021 checker](#) that can check to the WCAG, PDF/UA and even evaluates color contrast as well. This checker should be released in the near future so keep your eyes open for this one!

With all of this being said, checkers for the most part are pretty dumb. I say this with all due respect as I rely on them heavily to catch things that I've missed in a file. But they're far from perfect! They can't tell you that text is missing a hyperlink, or that a table doesn't have row headers defined when it should, or that text is tagged as a <P> when it should be an <H3>, or that a caption should be tagged as such and not as a <P>, or that a figure has no value and shouldn't be read. Should I keep going? We're at a point in PDF remediation where success still relies heavily on a knowledgeable person who can decipher the content of a file and appropriately structure it for

success when someone reads the file using assistive software.

Speaking of which, if you really want to ensure that an accessible PDF file can be read using assistive software, consider installing JAWS, NVDA, and other assistive software applications on your computer and testing your files using those applications. It's the best way to ensure that your content will be accessible and read correctly using those technologies.

One last note about assistive software. Even once you install the applications (JAWS, NVDA, etc.), you can't just dive in and start using it. You need to [learn how to properly use those tools](#). It's software just like any other application and there are correct ways to use them and incorrect ways to use them.



The PAC 2021 Checker.