

Accessible PDFs with LaTeX

Hunter Lehmann

Georgia Institute of Technology

September 18, 2025

A Motivating Example

```
\documentclass{article}
\usepackage{amsmath}
\usepackage{unicode-math}

\title{Demo of Accessible LaTeX}
\author{Hunter Lehmann}
\date{September 18, 2025}

\begin{document}

\maketitle
\section{Introduction}

This is a demo document.
```

```
\section{Mathematics}

Here is an equation  $(a^2+b^2=c^2)$ .
Next we have a numbered display equation:
\begin{equation}
\lambda \begin{bmatrix}
1 & 2 \\
3 & 4
\end{bmatrix} = \begin{bmatrix}
1 & 2 \\
3 & 4
\end{bmatrix} \mathbf{v}
\end{equation}
\end{document}
```

A Motivating Example (cont.)

A short snippet of a screen reader the untagged PDF.

A screen reader reading the same document with proper tagging and MathML automatically generated.

Why Now?

- Long term issue; vicious cycle starting to be broken
- Recent updates -> better support, especially for math
- New rule on web content accessibility in April 2026
- Most basic documents (e.g. articles, course notes, worksheets, quizzes) can now be automatically tagged and made fully screen-reader accessible with minimal effort*
- More complex documents (e.g. exams, slides) need more work but are also doable

How to Do It

- Use an up-to-date TeX distribution (e.g. TeX Live 2025 or later)

How to Do It

- Use an up-to-date TeX distribution (e.g. TeX Live 2025 or later)
- Add the following lines to your preamble:

```
\DocumentMetadata{  
  lang          = en-US,  
  pdfstandard   = ua-2,  
  pdfstandard   = a-4f,  
  tagging=on,  
  tagging-setup={math/setup={mathml-SE,mathml-AF}}  
}
```

How to Do It

- Use an up-to-date TeX distribution (e.g. TeX Live 2025 or later)
- Add the following lines to your preamble:

```
\DocumentMetadata{  
  lang          = en-US,  
  pdfstandard   = ua-2,  
  pdfstandard   = a-4f,  
  tagging=on,  
  tagging-setup={math/setup={mathml-SE,mathml-AF}}  
}
```

- Add alt-text to graphics:

```
\includegraphics[alt={A description of the image}]{imagefile}  
\includegraphics[artifact]{decorationimage}
```

This also works with `tikz`.

How to Do It (cont.)

- Mark table headers:

```
\tagpdfsetup{table/header-rows={1}}  
\begin{tabular}{|c|c|}  
  \hline  
  \textbf{Header 1} & \textbf{Header 2} \\ \hline  
  Cell 1 & Cell 2 \\ \hline  
\end{tabular}
```

or `\tagpdfsetup{table/tagging=presentation}` for tables used for layout only

How to Do It (cont.)

- Mark table headers:

```
\tagpdfsetup{table/header-rows={1}}  
\begin{tabular}{|c|c|}  
  \hline  
  \textbf{Header 1} & \textbf{Header 2} \\ \hline  
  Cell 1 & Cell 2 \\ \hline  
\end{tabular}
```

or `\tagpdfsetup{table/tagging=presentation}` for tables used for layout only

- Add `\usepackage{unicode-math}` and compile with `LuaATeX`

How to Do It (cont.)

- Mark table headers:

```
\tagpdfsetup{table/header-rows={1}}  
\begin{tabular}{|c|c|}  
  \hline  
  \textbf{Header 1} & \textbf{Header 2} \\  
  \hline  
  Cell 1 & Cell 2 \\  
  \hline  
\end{tabular}
```

or `\tagpdfsetup{table/tagging=presentation}` for tables used for layout only

- Add `\usepackage{unicode-math}` and compile with Lua^AT_EX
- Adjust remaining incompatibilities

How to Do It (cont.)

- Mark table headers:

```
\tagpdfsetup{table/header-rows={1}}  
\begin{tabular}{|c|c|}  
  \hline  
  \textbf{Header 1} & \textbf{Header 2} \\  
  \hline  
  Cell 1 & Cell 2 \\  
  \hline  
\end{tabular}
```

or `\tagpdfsetup{table/tagging=presentation}` for tables used for layout only

- Add `\usepackage{unicode-math}` and compile with Lua^AT_EX
- Adjust remaining incompatibilities
- Test with screen readers and PDF accessibility checkers

Notable Packages w/ Incompatibilities

Partially Compatible

- Doc Classes:
amsart,amsbook,amsproc, amseconf
- amsmath,amsthm
- cleveref, enumitem
- pstricks, tcolorbox

Not Compatible with Unicode-Math

- amssymb,amsfonts,bm

Currently Incompatible

- Doc Classes: exam, IEEEtran
- amscd, chemfig, soul
- tikz, titlesec, todonotes

No Support

- Doc Classes: beamer

Full List of Known Compatibility Issues: <https://latex3.github.io/tagging-project/tagging-status/>

Best Practices

- Use semantic markup (e.g. `\maketitle`, `\section`, `\emph` vs just changing font size/style)
- Provide alt-text for all images
- Use meaningful commands when possible (e.g. `\abs{x}` instead of `|x|`)
- If using custom colors, ensure sufficient contrast and test with colorblindness simulators
- Avoid using color as the only means of conveying information

- Conceptually harder to properly tag slides
- Beamer is not supported and will not be
- Instead, use the `ltx-talk` package (what I'm using here)

- Conceptually harder to properly tag slides
- Beamer is not supported and will not be
- Instead, use the `ltx-talk` package (what I'm using here)
- Similar syntax to Beamer, but some changes do exist
- Requires the November 2025 release of LaTeX; access pre-release via compiling with `lualatex-dev`
- Can be done in Overleaf but is a pain; strongly recommend local compilation

- The LaTeX Tagging Project website:
<https://latex3.github.io/tagging-project/>
- The ltx-talk package on CTAN:
<https://ctan.org/pkg/ltx-talk>
- My shared folder with examples/resources:
<https://tinyurl.com/GTTaggedLaTeX>



Practice!

Pick a short document you've written and try to make it accessible using the steps outlined here.