

Updates from PDF Association accessibility working groups: New examples! & an API mapping

Presented by: Zak Kinsey of TargetStream Technologies



Presentation Overview

- Who am I?
- Who is the PDF Association?
- What are the Working Groups?
 - PDF Accessibility Liaison Working Group
 - PDF/UA Processor Liaison Working Group
- Newly Released Accessibility LWG Atomic Example Files
 - What are they?
 - Where to find them & how to use them
- Ongoing work for the PDF-AAM (PDF Accessibility API Mapping)
 - What is the AAM? Who is it for?
- What is next for the working groups?



Who am I?

- Zak Kinsey, TargetStream Technologies
- I've been working with clients to implement on-demand enterprise-level digital accessible document remediation solutions, via PDF/UA, for almost 10 years.
- Contributing member and author in PDF Association Accessibility LWG & PDF/UA Processor LWG
- Chair of PDF Association PDF/UA Marketing Working Group
- Co-chair of PDF/UA Processor LWG
- Author and founding member of PDF-AAM task force through W3C.



Who is the PDF Association?

- We are an international collaboration of member organizations and individuals engaged in a vendor-neutral platform for developing open specifications and standards for PDF technology. Together, we are driving the world's digital document format into the 21st century and beyond.
- **Mission Statement:** Delivering a vendor-neutral platform for developing open specifications and standards for PDF technology.



Who is the PDF Accessibility Liaison Working Group?

The PDF Accessibility Liaison Working Group (LWG) was originally formed to continue the work started at the December 2018 PDF Techniques Accessibility Summit; producing industry-supported example PDF files demonstrating techniques for achieving accessible PDF.

Today the PDF Accessibility LWG meets weekly to review example “pass” and “fail” PDF files (“techniques”), and develop appropriate metadata.

Participation in this LWG is open to all PDF Association members, accessibility professionals and end users. We strive to accommodate users with disabilities in all our communications, meetings and processes.



Making PDF Accessible

Unlike HTML, which starts as text, PDF starts from a digital canvas. Inherently the digital canvas itself does not contain anything accessible, much like the ink on a printed page does not. The ways in which we must approach remediating PDF to accessible PDF/UA requires a completely different approach. Simply put we ADD accessibility TO the PDF in order to make it and its contents therein accessible.

The following PDF Techniques for accessibility are intended to help educate and guide. To, if I might, make accessible PDF/UA....more accessible!



Techniques for Accessible PDF

WHAT THEY ARE

- Example PDF files, each demonstrating a single aspect of accessibility (for PASS cases) or inaccessibility (for FAIL cases)
- Test procedures to check compliance.
- PASS cases conform to PDF/UA (ISO 14289) in all particulars, while FAIL cases intentionally do not.
- These examples are “atomic” in the sense that they contain only the content necessary for demonstrating that specific aspect of the Technique.

In general terms, they are intended to provide practical examples of the various requirements of ISO 32000 and ISO 14289, as well as indicate some (not all!) of the ways in which these standards are commonly violated in PDF files that are not accessible.



Techniques for Accessible PDF

WHAT THEY ARE NOT

These Techniques do not provide instructions for the use of a certain software. They do not explain how to make accessible PDF files or remediate (apply corrections to) existing PDF files. Although the files are valid PDF files, they are merely atomic examples, and as such, are not “real-world” files.



What all is in a Technique?

- Plain language description of the concept or tagging structure
- Test procedures on how to check for compliance.
- Atomic PDF example file demonstrating the technique.
 - For a given technique there may be multiple pass and fail example files.
- Matterhorn Protocol Checks
- Applicable WCAG Success Criteria
- Technique Finder Code



Technique Finder Code

Each Technique will be posted on an individual page on pdfa.org. In addition, the PDF Association is introducing the "Accessibility Technique Finder", a string of characters that uniquely identifies the Technique to search engines.

The purpose of the Technique Finder is to help users to rapidly determine whether a given vendor or software package has stated their support for a given Technique. Users can simply search for the vendor or product name and a specific Accessibility Technique Finder code in order to locate web pages documenting the vendor's approach to that specific technique.

Accessibility Technique Support Finder

The Accessibility Technique Support Finder allows you to quickly locate software and services that claim to support a given Technique. Simply search the internet with a technique finder code together with the name of your product.

The technique finder code for this Technique is: **UA1_Tpdf-F2_07**

NOTE: the "technique support finder" concept was introduced in November 2024; please allow time for adoption.



Why provide fail examples?

- To provide clear contrast between conforming and non-conforming tagging structures
- FAIL cases can be used to confirm improper tagging structures.
- Not all fail cases in PDF/UA are failure criteria in WCAG terms
 - In such instances it will be clear on the difference between PDF/UA and WCAG Success Criteria
- To provide advisory techniques that fall between PDF/UA and WCAG Success Criteria
- To clearly demonstrate some of the most common ways in which elements are improperly tagged.



Who are these techniques for?

- Remediators
- Software developers
 - Remediation tools
 - Validation tools
 - AT Software
- Procurement teams/officers



5 Fundamentals of tagged PDF

1. Basic Technical Rules
2. Text
3. Content
4. Logical Content
5. Appropriate Semantics

The Accessibility LWG has identified these 5 fundamental concepts that apply to all tagged PDF. Conforming to these fundamentals provides a strong foundation to creating not only compliant tagged PDFs, but to also ending with a PDF/UA document that achieves the goal of an accessible digital document.



Fundamental 1: Basic technical rules are followed

Software that creates a PDF must follow basic technical rules for tagged PDF, so that other software can process the PDF for accessibility.

PDF's rules are defined in the respective PDF specification documents, including ISO 32000, the PDF specification, and ISO 14289 (PDF/UA), the PDF specification for accessible PDF documents, which adds requirements to those of ISO 32000.

A PDF/UA-compliant file is also required to conform to ISO 32000. In general, basic technical rules can be unambiguously checked by software commonly known as a “validator”.



Fundamental 2: Text content is machine-readable

Most information in typical documents is textual (visible text, alternative text or bookmarks). In order to understand the challenges related to machine-readable textual information in PDFs, it is necessary to distinguish between text content and non-text content intended to be consumed as text.

Requirements for text content:

- Each character has a corresponding Unicode value
- All text has a specified language

Non-text content intended to be consumed as text will have an equivalent text content using one of the following mechanisms:

1. Creating invisible text (by using OCR tools, for example)
2. Adding ActualText to a Span marked content sequence
3. Adding ActualText to a tag



Fundamental 3: Real content and Artifacts are distinguished

Not all content in a PDF document is essential for understanding. To enable software to distinguish between relevant content (e.g., headings and paragraphs) and non-relevant content (e.g., decorative lines or running headers), PDF files provide the option of distinguishing relevant content, known as real content, from the rest, known as artifacts. In accessible PDF documents, content is marked as either real content or artifact, and can never be both at the same time.



Fundamental 4: Logical Content Order

In addition to appropriately tagging real content, the tags must appear in the structure tree in an appropriate order to enable correct presentation by software (including assistive technology). This is referred to as the logical content order. The logical content order must reflect the order of real content as intended by the author.



Fundamental 5: Appropriate Semantics

In order for real content to be correctly understood by a user, the most semantically appropriate tag must be used, and each unit of real content must be tagged with a single appropriate tag, even if parts of the real content are visually separated.

Example of a semantically appropriate tagging:

- A list is tagged using L, LI, Lbl, and LBody tags

Example of a semantically inappropriate tagging:

- A single paragraph is tagged using two or more P tags



How remediators can use the techniques


For remediators these techniques, and especially the example files, provide a clear example of what is and is not conformant for a given technique. The plain language description aids in educating the intent of the specifications for real world application into a PDF/UA file.

Then, the example files allow the remediators to interact with and view an example PDF/UA file that demonstrates the technique to further educate. The example files can be used for comparison and guidance to remove any confusion about what was meant, or intended, by the specifications.



How developers can use these techniques

Developers will find the example files and test procedures a particular boon for use in the development of their software. Whether the software creates or remediates PDF/UA, or consumes it, developers can leverage the resources provided by these techniques to develop and improve their software to provide the intended benefits for the creation and use of an accessible PDF.



How procurement can use these techniques

Vendors can declare their conformance to the PDF/UA Accessibility Techniques allowing procurement teams to purchase accessibility tools/software with a deeper level of understanding of its capabilities to ensure their purchase(s) meet their requirements.

Screenshots of example files

The image displays two side-by-side screenshots of an accessibility tool's 'Accessibility tags' window. Both screenshots show a document structure with a paragraph of text: "This is the paragraph.", where "paragraph." is highlighted with a purple box. The text "This is the" is also visible above the highlighted text.

Left Screenshot (Pass): A green box labeled "Pass" is positioned above the text. Two green arrows point from this box to the accessibility tag for the highlighted text. The tag is identified as ``. The "Actual Text" field for this tag contains the text "paragraph.". The "Type" field is set to "Span".

Right Screenshot (Fail): A red box labeled "Fail" is positioned above the text. Two red arrows point from this box to the accessibility tag for the highlighted text. The tag is identified as `<Figure>`. The "Actual Text" field for this tag contains the text "paragraph.". The "Type" field is set to "Figure".



Accessibility LWG 2025 Roadmap

- Initial release of the fundamental techniques is imminent!
 - Keep an eye on pdfa.org for the announcement of their publication!
- Plan to release techniques on a quarterly basis
- Next technique topics include:
 - Lists
 - Headings
 - Table of Contents
- Engage with the user base through commenting on GitHub to obtain feedback and adapt to the needs and interests of all of you!
- Continuous improvement of the [glossary of accessible terminology](#)



Who is the PDF/UA Processor Liaison Working Group?

Improving accessibility support for PDF documents means improving the way PDF viewers, other PDF processors and assistive technology (AT) handle tagged PDF.

To support this objective the PDF/UA Processor LWG was created at the request of ISO TC 171 SC 2 WG 9, the ISO working group responsible for development of PDF/UA.

This Liaison Working Group is a meeting-place for the PDF and assistive technology industries to help drive the formalization of common understandings on processing PDF files that conform to ISO 14289 (PDF/UA), the standard for accessible PDF documents.



Who is the PDF/UA Processor Liaison Working Group? (Continued)

Processor requirements will be developed in a separate dedicated specification, the work of this LWG.

Critical to this work is the involvement of developers who focus on assistive technology, including accessibility APIs. For this reason the Board of Directors has decided to make it possible for non-members with tangential interest in PDF technology to participate in the committee's online discussions, meetings and development of the draft specification.



What is an Accessibility API Mapping?

Accessibility APIs make it possible to communicate accessibility information about user interfaces to assistive technologies. This information includes:

1. Descriptive properties (role, name, value, position, etc.)
2. Transient states (pressed, focused, etc.)
3. Events (text changed, button was clicked, checkbox was toggled)
4. Actions the user might take (click, check/toggle, drag, etc.)
5. Relationships (parent/child, description/described object, previous object/next object, etc.)
6. Textual content



Which APIs are covered?

- MSAA with IAccessible2 1.3 [IAccessible2]
- User Interface Automation [UI-AUTOMATION]
- ATK - Accessibility Toolkit [ATK] and Assistive Technology Service Provider Interface [AT-SPI], referred to hereafter as "ATK/AT-SPI"
- macOS Accessibility Protocol [AXAPI]

What is an AAM role mapping? HTML-AAM button entry as an example

§ 3.5.20 button

HTML Specification	button
[wai-aria-1.2]	button role
Computed Role	Use WAI-ARIA mapping
MSAA + IAccessible2	Use WAI-ARIA mapping
UIA	Use WAI-ARIA mapping
ATK	Use WAI-ARIA mapping
AX	Use WAI-ARIA mapping
Comments	A button's mapping will change if the aria-pressed or aria-haspopup attributes are specified.



PDF Accessibility API Mappings (PDF-AAM)

The PDF-AAM will define how user agents map PDF/UA structure elements and attributes to platform accessibility application programming interfaces (APIs).

Documenting these mappings promotes interoperable exposure of roles, states, properties, and events implemented by accessibility APIs and helps to ensure that this information appears in a manner consistent with author intent.

The PDF-AAM is a part of the WAI-ARIA suite and will be published by the W3C



PDF-AAM Roadmap & Timeline

Currently the task force for the project has recently kicked off. At this time we do not have a deadline for the initial release of the PDF-AAM, however we have a goal to have the first iteration released within 2025 to begin gathering feedback from the community.



PDF Association PDF/UA Processor LWG 2025 Roadmap & Timelines

- Produce the initial release of the PDF-AAM
- Begin testing the atomic sample documents from the Accessibility LWG through known AT software and reporting the results as it relates to the PDF/UA Techniques.
 - The Technique Finder Code will be useful here in reporting.



Questions?

Thank you for attending my presentation. Please don't hesitate to ask questions at this time!