

The Tag Editor User Guide.

Open, edit, verify, apply — and how things should be tagged along the way.

Automation does the heavy lifting. You get the final say.

When you upload a PDF, AccessiblePDF auto-tags the structure, repairs reading order, generates alt text, and OCRs scanned pages — then validates everything against PDF/UA. Most issues are fixed before you ever see them. But some calls need a human: alt-text nuance, tricky tables, reading-order judgment. The *Tag Editor* is where you make those calls, guided by plain-English findings, without ever opening Acrobat.

What's a "tag," anyway? A tag is an invisible label on a piece of content — `<H1>` for a top heading, `<Table>` for a table, `<Figure>` for an image. Screen readers don't see your layout; they walk the tag tree, in order, and announce what each tag claims to be. Tagging a document correctly is what makes it *accessible* — and it's what standards like WCAG 2.1, PDF/UA-1 and Section 508 audit.

Contents

1 · Opening a document	3
2 · Finding your way around — the three panels	5
3 · Editing the structure	6
4 · The Properties panel	8
5 · Issues, Run checks & the Reader preview	10
6 · How things should be tagged — the reference	11
Headings, paragraphs & reading order	11
Figures, alt text & artifacts	12
Tables	13
Lists, links, footnotes & forms	14
7 · Apply to PDF	15
8 · Keyboard shortcuts & pre-flight checklist	16

SECTION 1

1 • Opening a document

From the Tag Editor page there are two ways in: **drop any PDF on the upload panel** (it opens immediately, with a free compliance check running in the background so you have a baseline), or **open a document you've already processed** from the list below it.

If the file already has tags

You'll see this choice the moment the document opens:

This PDF already has tags

We found **316 tags** in this document. Most people keep them and fix what's wrong. Rebuilding throws them out and drafts a fresh structure from the document's text — your PDF file itself is never changed either way.

Rebuild from scratch

Keep them (recommended)

Keep them (recommended) — the right choice for files our automation already remediated, or files tagged in another tool that mostly got it right.

Rebuild from scratch — for when the existing tags are junk (a scanner's auto-tags or an old export).

This is the only question the editor asks; starting over never opens a second one.

1 · Opening a document

If the file has no tags

Untagged files are **auto-drafted**: the editor reads the text layer column-by-column, guesses headings from font size, and builds a starter structure. Treat it as a draft, not a verdict — expect to review every role and check the reading order.

Scanned or unreadable text layer?

If a page is a scan — or has text whose positions are garbage (common in bad OCR) — a banner appears with **Redraft text with OCR**. One click rebuilds the text layer on our servers with real coordinates and reloads the document so tags and region boxes claim real words. Your original file is never modified. Takes a minute or two.

⚠ This page has no usable text layer.

Redraft text with OCR

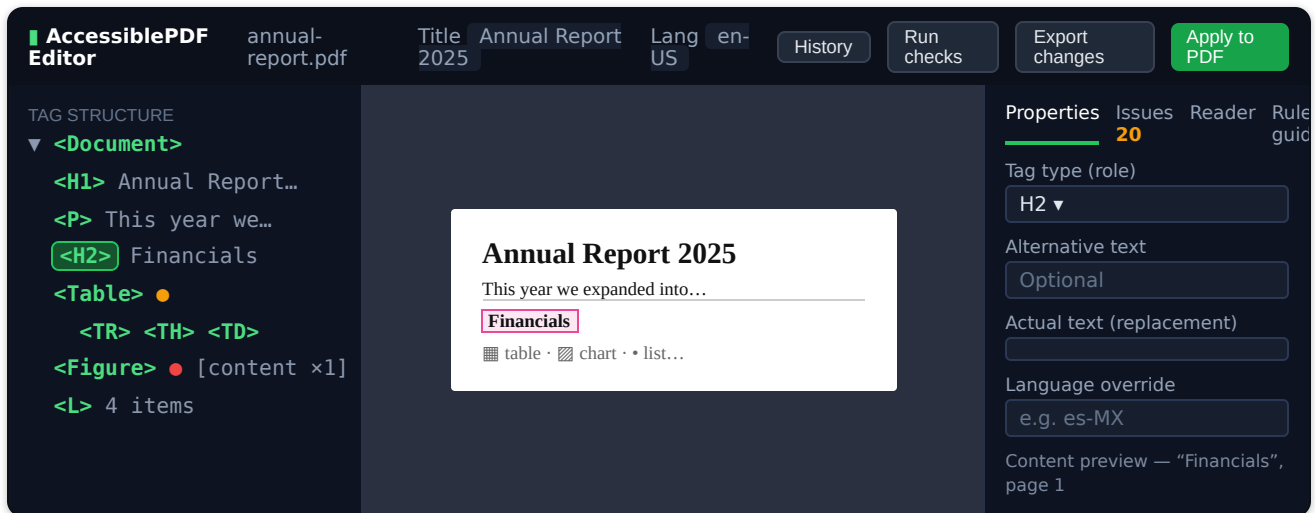
Your work saves itself

- ✓ Every edit is saved **locally in your browser** instantly.
- ✓ Sessions sync **to your workspace cloud** — watch for ✓ **saved to cloud** in the status bar.
- ✓ Reopen the document later on any machine and the editor offers to **resume exactly where you left off**.
- ✓ The **History button** (top right) lists snapshots of your tag structure: one is saved automatically before every Apply, and you can save one manually any time. Restore any snapshot from the last 30 days — restoring saves your current state first, so nothing is ever lost.

SECTION 2

2 • Finding your way around

The editor is three panels. Left is *what the document claims to be*, center is *what it looks like*, right is *everything about the current selection*.



① **Tag structure** — the tree a screen reader walks, top to bottom. **Tree order is reading order.** Colored dots flag tags with errors (red) or warnings (amber).

② **The live page** — click any content to select its tag; click a tag to highlight its content in pink. Right-click either side for the full action menu.

③ **Right panel** — **Properties** edits the selected tag, **Issues** lists validation failures (click one to jump to it), **Reader** previews what a screen reader says, **Rules guide** is the built-in rulebook.

Don't skip the header bar. Document **Title** and **Language** sit at the top for a reason — they're the two most common instant-fail checks in any audit. Set the title to something human ("2025 Annual Report", not "final_v3_FINAL"), and the language to the document's primary language, e.g. en-US.

3 · Editing the structure

Change what something is

Select a tag, then pick a new type in **Properties** → **Tag type**, or right-click for quick chips (P, H1, H2, Figure, Caption, Artifact and the full set). Fastest of all is the keyboard: **1-6** retypes the selection to H1-H6; **P F A C S T** retype to Paragraph, Figure, Artifact, Caption, Sect and Table.

Reorder & re-nest — the tree order IS the reading order

Move tags with the **↑ / ↓** buttons (or **Alt + ↑/↓**), promote and demote with **↶ / ↷**, or simply **drag a tag** and drop it before, after, or *into* another — the drop zone highlights as you hover. Multi-select with **Shift**-click (range) or **Ctrl**-click (toggle) to move whole groups at once.

Split & merge

Auto-drafts sometimes glue two paragraphs into one tag, or shatter one paragraph into five. Right-click a paragraph → **“Split into N lines”** / **“Split at this line”**, or take the **✂** tool and click exactly where the break belongs. To merge, **Ctrl**-click the fragments and press **Ctrl + M** — or drag across them with the **🔲** merge tool.

Split into words — for anything a line split can't separate

Some documents run a bold lead-in and its paragraph together on one line (“Budget visibility. The department operates...”). Right-click the tag → **Split into words**: every word becomes its own tag carrying its true position on the page. **Ctrl**-click the words that belong together and press **Ctrl + M** — the lead-in becomes its own tag claiming exactly its own text (press **1-6** to make it a heading), and the rest merges back into the paragraph. Nothing ever claims text it doesn't cover.

3 · Editing the structure

The canvas tools



Select

Click a tag; drag a region to open the region menu



Figure tool

Drag a region — tagged as Figure instantly



Artifact tool

Drag a region — marked decorative instantly



Split tool

Click inside a paragraph to split at that line



Merge tool

Drag across tags to merge; or click several, then Enter

The sweeper — fastest first cleanup

The sweeper scans the whole document for **repeated headers, footers and page numbers** and artifacts them all in one confirmable step. On a long document, run it first — it routinely clears dozens of issues in one click.

auto-fix

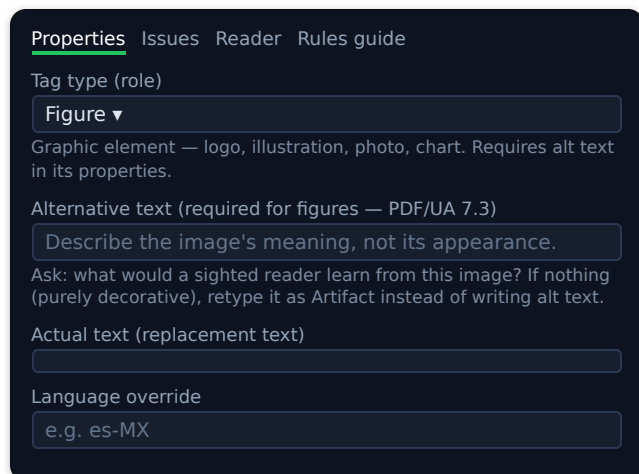
With **auto-fix** on (the checkbox above the tree), dropping stray tags into a `<Table>` or `<L>` wraps them into proper `<TR>/<TD>` or `<LI>` structure automatically. **“Conform children”** in the right-click menu does the same for an entire container in one go.

Relax — everything is undoable. `Ctrl + Z` gives you 60 levels of undo, including bulk actions like the sweeper.

SECTION 4

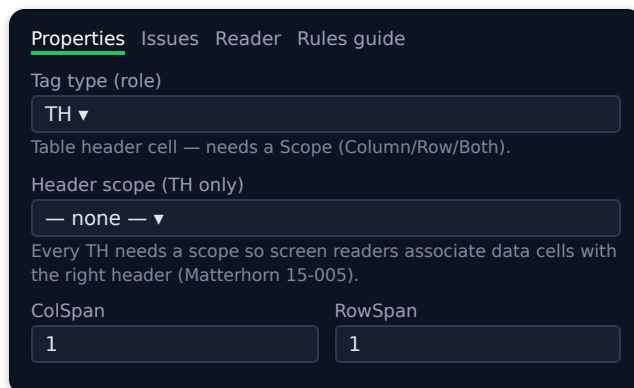
4 • The Properties panel

Whatever you select, Properties shows exactly the fields that matter for that role — and tells you what the standard expects.



The screenshot shows the 'Properties' panel for a 'Figure' tag. The 'Tag type (role)' dropdown is set to 'Figure'. Below it, a description states: 'Graphic element — logo, illustration, photo, chart. Requires alt text in its properties.' The 'Alternative text (required for figures — PDF/UA 7.3)' field contains the text 'Describe the image's meaning, not its appearance.' Below this is a note: 'Ask: what would a sighted reader learn from this image? If nothing (purely decorative), retype it as Artifact instead of writing alt text.' The 'Actual text (replacement text)' field is empty. The 'Language override' field contains 'e.g. es-MX'.

Selecting a **Figure** — alt text front and center.



The screenshot shows the 'Properties' panel for a 'TH' tag. The 'Tag type (role)' dropdown is set to 'TH'. Below it, a description states: 'Table header cell — needs a Scope (Column/Row/Both).' The 'Header scope (TH only)' dropdown is set to '— none —'. Below this is a note: 'Every TH needs a scope so screen readers associate data cells with the right header (Matterhorn 15-005).' The 'ColSpan' field contains '1' and the 'RowSpan' field contains '1'.

Selecting a **TH** — scope and spans appear.

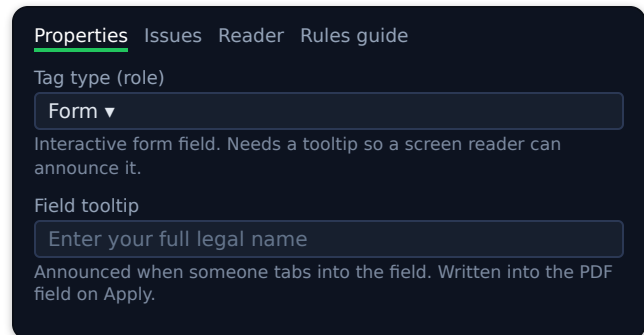
The fields, in plain English

Field	What it does
Tag type (role)	What this content <i>is</i> . Changing it here is identical to retyping via keyboard or right-click.
Alternative text	What a screen reader speaks for a Figure. Required on every Figure (PDF/UA 7.3). Describe meaning, not appearance.
Text in this tag / Actual text	Properties pulls the full text from inside the selected tag. Leave it unchanged and the page text is used as-is. Edit it — fix an OCR misread, correct wording — and your version is stored as <i>Actual text</i> : exactly what a screen reader says after Apply. Put it back to match the page and the override clears itself.
Language override	Set only when this element's language differs from the document's — e.g. a French quotation inside an English report gets fr-FR.
Header scope (TH)	Column, Row, or Both — which cells this header governs.
ColSpan / RowSpan (TH/TD)	How many columns/rows a merged cell covers. Must match the visual table.

4 • The Properties panel

Form fields — see them, describe them, place them

Every fillable field shows as a **blue clickable box** on the page (green with a ✓ once described). Click a field: it links into the tree as a **<Form>** tag at its spot in the reading order, and Properties opens straight into the **Field tooltip** — what a screen reader announces when someone lands in the field. Write “Enter your full legal name”, not “Text1”. Drag the Form tag in the tree if the field belongs elsewhere in the reading order. On **Apply**, the tooltip is written into the PDF field itself and your order is what ships.

The image shows a dark-themed 'Properties' panel. At the top are tabs: 'Properties' (selected), 'Issues', 'Reader', and 'Rules guide'. Below the tabs is a section 'Tag type (role)' with a dropdown menu showing 'Form' with a downward arrow. Underneath is a text description: 'Interactive form field. Needs a tooltip so a screen reader can announce it.' Below that is a section 'Field tooltip' with a text input field containing the placeholder text 'Enter your full legal name'. At the bottom is a text description: 'Announced when someone tabs into the field. Written into the PDF field on Apply.'

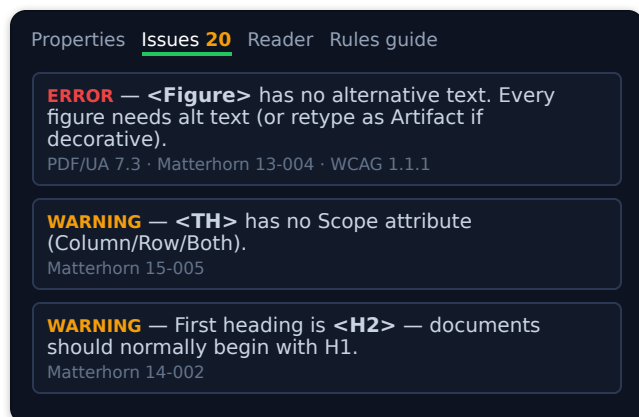
Selecting a **Form** field — tooltip front and center.

5 • Issues, Run checks & the Reader

The Issues tab

Two things live here: a **headings outline** of the whole document (a quick sanity check that your H1→H2→H3 story makes sense), and the **validation findings** — veraPDF/PAC-style results translated into plain English, each citing the exact clause your auditor would (Matterhorn, PDF/UA, WCAG).

Click any issue to jump straight to the offending tag. Fix it, and move to the next.



Run checks

The **Run checks** button re-validates the document against ~25 PDF/UA & PAC-style rules — heading order, missing alt, table headers, list structure, empty tags and more. Run it after every meaningful editing pass; the status bar keeps a live count (“7 errors · 13 warnings”).

The Reader tab — hear your document

Reader is a **screen-reader simulation**: a linear preview of exactly what assistive tech will announce, with real speech (► **Read aloud**) and NVDA-style navigation:

Key	Moves to
↓ / ↑	Next / previous item
H · 1 – 6	Next heading · heading by level
G · T · L · K	Next graphic · table · list · link

Artifacts are skipped — exactly as a real screen reader would skip them. **If it reads wrong here, it reads wrong for your users.**

A passing checker is necessary, not sufficient. Automated rules can verify that alt text *exists* — not that it's *true*. Before certifying a document, do one full pass in the Reader tab, and ideally one in a real screen reader like NVDA.

6 • How things should be tagged

This is the judgment-call reference: for each kind of content, what correct tagging looks like, and the mistakes we see most. Everything here is also in the editor's **Rules guide** tab, one click from where you're working.

Headings — the document's skeleton

✓ CORRECT

```
<H1> Annual Report
  <H2> 1. Financials
    <H3> 1.1 Revenue
  <H2> 2. Operations
```

✗ WRONG

```
<P bold 18pt> Annual Report  ← styled, not tagged
<H3> 1. Financials          ← skips H1-H3
<H1> 1.1 Revenue           ← level ≠ importance
```

- ✓ One **<H1>** for the document title; start the hierarchy there and **never skip levels** downward (H1→H2→H3...).
- ✓ If it *looks* like a heading but is tagged **<P>**, screen-reader users can't navigate to it — retype it (**1**–**6**).
- ✓ Pick the level by **position in the outline**, not by how big the font is.
- ✓ Sanity-check the whole outline in **Issues → Headings outline**.

Paragraphs, spans & inline content

- ✓ Body text is **<P>**. Bold/italic runs inside it usually need no tag of their own (**** is fine, and harmless).
- ✓ A quotation set off from the text is **<BlockQuote>**; a quotation inside a sentence is **<Quote>**.
- ✓ Foreign-language runs get a **Language override** on their span — « la qualité avant tout » in an English report should be read with French pronunciation, not mangled.
- ✓ One visual paragraph = one **<P>**. Auto-drafts sometimes split at line breaks (merge: **Ctrl** + **M**) or glue columns together (split: **⌘**).

Reading order — the tree top-to-bottom

- ✓ The tag tree **is** the reading order. If the tree says caption-before-figure, that's what gets read.
- ✓ **Multi-column pages:** the order must run all the way down column one, then start column two. Geometric top-to-bottom interleaving is the classic failure — check these pages in the Reader tab.
- ✓ Sidebars and pull-quotes go where they make sense in the *story*, not where they sit on the page.
- ✓ Fix order by dragging tags in the tree or with **Alt** + **↑/↓**.

6 · Figures, alt text & artifacts

Every image is one of exactly two things

It carries meaning → **<Figure>** + alt text.

Charts, photos, diagrams, logos that identify the sender. The alt text is what a screen-reader user gets *instead of* the image — it's required (PDF/UA 7.3).

It's decoration → Artifact. Divider flourishes, background textures, watermarks, repeated page furniture. Artifacts are skipped entirely by assistive tech — which is exactly what you want. No alt text needed, ever.

Writing alt text that's actually useful

✓ SAYS WHAT THE IMAGE MEANS

"Bar chart: revenue grew each quarter of 2025, from \$4.2M in Q1 to \$8.3M in Q4."

✗ SAYS WHAT THE IMAGE LOOKS LIKE

"Chart."
"Blue bar graph image, 700 by 450 pixels, chart.png"

- ✓ Ask the editor's own question: *what would a sighted reader learn from this image?* Write that. If the answer is "nothing," it's an Artifact — retype it (**A**) instead of writing alt text.
- ✓ Keep it to a sentence or two. Put long descriptions (complex diagrams, formulas) in nearby body text and reference them.
- ✓ Don't start with "Image of..." — the screen reader already announces it as a graphic.
- ✓ A chart's alt text should state the **takeaway** ("costs fell 12%"), not enumerate every data point.
- ✓ If a **<Caption>** already tells the story, the alt text can be brief — don't read the same sentence twice.

In the editor

1. Untagged images show a purple **+ Figure** chip on the page — click it to tag the image, then write the alt text in Properties.
2. Any region can become a figure or artifact by **dragging**: **A** tags the region as Figure instantly, **D** marks it decorative.
3. Run the **sweeper** once per document — it finds repeated headers, footers and page numbers on every page and artifacts them in one confirmable batch.

Scanned & image-only pages

A page that is one big picture of text has **nothing for a screen reader to read**. AccessiblePDF's OCR normally rebuilds the text layer during remediation. If a page slipped through — or its text layer is unusable — click **Redraft text with OCR** in the editor's banner; for a page that's genuinely an exhibit, tag the page image as a **<Figure>** whose alt text summarizes the content ("Scanned 1987 memorandum instructing departments to archive pre-1980 records...").

6 • Tables

A screen-reader user hears a table one cell at a time — "row 3, column 2: 4,900." The only thing connecting that number to "*Spectrometers, FY 2025, Units*" is your header structure.

The anatomy

✓ CORRECT

```
<Table>
  <TR>  <TH scope=Column> Region
        <TH scope=Column> Manager
  <TR>  <TD> Mountain West
        <TD> D. Okafor
```

✗ WRONG

```
<Table>
  <TR> <TD bold> Region ← header styled
        <TD bold> Manager    as data
<P> Mountain West  D. Okafor
    ← "table" faked with tabs
```

- ✓ Header cells are **<TH>**, data cells are **<TD>** — visual bolding doesn't count.
- ✓ **Every TH needs a Scope** — Column for the header row across the top, Row for row headers down the left, Both when it governs both (Matterhorn 15-005). Set it in Properties.
- ✓ Merged cells must declare their size: a header spanning two columns gets **ColSpan 2**; one covering two rows gets **RowSpan 2**. The numbers must match the visual table exactly.
- ✓ Only *data* gets a Table tag. Content merely *arranged* in a grid (a form layout, a photo wall) should not be a Table — it forces row/column announcements on things that aren't tabular.
- ✓ Give the table a **<Caption>** when one exists on the page, as the first child of the Table.

Fixing tables fast in the editor

1. Select the table on the page — the tree jumps to it, and each **<TH>** missing a scope carries an amber dot.
2. Walk the header cells and set **Header scope** in Properties (Column across the top row; Row for the leftmost column of a matrix table).
3. Check merged cells: select each and confirm ColSpan/RowSpan match the visual span.
4. Dropped a stray row or paragraph into the table? With **auto-fix** on it wraps into proper **<TR>/<TD>**; or right-click the Table → **Conform children**.
5. Finish with the **Reader tab**: press **T** to jump to the table and arrow through a few cells — the headers should be announced with each data cell.

Complex-table rule of thumb: if you can't say in one sentence which header governs a cell, a screen reader can't either. Consider splitting a monster table into two simpler ones — it's usually better for everyone.

6 • Lists, links, footnotes & forms

Lists

✓ THE PATTERN

```
<L>
  <LI> <Lbl> 1.
    <LBody> First item...
  <LI> <Lbl> 2.
    <LBody> Second item...
<L> ... nested sublist ...
```

- ✓ **<L>** holds **** items; inside each, **<Lbl>** is the bullet/number and **<LBody>** is the content.
- ✓ A sublist nests as an **<L>** *inside* the parent **** — never as a sibling.
- ✓ Bullet-looking paragraphs aren't a list until they're tagged as one — screen readers announce "list, 5 items" only for real **<L>** structure. Drop stray items into an L with auto-fix on.

Links

- ✓ Every link is a **<Link>** tag wrapping the link text **plus an** **<OBJR>** — the object reference connecting the tag to the clickable region. A Link without its OBJR looks fine and fails the audit.
- ✓ Link text should make sense on its own: "the compliance standard" beats "click here" — screen-reader users often browse a list of links out of context.

Footnotes

- ✓ The superscript citation in the body is a **<Link>** (so users can jump); the note text itself is a **<Note>** (or a P inside a Note/Div at the page bottom).
- ✓ The note body should sit in the reading order where it's referenced *logically* — not interrupt a sentence mid-column.

Form fields

- ✓ Each field widget is wrapped in a **<Form>** tag with its **<OBJR>**.
- ✓ Every field needs a **tooltip** (the /TU description) — it's what a screen reader announces when the user lands in the field. "Enter your full legal name", not "Text1".
- ✓ The visible label next to the field stays regular text — the tooltip covers the announcement.
- ✓ Radio buttons in one choice share one field name (one question, N widgets).
- ✓ In the editor, all of this is point-and-click: click the field's blue box, type the tooltip, drag its tag into reading position. **Apply** writes it into the PDF.

Tables of contents

- ✓ A real TOC is **<TOC>** containing **<TOCI>** items, each holding a **<Link>** to its section. A TOC tagged as a plain list still works — but TOC/TOCI is what PDF/UA expects for navigation.

When in doubt, open the Rules guide tab. Every topic on these pages has a longer article inside the editor — with the exact Matterhorn/PDF-UA clause numbers your auditor will cite — searchable, one click from the tag you're fixing.

SECTION 7

7 • Apply to PDF

Editing happens in a live working copy. Nothing touches your PDF until you say so.

1. **Run checks** one last time — aim for zero errors, and read any remaining warnings deliberately.
 2. Hit **Apply to PDF**. We rebuild the document's tag structure server-side from your edits — creating the underlying marked content even where the original never had any — set the bookmarks and metadata (PDF/UA identifier included), and re-run full validation.
 3. Download the fresh file — `yourfile.edited.pdf` — from the job page, PAC report alongside.
- ✓ **Apply runs are included** in the Editor subscription — no per-page charges. Iterate as much as you like.
 - ✓ Each new apply **supersedes the previous version** automatically; your document list shows the original and the latest, not every attempt.
 - ✓ **Export changes** (next to Apply) downloads your edit log — handy for reviews and audit trails.
 - ✓ A **version snapshot** is saved automatically before every Apply — the **History** button restores any snapshot from the last 30 days if you ever want to walk an apply back.

Before you certify — the pre-flight checklist

Check	Where
Document Title and Language set	Header bar fields
Headings outline tells a sensible story, starts at H1, no skipped levels	Issues tab → Headings outline
Zero errors; every remaining warning consciously accepted	Run checks / status bar
Every Figure has alt text that states its <i>meaning</i> ; decorations are Artifacts	Issues tab → figure errors
Every TH has a Scope; merged cells have correct spans	Properties, per header cell
Reading order survives multi-column pages and sidebars	Reader tab, page by page
Headers, footers, page numbers artifacted (🔧)	Sweeper / page view
One full listen of key pages	Reader tab ► Read aloud — ideally NVDA too

SECTION 8

8 · Keyboard shortcuts

Editing

Retype selected tag to H1-H6

1 - 6

Retype to P · Figure ·
Artifact · Caption · Sect ·
Table

P F A C S T

Walk the tree

↑ / ↓

Move tag up / down (reading
order)

Alt + ↑ / ↓

Select a range /
toggle selection

Shift -click / Ctrl -click

Merge selected tags

Ctrl + M

Delete selected tag(s)

Delete

Undo — 60 levels

Ctrl + Z

In the Reader tab

Next heading · heading by level

H · 1 - 6

Next graphic · table · list ·
link

G · T · L · K

Where to get help

- ✓ **Rules guide tab** — the full rulebook, searchable, inside the editor.
- ✓ **Full online guide** — accessiblepdf.app/editor/guide
- ✓ **Support** — support@accessiblepdf.app · we answer fast.

One last word of craft.

A passing checker is necessary, not sufficient. The point of every tag in this guide is a person on the other end of a screen reader having the same experience your sighted readers get. When you're unsure how to tag something, ask what *they* should hear — the right answer usually follows.