

How to Make Your LaTeX PDF Accessible

T. Jia*, J. Roche, K. Chudali.

August 2026

Abstract

Making PDFs accessible ensures that their content can be used by the widest possible audience, including people who rely on assistive technologies such as screen readers. Accessibility is also required for PDFs covered by federal digital accessibility requirements, such as documents made available through a public university's web content. This documentation provides a practical guide to creating scientific PDFs with LaTeX that are accessible to screen-reader users. We focus on technical recipes and practical steps that can be incorporated into a typical LaTeX workflow.

Contents

1	Introduction	3
2	Tools	3
3	Document Setup	3
3.1	Document Metadata	3
3.2	Document Class	4
3.3	Fonts	4
3.4	Mathematics Packages	4
4	Document Structure	5
4.1	Tags	5
4.2	Headings	5
4.3	Paragraphs	5
4.4	Lists	6
5	Figures	6
5.1	Alternative Text	6
6	Tables	7
7	Mathematics	7
8	Validation	8
9	Screen-Reader Testing	8
10	Debugging	9

*Physics and Astronomy Department, Ohio University, tj066824@ohio.edu

A	Test the commonly used formulas	10
A.0.1	Plain-text control	10
A.0.2	Basic mathematical content	10
A.0.3	Right-side superscripts and subscripts	10
A.0.4	Left-side scripts and nuclear notation	10
A.0.5	Greek letters and common physics symbols	11
A.0.6	Fractions, roots, and delimiters	11
A.0.7	Calculus and large operators	12
A.0.8	Functions and operators	12
A.0.9	Matrices, vectors, and systems	12
A.0.10	Accents, decorations, and special scripts	13
A.0.11	Units and mixed text	13

1 Introduction

What is digital accessibility? Digital accessibility is a property of electronic content that allows it to be accessed, navigated, understood, and used by everyone, including people with disabilities. For a PDF, this means not only making the document visually readable, but also ensuring that its text, structure, figures, tables, and other content can be understood and navigated using assistive technologies such as screen readers. Ohio University maintains a comprehensive Digital Accessibility website¹ with additional information, guidelines, resources, and training opportunities, including a course on digital accessibility.

Ohio University requirements All theses and dissertations submitted at Ohio University must meet the Minimum Digital Accessibility Standards (MDAS), which establish the accessibility requirements for electronic documents. Thesis and Dissertation (TAD) Services² checks compliance with these standards as part of the submission process.

Authors are responsible for ensuring that their final PDF fulfills these accessibility requirements. Until now, LaTeX users have typically had to perform some post-processing of the PDF, for example using Adobe Acrobat, to address accessibility issues. This guide presents a different approach: creating an accessible PDF directly from LaTeX, minimizing or eliminating the need for post-processing.

2 Tools

- **Compiler:** LuaLaTeX. In our project, we use Overleaf. Choose File -> Settings -> Compiler-> Compiler-> lualatex. Theoretically, this recipe also applies to any LaTeX editor.
- **Accessibility validator:** veraPDF. veraPDF is a multiplatform, open-source, file-format validator that covers all PDF/A and PDF/UA standards. You can download it from this site: <https://verapdf.org/>. More detailed information on using veraPDF for validation is in section 8.
- **Screen reader:** NVDA. Even though veraPDF tells us if our file works, which parts don't work and why they don't work, it is still important for us to actually listen to the file through screen readers. In our test, we used NVDA. NVDA runs on Windows. Recent versions include MathCAT support for reading mathematical expressions. For older versions, the MathCAT add-on may need to be installed separately. More detailed information about how to use NVDA can be found in section 8.
- **PDF reader:** Adobe Acrobat Reader. It should be used together with NVDA to test how the final PDF is presented to a screen-reader user.

3 Document Setup

This section gives a minimal setup for producing an accessible scientific PDF with LuaLaTeX. The commands in the preamble determine the PDF standard, document structure, fonts, mathematical accessibility, figures, and hyperlinks.

3.1 Document Metadata

The accessibility configuration should be placed before `\documentclass`:

```
\DocumentMetadata{  
  lang=en,  
  pdfstandard=ua-2,
```

¹<https://www.ohio.edu/oit/accessibility>

²<https://www.ohio.edu/graduate/current-students/thesis-dissertation-services/tad-process-checklist>

```

tagging=on,
tagging-setup={
  math/setup={mathml-AF,mathml-SE}
}
}

```

The individual settings have the following purposes:

- **lang=en** specifies the primary language of the document. This information helps assistive technologies use the appropriate pronunciation and language rules.
- **pdfstandard=ua-2** specifies PDF/UA-2 as the target accessibility standard.
- **tagging=on** enables tagged PDF output. Tags describe the logical structure of the document, such as headings, paragraphs, lists, tables, figures, captions, and mathematical expressions. Screen readers use this structure to determine the reading order and meaning of the content.
- **math/setup={mathml-AF,mathml-SE}** enables MathML representations of mathematical expressions. In our testing, using both MathML Associated Files (AF) and MathML Structure Elements (SE) provided reliable access to mathematical expressions with NVDA and MathCAT, including expressions containing left superscripts and subscripts.

The last setting is particularly important for scientific documents.

3.2 Document Class

After the metadata declaration, select the normal LaTeX document class:

```
\documentclass{article}
```

The **article** class is sufficient for this guide. Other standard classes may be used when appropriate, but it is useful to begin with the simplest possible document structure when developing or debugging an accessible PDF.

3.3 Fonts

An OpenType font is required. We used Unicode-capable OpenType fonts in our project:

```

\usepackage{fontspec}
\usepackage{unicode-math}
\setmainfont{Latin Modern Roman}
\setmathfont{Latin Modern Math}

```

3.4 Mathematics Packages

For scientific mathematics, we use:

```

\usepackage{amsmath}
\usepackage{mathtools}
\usepackage{luamml}

```

mathtools is especially important here, it extends **amsmath** and provides additional mathematical commands. It is also required for commands such as **\prescript**, which we use for accessible left superscripts and subscripts.

4 Document Structure

Use normal LaTeX structures and allow LuaLaTeX to tag them automatically.

4.1 Tags

Tags describe the semantic structure of a PDF. They tell assistive technology what each piece of content represents, rather than only how it looks visually.

Common tags include:

- **P**: paragraph
- **H1**, **H2**, **H3**: heading levels
- **L**: list
- **LI**: list item
- **Figure**: figure
- **Caption**: figure or table caption
- **Table**: table
- **TR**: table row
- **TH**: table header cell
- **TD**: table data cell
- **Formula**: mathematical expression

For example, a sighted reader may recognize large bold text as a heading, but a screen reader needs a heading tag to identify it as a heading. Similarly, a table must be tagged as a table with rows and header/data cells so that assistive technology can interpret its structure correctly.

When `tagging=on` is enabled, modern LaTeX can generate many of these tags automatically from standard LaTeX environments. But in our test we find that for figures we still need manual tagging, detailed information is in section 5.

4.2 Headings

Use real section commands:

```
\section{...}  
\subsection{...}  
\subsubsection{...}
```

Do not use bold text as a replacement for a heading.

4.3 Paragraphs

Separate paragraphs with a blank line or `\par`. Do not use `\\` to create a new paragraph. In our test with veraPDF, failing to end a paragraph before a table caused errors such as:

```
P shall not contain Caption  
P shall not contain Part  
P shall not contain P
```

Adding a blank line or `\par` before the table fixed the problem.

4.4 Lists

Use standard list environments:

```
\begin{itemize}
  \item First item
  \item Second item
\end{itemize}
```

or

```
\begin{enumerate}
  \item First item
  \item Second item
\end{enumerate}
```

Do not imitate a list using manually typed numbers and line breaks.

If you want to learn more about concepts like tags, heading and so on, Ohio University provides this detailed course: <https://ohio.instructure.com/courses/69542>

5 Figures

Keep the figure in a logical location in the LaTeX source so that the reading order remains sensible.

Use the normal figure environment, and remember to add tags manually, that is to wrap your figure code with

```
\tagstructbegin{tag=Sect}
```

and

```
\tagstructend
```

For example, your figure code should look like:

```
\tagstructbegin{tag=Sect}
\begin{figure}[H]
  \centering
  \includegraphics[
    width=0.6\linewidth,
    alt={Alternative text describing the important visual information.}
  ]{figure.png}
  \caption{Figure caption.}
  \label{fig:example}
\end{figure}
\tagstructend
```

5.1 Alternative Text

- Give every informative figure useful alternative text.
- Do not simply repeat the caption.
- Describe information that is available visually but is not already explained in the surrounding text or caption.

- If a question requires information from a figure, the alternative text must provide enough information for a screen-reader user to answer the question.
- For complicated plots, describe the axes, overall trend, and important peaks, edges, curves, or numerical values.
- For complicated diagrams, describe relationships and signal flow rather than simply listing every label.

6 Tables

Use a normal table and tabular environment and do not add manual tag.

```
\begin{table}[H]
\caption{Sample data for Cu powder pattern}
\makebox[\textwidth][c]{
\begin{tabular}{|c|c|c|c|}
\hline
 $\sin^2\theta$  &  $\sin^2\theta/0.1369$  &  $(\sin^2\theta/0.1369)*3$  & rounded & hkl \\
\hline
0.1369 & 1 & 3 & (111) \\
0.1809 & 1.32 & 4 & (200) \\
0.3642 & 2.66 & 8 & (220) \\
0.5012 & 3.66 & 11 & (311) \\
0.5455 & 3.98 & 12 & (222) \\
\hline
\end{tabular}
}
\label{XRD:table2}
\end{table}
```

Modern LuaLaTeX can automatically tag common structures such as paragraphs, headings, lists, captions, mathematics and tables. So DO NOT manually tag tables or you will mess up the table structure and get errors from veraPDF such as:

Caption should be first or last child.

And one more important thing is to end the paragraph before the table with a blank line or `\par`. If veraPDF reports errors such as

P shall not contain Caption

first check whether the paragraph immediately before the table has been properly ended.

7 Mathematics

Normal LaTeX mathematical environments can be used except for prescripts:

```
\begin{equation}
E = mc^2
\end{equation}
```

For prescripts we need to add the following environment:

```
math/setup={mathml-AF,mathml-SE}
```

and `LuaMML`, `mathtools`, and `unicode-math` enabled. For left superscripts and subscripts, avoid constructions such as:

```
{ }^{\{226\}_{88}}\mathrm{Ra}
```

because they may look correct visually but not be read correctly by a screen reader. Use `\prescript` instead:

```
\prescript{\{226\}_{88}}{\mathrm{Ra}}
```

Only a left superscript:

```
\prescript{\{226\}}{\mathrm{Ra}}
```

Only a left subscript:

```
\prescript{\mathrm{Ra}}{\{88\}}
```

Always test mathematical expressions with a screen reader. Passing an automated PDF validator does not guarantee that every formula will be read correctly.

8 Validation

Use veraPDF to validate the final PDF against PDF/UA-2. Do not only look at whether the file passes or fails. Read the exact rule and structure-tree path reported by veraPDF. After making structural changes, recompile the document from scratch and run the validator again.

9 Screen-Reader Testing

Always test the final PDF with a screen reader. For this project we used:

- Adobe Acrobat Reader,
- NVDA,
- MathCAT for mathematical expressions.

Check at least:

- heading navigation,
- paragraph reading order,
- lists,
- figure alternative text,
- tables and table structure,
- hyperlinks,
- mathematics, especially superscripts and subscripts,

A PDF that passes veraPDF may still contain content that is difficult or impossible to understand with a screen reader. Automated validation and screen-reader testing are both necessary.

10 Debugging

When something stops working, do not immediately debug the entire document. Instead use the following techniques:

1. Create a small LaTeX file containing only the feature that fails.
2. Confirm that the feature works in the small file.
3. Change only one variable at a time.
4. Recompile from scratch after important tagging changes.
5. Use the veraPDF error path to locate the malformed structure.
6. Test with a screen reader!

This method was especially useful for mathematical accessibility. When I debugging the subscript error, a minimal test file showed that `\prescript` worked correctly. Comparing that file with the full document then revealed that the full document used only `mathml-SE`. Adding `mathml-AF` helped correct reading of the mathematical expressions.

A Test the commonly used formulas

The following formulas are tested with veraPDF and NVDA screen reader with MathCat add-on running in Adobe Acrobat Reader.

A.0.1 Plain-text control

$$\vec{a} \tag{1}$$

The following letters and numbers are ordinary text rather than mathematics:

E, m, c, x, y, z, A, Z, N, Ra, U, 2, 19, 226, 238.

A.0.2 Basic mathematical content

$$E = mc^2 \tag{2}$$

$$x + y - z = 0 \tag{3}$$

$$\frac{ab}{c} = d \tag{4}$$

A.0.3 Right-side superscripts and subscripts

$$x^2 \tag{5}$$

$$x_i \tag{6}$$

$$x_i^2 \tag{7}$$

$$e^{-x^2} \tag{8}$$

$$2.01 \times 10^{19} \tag{9}$$

A.0.4 Left-side scripts and nuclear notation

$$^AX \tag{10}$$

$$_ZX \tag{11}$$

$$\overset{A}{Z}X \tag{12}$$

$$\overset{A}{Z}X_N \tag{13}$$

$$^{226}_{88}Ra \tag{14}$$

$$^{238}_{92}U_{146} \tag{15}$$

$${}_Z^AX_N\rightarrow {}_{Z-2}^{A-4}X_{N-2}+\alpha \tag{16}$$

$${}^{226}_{88}Ra\rightarrow {}^{222}_{86}Rn+{}_2^4\alpha \tag{17}$$

$${}_Z^AX_N\rightarrow {}_{Z+1}^AY_{N-1}+e^-+\overline{\nu}_e \tag{18}$$

$${}_Z^AX_N\rightarrow {}_{Z-1}^AY_{N+1}+e^++\nu_e \tag{19}$$

A.0.5 Greek letters and common physics symbols

$$\alpha, \ \beta, \ \gamma, \ \Delta, \ \lambda, \ \mu, \ \sigma, \ \Omega, \ \omega \tag{20}$$

$$\frac{\partial f}{\partial x} \tag{21}$$

$$\mathbf{p}=m\mathbf{v} \tag{22}$$

$$\mathbf{a}\cdot\mathbf{b} \qquad \mathbf{a}\times\mathbf{b} \tag{23}$$

A.0.6 Fractions, roots, and delimiters

$$\frac{1}{2} \tag{24}$$

$$\frac{1}{1+\frac{x}{y}} \tag{25}$$

$$\sqrt{x^2+y^2} \tag{26}$$

$$\sqrt[n]{x} \tag{27}$$

$$|x| \qquad \|\mathbf{v}\| \tag{28}$$

$$(x+y) \qquad [x+y] \qquad \{x,y,z\} \tag{29}$$

A.0.7 Calculus and large operators

$$\int_0^\infty e^{-x^2} dx \quad (30)$$

$$\iint_D f(x, y) dA \quad (31)$$

$$\oint_C \mathbf{F} \cdot d\mathbf{r} \quad (32)$$

$$\sum_{i=1}^n x_i \quad (33)$$

$$\sum_{n=0}^\infty x^n \quad (34)$$

$$\prod_{k=1}^n a_k \quad (35)$$

$$\lim_{x \rightarrow 0} \frac{\sin x}{x} = 1 \quad (36)$$

A.0.8 Functions and operators

$$\sin^2 \theta + \cos^2 \theta = 1 \quad (37)$$

$$\ln x + \log_{10} y \quad (38)$$

$$f(t) = Ae^{-\lambda t} \quad (39)$$

$$P(A | B) = \frac{P(A \cap B)}{P(B)} \quad (40)$$

A.0.9 Matrices, vectors, and systems

$$\begin{pmatrix} a & b \\ c & d \end{pmatrix} \quad (41)$$

$$\det \begin{pmatrix} a & b \\ c & d \end{pmatrix} = ad - bc \quad (42)$$

$$\mathbf{v} = \begin{pmatrix} x \\ y \\ z \end{pmatrix} \quad (43)$$

$$A\mathbf{v} = \lambda\mathbf{v} \quad (44)$$

$$\begin{aligned} x + y &= 1, \\ x - y &= 0. \end{aligned} \quad (45)$$

$$f(x) = \begin{cases} x^2, & x \geq 0, \\ -x, & x < 0. \end{cases} \quad (46)$$

A.0.10 Accents, decorations, and special scripts

$$\hat{x} \quad \bar{x} \quad \dot{x} \quad \tilde{x} \tag{47}$$

$$\overline{A+B} \quad \underline{x+y} \tag{48}$$

$$\psi^* \quad A^\dagger \tag{49}$$

$$\langle\psi| \quad |\phi\rangle \quad \langle\psi|\phi\rangle \tag{50}$$

A.0.11 Units and mixed text

$$E = 5.3 \text{ MeV} \tag{51}$$

$$T_{1/2} = 138 \text{ days} \tag{52}$$

$$T_{1/2} = 162 \, \mu\text{s} \tag{53}$$

$$\lambda = \frac{\ln 2}{T_{1/2}} \tag{54}$$