# Pitfalls of Machine‐Generated PDFs

Still a work in progress

**Slide 2**

**The Presenter**

**Richard U. Steinberg**

[rsteinberg@austinchronicle.com](mailto:rsteinberg@austinchronicle.com) [richard.steinberg@wellsfargo.com](mailto:richard.steinberg@wellsfargo.com)

Richard Steinberg, Senior Accessible User Experience Specialist at Wells Fargo. Got into accessibility in 2006 after writing a story for the Chronicle on how Yellow Cab of Austin was hiring blind dispatchers and had to make sure their software worked with the JAWS screen reader. Texas law changed that year that required state agency websites to be Section 508 compliant. So I’ve been at it for about 16 years and I’ve been fixing PDFs for almost that long.

ht[tps://w](http://www.austinchronicle.com/news/2006)ww.aus[tinchronicle.com/news/2006](http://www.austinchronicle.com/news/2006)‐06‐09/373276/

**Slide 3**

# Outline

* Overview of PDFs
* What are PDF tags and their importance
* Converting HTML to PDF

**Slide 4**

**Overview of PDFs**

* History (including my history)
* History with assistive technology users
* Different ISO standards
* The future of PDFs (?)

**General History**

Many organizations take advantage of providing Portable Document Format (PDF) documents to their users. In 1991, Adobe cofounder Dr. John Warnock launched the paper‐to‐digital revolution with an idea he called The Camelot Project. The goal was to enable anyone to capture documents from any application, send electronic versions of these documents anywhere, and view and print them on any machine. By 1992, Camelot had evolved into PDF when Adobe released it to the public. Few people seem to remember that PDFs predates the widespread use of the internet.

On January 29, 2007, Adobe announced that it would release the full Portable Document Format 1.7 specification to the American National Standards Institute (ANSI) and the Enterprise Content Management Association (AIIM), for the purpose of publication by the International Organization for Standardization (ISO). ISO issues certifications for products that meet their standards across many industries. In 2008, ISO Technical Committee published the first PDF standard, ISO 32000‐1.

**Slide 5**

# “State of the Art” Web 1997

**My History with PDFs**

In the early days of the web, PDFs enabled me to start one of my first websites, which helped place children who had been abused or neglected and in my state’s care for adoption. Our existing recruitment system involved sending printed pages to our offices across Texas where local staff added them to a loose-leaf catalog. Prospective parents would make an appointment with a social worker to visit and view the catalog in person. The graphic designer who created these loose-leaf pages sent PDFs with children’s profiles to our print shop. One day, I asked him if he could give me some of those PDFs to put on our website. In those days, HTML was in its infancy with limited formatting options. PDFs allowed us to reach anyone who visited the site at their convenience. We started in 1994 with 12 children. Today, the site at adoptchildren.org has been converted to a searchable, HTML- based database with close to 1,000 children.

**Slide 6**

**The Evolution of PDFs**

PDFs have evolved over the years. Form fields, radio buttons, and checkboxes were added in 1996. JavaScript, multimedia, XML, and more have also been added over the years. Adobe created the PDF format, so the early favorites for compatibility were Adobe InDesign CS2 and Adobe FrameMaker 7, but Microsoft Office 2000. joined the club, too. It wasn’t until about 2005 that Adobe introduced tags to PDF files. It wasn’t until 2012 that PDF/UA, the International Standard for accessible PDF was first published as ISO 14289–1 to establish normative language for accessible PDF technology. ISO stands for International Organization for Standardization., which was formed in 1947 to develop and publish standardization in all technical and nontechnical fields other than electrical and electronic engineering. Since its inception, PDF has evolved to include several different standards set by ISO. There are even specific ISO standards for the type of JavaScript and XML used in PDF.

**Slide 7**

**The PDF Association**

In 2006, a global organization called the PDF Association was founded in Berlin, Germany whose mission is “to promote open standards-based electronic document implementations using PDF technology through education, expertise and shared experience for stakeholders worldwide.” In April 2021, the PDF Association published the Matterhorn Protocol, which identifies all possible ways to fail the PDF/UA standard, which is the accessibility standard. It consists of a set of 31 checkpoints comprised of 136 failure conditions.

Many of you are familiar with the Web Content Accessibility Guidelines (WCAG). The W3C has come up with a way to meet WCAG 2.0 standards for PDFs at [https://www.w3.org/TR/WCAG20](http://www.w3.org/TR/WCAG20-TECHS/pdf)-[TECHS/pd](http://www.w3.org/TR/WCAG20-TECHS/pdf)f. Some look to WCAG exclusively, but it fails to recognize many of the syntax requirements for PDFs. We’ll get more into those a little later.

**Slide 8**

**Reputation of PDFs**

I couldn’t find any recent studies of blind users’ opinions of the PDF format, but in my 15 years as an accessibility SME I have heard many groans from screen reader uses and other SMEs about PDFs. Many would like to blame Adobe, but as I started earlier, the format has been an open standard since 2008. Many programs have the ability to create PDFs.

Browsing JAWS user groups, I’ve found instances where some users rather try to copy PDF content to a text editor like Note Pad and read it from there. Some have accused Adobe of discriminating against blind users because author-based features such as password protection, restrictions to copy content, and copying content for accessibility are a direct afront. I never explored Acrobat’s security features before so I tried this out. It’s true: If you password protect a PDF, people cannot copy the content and paste it in another application.

**Slide 9**

**PDFs in Academia**

Link to paper brief

Full text PDF (ironically, the PDF fails Acrobat’s accessibility checker)

You are probably aware of problems with PDFs in academia. Whether it’s a school teacher or college professor posting a PDF, many of these files are not understandable to people with disabilities.

In 2007, Dr. Jonathan Lazar and colleagues published a study in the International Journal of Human–Computer Interaction, “What Frustrates Screen Reader Users on the Web: A Study of 100 Blind Users.” The study found that inaccessible PDFs was one of several frustrations of student and workplace users.

Lazar and colleagues published a paper in 2020 “PDF Accessibility of Research Papers: What Tools are Needed for Assessment and Remediation?” in which they point out that although many are familiar with WCAG, PDF accessibility is often the “step-child” of digital accessibility.

A report by a company called iText says that there are currently 2.5 trillion PDFs on the web, but it’s unknown how many of them are accessible. According to the PDF Association, iText is one of a handful of applications that successfully converts HTML to an accessible PDF—this depends on how accessible the source HTML is in the first place.

**Slide 10**

# The Future of PDFs

Many have speculated on the future of PDFs. In 2020, Adobe announced a new multi-year vision for the future of the PDF. Some have speculated the PDF format will go away—I think mainly this comes from those who concentrate on mobile accessibility or those who have labeled it the stepchild of accessibility. But I think this is short-sided; if you think dealing with PDFs on an iPhone is frustrating, try saving an HTML with its associated CSS and scripting languages to your mobile device or your computer. The PDF detractors fail to recognize the simplicity of creating PDFs (every browser can save a page as a PDF) and their ability to be used for archiving. In the world of academia, just about every scholarly article is published as a PDF in a journal.

Many accessibility specialists may have dabbled with PDFs in their early days, but have since put them in their rearview mirror. I have been on teams where some colleagues don’t want to work on PDFs. Yes, HTML is easier to get right for accessibility than PDFs. But I don’t think electronic documents are going away. In fact, as it stands now, in some circumstances it’s easier to fix a PDF for accessibility than MS Office or Adobe InDesign files. Try defining table row and column headers in MS Word. You can’t., you can only do column headers. And I have been handed MS PowerPoint files that were easier to fix in Adobe Acrobat than in PowerPoint itself.

**Slide 11**

**What are PDF tags?**

Simply put, PDF tags are elements within a PDF file that enable screen reader users to understand the content. Without PDF tags, PDF files are unintelligible. As I said earlier, it wasn’t until about 2005 that Adobe introduced tags to PDF files. That was the year I first got into accessibility and I can’t say for sure if I tested a PDF with a screen reader back then.

The order of the tags determines the announcement order to screen reader users. The Accessibility Checker, which would need a whole separate presentation, doesn’t tell you if the tags are in the correct order.

**Slide 12**

# Creating PDFs with Tags

It’s easy to create PDFs with tags. MS Office. Adobe InDesign, Adobe Experience Manager to name a few of the apps that can produce tagged PDFs. If the source documents are properly formatted or coded, then there’s a good chance the PDFs will be OK.

It’s even easier to create PDFs without tags. Adobe Illustrator, Adobe Photoshop. Scanning a piece of paper on a scanner and not using optical character recognition—and even so, OCR software often messes things up. There are more ways to create untagged PDFs than tagged.

* Funny thing about tags, however, is your file may have them, but they might not be right. You will need to be an accessibility subject matter expert to find out if they are right or wrong.
* It is possible to add tags to a PDF that doesn’t have them. It’s also very possible to edit existing tags with Adobe Acrobat Professional or CommonLook. However, the real check is to run

through a PDF with a screen reader. The accessibility checker doesn’t always tell you if content is tagged correctly.

* The Adobe Acrobat Reader has a “read out aloud” feature but people who are blind don’t use it like their screen readers. Read Out Aloud doesn’t explain semantic information like headings and lists, which screen reader users need to navigate.
* If You Use Word styles properly, PDF tags should be OK, but there are exceptions. I’ve seen a lot of problems with nested lists and tables that extend beyond one page especially. There are other things to mess up tags, for example, someone might put a hard return in between lists to make space. If you do this in a bulleted or numbered list, this breaks the whole sequence.
* If from within Word you save the file as PDF, it can result in an accessible PDF. But a lot depends on the complexity of the Word file. I have seen the resulting PDF missing or messing up things. The PDF will probably need a bit of remediation. The best way I’ve found to convert is to use the Adobe Acrobat plugin for MS Office that comes with Adobe Acrobat Professional.

**Slide 13**

## Some of the Worst Ways to Create PDFs Are From Web Pages…

From a web browser, if you select Save as PDF, Microsoft Print to PDF, or Adobe PDF, you’re more than likely to end up with an untagged PDF. Here we can blame every browser manufacturer. Maybe some day they’ll fix their software,

**Slide 14**

# …But Often the Worst

The Save as PDF button can be the worst because people may assume it creates a tagged PDF. In most cases it doesn’t. The PDF is usually created because of some app behind the scenes on the web server, usually coded in something like Java or JavaScript. There are many free JavaScripts or Java libraries on places like GitHub that convert HTML to PDF and I have yet to find one that creates a tagged PDF. Maybe a dev team didn’t have any other options and found a script years ago and it has been used ever since. But the result could be thousands of inaccessible PDFs created on‐ demand.

You cannot fix the tags in these PDFs because they are created on demand by one of your customers. Sometimes these PDFs are tied into backend systems that pull information from data sources and generate PDFs dynamically

I have been helping a team in commercial banking with customer-facing monthly statements. I’m working with developers to change their coding so that the PDFs they produce are accessible.

This is the reason I wanted to give this presentation. Sorry for taking a while to get here. I don’t think there has been much discussion on machine-generated PDFs.

There has been a lot of discussion on manually created PDFs. I hope others begin to look into this issue and come up with additional solutions. Or at least get the awareness out there. Are the machine- generated PDFs produced by your organization accessible? Is anyone even testing them?

**Slide 15**

**A Quick Search of GitHub**

A quick search of GitHub shows many libraries for converting HTML to PDF, but tagged PDFs are not among them. One company, iText has two libraries for iText7 on GitHub that create tagged PDFs. They're the only one.

**Slide 16**

# PDF Association

* Link to this resource

The PDF Association maintains a list of vendors that support the PDF/UA standard, Some those listed handle accessible HTML to PDF conversion. Many of the apps they list are desktop apps such as ABBYY, which does OCR of scanned documents or axaio, which produces an Adobe InDesign plugin. My company has multiple licenses for iText7, which converts HTML to a properly tagged PDF—this assumes that the HTML is coded correctly. So far, we have successfully deployed iText7 to several apps. I’m not a developer, nor am I getting kickbacks for mentioning iText7—I don’t know how to code or implement Java—but I’ve been able to point our team’s to iText’s documentation. FYI, previous versions of iText do not create tagged PDFs—you need iText7.

If you go to the PDF Association’s site, the list is relatively short. Why? Tagged PDFs have been around since 2005.

Sometimes it takes customer demand to see changes in the industry. Currently, my company is working with Microsoft to correct an Export as PDF function with their Microsoft SSRS (SQL server) product. It also exports to Excel. Both formats have accessibility issues. We’ll see where that goes.

**Slide 17**

# Further Reading

**General PDF History**

* Nearly every accountancy system, sales system, enterprise resource planning, or invoicing software in the world can generate a PDF. https://support.lightyear.cloud/portal/en/kb/articles/what‐the‐heck‐is‐a‐system‐generated‐pd
* PDF was created by Adobe in 1992 to present documents, including text formatting and images, in a manner independent of application software, hardware, and operating systems
* https://en.wikipedia.org/wiki/PDF https://en.wikipedia.org/wiki/History\_of\_PDF

**Future of PDFs**

* ht[tps://www](http://www.techradar.com/news/adobe).techr[adar.com/news/adobe](http://www.techradar.com/news/adobe)‐announces‐biggest‐change‐to‐the‐pdf‐in‐a‐generation
* https://[www.pdftron.com/blog/pdf](http://www.pdftron.com/blog/pdf)‐format/a‐vision‐for‐the‐future‐of‐pdf/
* https://[www.protocol.com/adobe](http://www.protocol.com/adobe)‐future‐of‐pdf

**Vendors that Support PDF/UA**

* https://[www.pdfa.org/supporting](http://www.pdfa.org/supporting)‐pdf‐ua/

**Index of PDF Syntax‐related and ISO Publications**

* https://[www.pdfa.org/index](http://www.pdfa.org/index)‐of‐pdf‐related‐iso‐publications/
* https://[www.pdfa.org/wp](http://www.pdfa.org/wp)‐content/until2016\_uploads/2013/08/PDFUA‐in‐a‐Nutshell‐PDFUA.pdf
* https://[www.pdfa.org/resource/tagged](http://www.pdfa.org/resource/tagged)‐pdf‐best‐practice‐guide‐syntax/

# Thank you!