

Author's guide to texedbook

Riley Hanus

November 28, 2023

Abstract

The **texedbook** code base is a tool for publishing articles and textbooks online without the need to learn html, css, and javascript. The author writes and compiles the content in latex and (after following the set-up instructions in the README.md) runs

```
python run.py ./path/to/latex/project/directory
```

which generates a collection of html and css files that can be directly published online. The product is a computer and mobile friendly webpage. In addition to supporting most of the native latex features, texedbook provides tools for embedding digital and interactive content such as videos, quizzes, code editors and compilers, etc.

This article acts as an author's guide to **texedbook** and will demonstrate the document features that are explicitly supported, and the specific way they must be used for a clean output html page. First, the native latex capabilities will be demonstrated including figures, equations, tables, cross-referencing, citations, etc. Then the **texedbook** specific features will be presented which allow the author to embed digital content (essentially anything that can be contained in an iframe) straight from the latex document.

1 Motivation for texedbook project

When writing an article or textbook, especially one that is technical in nature, the need for proper handling of standard document features is glaring. Without a framework to efficiently manage citations, cross-reference document items (sections, equations, figures, etc.), and write math in real time, writing anything with technical substance becomes prohibitively difficult. Latex, despite its quarks, is a very good framework to manage these critical writing tools and compiling beautifully typeset pdf documents.

At the same time, publishing and education has transitioned to digital-first for the consumer, yet publishers still cling to a print-first model. There is a clear need for a tool enabling authors to publish digital-first articles, books, and courses. The natural medium this digital-first publishing is an html/css/js based webpage.

texedbook combines the strengths of latex for authoring articles and books with the versatility and universality of an html/css/js based webpage.

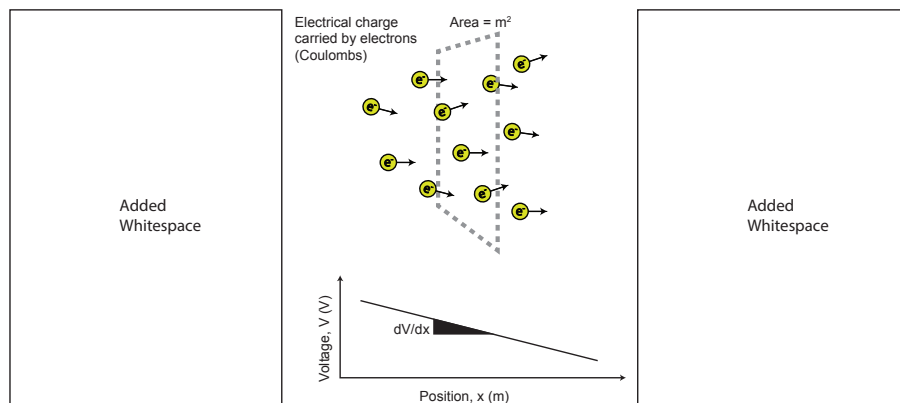


Figure 1: Example figure. Since the convention for texedbook is for figures to span the text width, whitespace added to control size of the image.

2 Native latex features

2.1 Figures

All figures used in the document must be contained in a directory named `./figures/` and that folder must be located in the same location as `main.tex`. In addition, all figures must be in pdf format.

Ensuring that the figures are displayed correctly in the resulting html is tricky. The simplest and most effective way is to design the width of all figures to span width of the text on the page. This display condition reliably converts from latex to html and translates from computer to mobile formats well, which is not the case for all display options. If you would like the figure to display smaller, simply add white space to the left and right of the pdf as is done in Figure 1.

An example of the latex code used to add a figure is

```
\begin{figure}[t]
  \centering
  \includegraphics[width=0.99\textwidth, keepaspectratio]{figures/fig.pdf}
  \caption{Example figure. Since the convention for texedbook is for figures to span t
  \label{fig:example}
\end{figure}
```

The `[t]` places the figure at the top of a page in the pdf, and does nothing to the html output. The html output will place the figure where it appears in the code.

Table 1: Example table.

Sample	Data 1	Data 2	Data 3
A	53	62	75
B	51	61	72
C	58	69	71

2.2 Equations

Equations are an inherently tricky problem for digital publishing. The core of the problem lies in the fact that html was designed around the standard alphanumeric alphabet, and math requires a wider range of complex symbols and typesetting. The default usage of `texedbook` leverages `mathjax`, allowing all the native latex equations to be reliably reproduced in the html output.

In line equations can be included $n\lambda = 2d \sin \theta$, as well as displayed equations

$$n\lambda = 2d \sin \theta. \tag{1}$$

Cross-referencing equations is done using the `\mjref` function provided in `texedbook.tex`. For example, Eq. 1 is Bragg’s Law, commonly used to find diffraction conditions of light of wavelength λ interacting with a crystal with lattice spacing d at an incident angle of θ . The `\mjref` function simply substitutes `\ref` when the latex is compiled. However, when `tex4ebook` converts the project to html, `config.cfg` file tells `tex4ebook` to insert `\eqref` in place of all `\mjrefs`, which `mathjax` knows to look for in the html and use for equation numbering and cross-referencing.

2.3 Tables

Tables are inherently difficult to write in latex. Tables Generator is a great tool for generating latex code for tables. The table is translated into html. Note that the table styling in the html output is different than the latex output.

2.4 Code

Code can be displayed using the `verbatim` environment native to latex

```
\begin{verbatim}
  for i in list:
    i='some code'
\end{verbatim}
```

Code can be written in-line using the `\verb` function. This function comes from the `verbatim` package and is unique in that it doesn’t use the curly brackets for its argument. It takes the first non-letter character as the open argument and looks for that same character again to close the argument. I typically use `\verb'code here'`.

2.5 Cross-referencing and hyperlinks

Cross-referencing can be used as normal in latex and all of the hyperlinking functionality will be preserved in the html. For example, Section 2, Table 1, and Figure 1. The only important exception is Equations where the `\mjref` command must be used such that mathjax can properly label and cross-reference equations locally in the html page using javascript (see Section 2.2).

Hyperlinks to external webpages can be added in latex using the `\href` command.

```
\href{https://www.webpage.com}{Display name in document}
```

2.6 Citations

The native bibliography features are maintained. For example, one can cite an article [1] or multiple articles [2, 1, 3]. Note that the hyperlinking to the bibliography in the pdf, a result of using the `hyperref` package, is maintained in the resulting html. The default bibliography style used in `texedbook` is `unsrturl`. Note that this style supports hyperlinking to the url, and/or doi if that field is present in the .bib entry, and the hyperlinking is maintained in the resulting html. It is recommended that the doi field is used instead of the url, since line breaking needed to typeset urls is handled poorly by latex. Latex's performance on typesetting doi hyperlinks seems to work more reliably.

References

- [1] Riley Hanus, Ramya Gurunathan, Lucas Lindsay, Matthias T. Agne, Jingjing Shi, Samuel Graham, and G. Jeffrey Snyder. Thermal transport in defective and disordered materials. *Applied Physics Reviews*, 8(3):031311, 08 2021. doi:10.1063/5.0055593.
- [2] Riley Hanus, Matthias T. Agne, Alexander J. E. Rettie, Zhiwei Chen, Gangjian Tan, Duck Young Chung, Mercouri G. Kanatzidis, Yanzhong Pei, Peter W. Voorhees, and G. Jeffrey Snyder. Lattice softening significantly reduces thermal conductivity and leads to high thermoelectric efficiency. *Advanced Materials*, 31(21):1900108, 2019. doi:https://doi.org/10.1002/adma.201900108.
- [3] Shawn A. Gregory, Riley Hanus, Amalie Atassi, Joshua M. Rinehart, Jamie P. Wooding, Akanksha K. Menon, Mark D. Losego, G. Jeffery Snyder, and Shannon K. Yee. Quantifying charge carrier localization in chemically doped semiconducting polymers. *Nature Materials*, 20(10):1414–1421, 2021. doi:10.1038/s41563-021-01008-0.