Preparing your thesis in LaTeX

David S. Latchman

Abstract

The submission of a thesis or dissertation is the culmination of many a graduate student's career. Given the time and effort toward research and attaining their degrees, this can often be a stressful time for many students. LaTeX offers the advantage of separating form from content and as the typical university thesis class can take care of a university's formatting requirements, this makes a student's life easier — well, at least it is supposed to.

Unfortunately, some formatting 'blends' into content thereby adding to the stress of an already unpleasant task. But there is some light at the end of the tunnel. With some preparation, typesetting a thesis in LaTeX can be relatively pain-free. But it's not just a matter of knowing what packages but *how* to use them and what is needed to use them effectively. Topics covered will include the typesetting of equations — both mathematical and chemical — as well as the proper formatting of tables and bibliographies.

1 The graduate student and LaTeX

As a TeX consultant, I often work with graduate students to help them get their theses and dissertations ready for final submission. For many graduate students, this marks their first foray into LaTeX and can be, for some, a most stressful experience; the emotional roller coaster ride of dealing with a review board notwithstanding. In most cases, the writing and editing process begins in Microsoft Word and continues there to the final editing and proofreading stages, where they are now getting ready to convert their thesis to LaTeX and eventual submission. There are many reasons why LaTeX isn't used at the start of the writing process. Some academic advisers and students are simply unfamiliar with LaTeX. It can also be difficult to annotate and track changes in LaTeX.

While many universities have their own thesis and dissertation class files to automate formatting of the document itself, class files intentionally fall short in many ways. Such class files are also designed to meet the needs of as many students as possible. Students can then add packages in the preamble depending on their personal needs and requirements. Applied correctly, the right package can help a student stay focused on writing and explaining themselves and not on how to format something.

Typically, a student turns to a consultant when they are unable to compile their document and help is not available. There might be that lone person in the lab or research group who knows LaTeX but who might not have the time or may be unwilling to put the effort into correcting errors. Moreover, consulting with students isn't just about fixing errors. After a project is completed, students are going to continue to make edits. Thus fixing any problems also means educating clients on how they can do things better and make their lives easier. Some of the packages and techniques discussed in this article are based on past projects I have worked on.

2 A theology dissertation

2.1 Language-specific packages

As a science graduate, I often find humanities projects interesting as I infrequently have the opportunity to work in the area. One recent project was a theology thesis that needed to be converted from Word to LaTeX. By itself, that would have been an interesting thesis to read — but since it was in German, and I don't speak a word of the language I could focus only on the technical aspects of the project.

LaTeX is structured for academic writing. While authors generally attempt to divide their writing into chapters and sections in Word, they don't always format their document properly by using the appropriate headings. This isn't necessary only for Word to create a table of contents but for any program to convert a Word document to LaTeX. Once the appropriate headings are set, conversion can take place.

While most universities have very strict formatting guidelines, in the case of this project there were none. Some students think of that as a positive, as it allows the freedom to design and format a thesis to look as they would like. While that may be true to some extent, we need to follow some typographic rules and formatting and not ignore them completely. I decided on the use of the KOMA-Script package as it follows European rules for typography and defaults to ISO A4 sized paper [1]. After an appropriate cover was designed, I focused on typography.

The KOMA-script package has book, article and report classes. The `DIVcalc` option for the `scrbook` class allows for the automatic calculation of both the page margins and text area based on font size. While it may seem lazy to make the package do these calculations, it removes the temptation for users to interfere with the settings of their `.tex` file. This can be done by specifying the options in the `\documentclass`:

```
\documentclass[DIVcalc, 11pt]{scrbook}
```

As the thesis is in German, the `babel` package was also needed [2]:

```
\usepackage[german]{babel}
```

This allows LaTeX to follow German-specific typesetting rules and ensure proper hyphenation.

The next step is to allow my client to type their thesis as easily as possible using the text editor of their choice. Generally, native English writers don't have to contend with accented characters and may forget that they even exist. If we do need them LaTeX has various commands to produce them. This is fine for the occasional accented character but it would be tedious if one is writing an entire report in a non-English language. As most text editors and word processors encode text in UTF-8, which includes characters with accents, we can use the `inputenc` package in the document's preamble to allow an author to write and edit their thesis as naturally as possible [3]:

```
\usepackage[utf8]{inputenc}
```

We also need to load fonts that will support accented characters found in European languages:

```
\usepackage[T1]{fontenc}
```

With the inclusion of the above packages, a student can type and edit the thesis with their favorite text editor.

2.2 Bibliographies

One of the strengths of LaTeX is its ability to handle bibliographic data using BibTeX [4]. English users generally experience no problems compiling their documents to produce a bibliography, but non-English language users are not always so fortunate. BibTeX was first released in 1988 and predates the advent of Unicode. Thus, it does not support files encoded in UTF-8, the default file encoding on most modern OSes. While it is possible to recode a `.bib` file into something that BibTeX can understand, this may be a complex task for a non-computer-savvy first-time LaTeX user. A solution should always allow someone to compile their thesis with little to no effort.

LaTeX does offer some solutions to this problem; the one that stands out for me is the BibLaTeX package. BibLaTeX has several advantages over BibTeX. For one, it can handle files that are encoded in UTF-8. While BibLaTeX has suffered from a lack of bibliographic styles in the past, this has changed in recent times. For German humanities students, we now have the `biblatex-fiwi` package [5]:

```
\usepackage[style=fiwi]{biblatex}
```

Using the BibLaTeX package is slightly different from using BibTeX. We still need the `.bib` file where the bibliographic data is stored:

```
@BOOK{KR,
   AUTHOR = "Kernighan, Brian W. and Ritchie, Dennis
        M.",
   TITLE  = "{The C Programming Language Second
        Edition}",
   PUBLISHER = "Prentice-Hall, Inc.",
   YEAR = 1988
 }
```

To incorporate a bibliography with BibLaTeX:

```
\documentclass[DIVcalc, 11pt]{scrbook}
\usepackage[german]{babel}
\usepackage[utf8]{inputenc}
\usepackage[T1]{fontenc}
\usepackage[style=fiwi]{biblatex}
\bibliography{refs}
\begin{document}
All works should be properly cited~\cite{KR}.

\printbibliography
\end{document}
```

In BibTeX, the `\bibliography` command serves two purposes: it controls where the bibliography appears in the text and specifies the file containing the bibliography information. With BibLaTeX, however, the `\bibliography` command just reads the file and appears before the `\begin{document}` environment while the position of the `\printbibliography` command controls where the bibliography appears.

BibLaTeX also makes compiling the thesis easier. With BibTeX, LaTeX (here, `pdflatex`) has to be run twice after `bibtex`:

```
pdflatex mythesis.tex
bibtex mythesis.aux
pdflatex mythesis.tex
pdflatex mythesis.tex
```

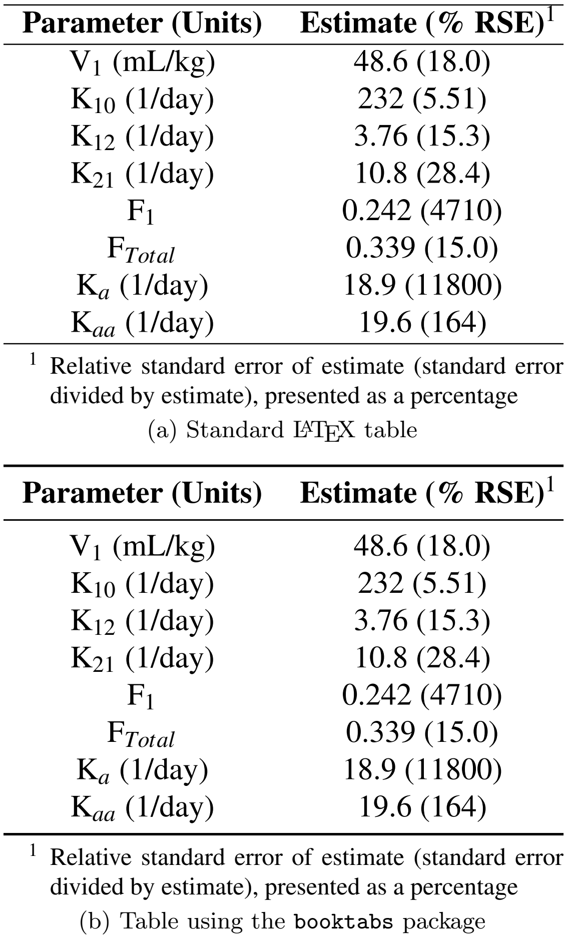
With BibLaTeX, it only needs to be run once:

```
pdflatex mythesis.tex
bibtex mythesis.aux
pdflatex mythesis.tex
```

3 Publication-quality tables

Proper formatting of tables is another task I look at. Tables feature mainly in the sciences and engineering and may be infrequently seen in the humanities. Nonetheless, proper formatting is important. A properly formatted table organizes and displays data in rows and columns and should make it easy to look up values.

For many people, the standard practice is to design tables in Excel or some other spreadsheet program and translate this into LaTeX. Unfortunately, this often leads to poor table design. It may come as a surprise to some, but there are rules for proper table design which have been established for centuries [6]. Proper table design can be an art in itself, but the fundamentals are easy to follow. They are:



Parameter (Units)	Estimate (% RSE)[1]
V_1 (mL/kg)	48.6 (18.0)
K_{10} (1/day)	232 (5.51)
K_{12} (1/day)	3.76 (15.3)
K_{21} (1/day)	10.8 (28.4)
F_1	0.242 (4710)
F_{Total}	0.339 (15.0)
K_a (1/day)	18.9 (11800)
K_{aa} (1/day)	19.6 (164)

[1] Relative standard error of estimate (standard error divided by estimate), presented as a percentage

(a) Standard LaTeX table

Parameter (Units)	Estimate (% RSE)[1]
V_1 (mL/kg)	48.6 (18.0)
K_{10} (1/day)	232 (5.51)
K_{12} (1/day)	3.76 (15.3)
K_{21} (1/day)	10.8 (28.4)
F_1	0.242 (4710)
F_{Total}	0.339 (15.0)
K_a (1/day)	18.9 (11800)
K_{aa} (1/day)	19.6 (164)

[1] Relative standard error of estimate (standard error divided by estimate), presented as a percentage

(b) Table using the booktabs package

Figure 1: A table using LaTeX vs. booktabs.

1. Avoid use of vertical lines,
2. Avoid double horizontal lines,
3. Always place the units in the heading of the table instead of the body, and
4. Do not use quotation marks to repeat the content of cells.

Once a table has been edited to conform to the above rules, I include the booktabs package [7].

```
\usepackage{booktabs}
```

This package increases the vertical spacing between lines, increasing the white space in the table which allows data to be found and read easily.

We can see the differences between the two tables. The bottom table has more space between lines. This improves readability as the eye can follow a line more easily. This is an easy fix that makes a thesis look noticeably better.

4 Breaking equations

Sometimes students in the sciences and engineering are faced with the challenge of dealing with equations that span more than one line. The collection of AMS-LaTeX packages allows for the typesetting of multi-line equations with the align, split and gather environments. While these packages can and do work, they all require equations to be broken manually. This poses a problem as breaks are sensitive to changes in fonts, column width and various other edits in the equation itself. Ideally a solution should exist that allows LaTeX to automatically break equations and allow students to focus on the process of writing their thesis.

The equation shown (fig. 2) appears in a biophysics thesis on the effects of drugs on cancer growth. We can see in fig. 2(a) that there are several problems with using the align environment to manually break equations as the \left and \right delimiters for parentheses do not span lines. This has the effect of parentheses with different sizes on different lines. While it is possible to manually adjust the size of the parentheses, this means that considerable effort and attention must be given to equation editing.

One solution to this problem is the breqn package, which supports automatic line breaking [8], as seen in fig. 2(b). Usually, when edits have been made to equations, attention must be paid to carefully check that equations have been typeset properly. By allowing the package to handle equation breaking and formatting, students can focus on editing and writing their thesis. (The dmath environment is just like the equation environment except that it supports line breaking.)

5 Nitrogen isotopes and technical editing

Another project I worked on was a thesis that focused on the nitrogen isotope cycle in the Arctic. Nitrogen and its compounds cycle through the planet and the rates at which the various isotopes do so differ from each other. By tracking the differences in cycle rates, scientists can better understand the implications for climate change. This was a technical editing project and I needed to ensure that physical quantities and chemical equations were entered correctly. As these featured heavily in the thesis, an easier solution for both me and my client was needed.

Physical quantities comprise a number and an associated SI unit (*Système International d'Unitès*) and chemical equations are symbolic representations of chemical reactions from beginning to end. Both can be entered in LaTeX by entering math mode but this poses problems — starting with editing. While entering equations in TeX is intuitive, this is only applicable to mathematical equations and one has to literally enter an SI unit or chemical equation as a mathematical equation which brings us to the next

```
\begin{align*}
\frac{d}{dt}\ApopSig(t) &= K_{\inApop,\UpSig} \cdot \UpSig(t) \cdot \\
 &\left( 1 - \right.\\
 & \left.\frac{\ProSrvlSig(t)}{IC_{50,\ProSrvlSig} + \ProSrvlSig(t)} \right) \\
 & - K_{\outApop} \cdot \ApopSig(t)
\end{align*}
```

$$\begin{aligned}\frac{d}{dt}\mathit{ApopSig}(t) &= K_{\mathit{inApop},\mathit{Upsig}} \cdot \mathit{Upsig}(t)\cdot\\ &\left(1-\right.\\ &\left.\frac{\mathit{ProSrvlSig}(t)}{IC_{50,\mathit{ProSrvlSig}} + \mathit{ProSrvlSig}(t)}\right)\\ &- K_{\mathit{outApop}} \cdot \mathit{ApopSig}(t)\end{aligned}$$

(a) Less than ideal line breaking using the `align` environment

```
\begin{dmath*}
\frac{d}{dt}\ApopSig(t) = K_{\inApop,\UpSig} \cdot \UpSig(t) \cdot \left( 1 -
  \frac{\ProSrvlSig(t)}{IC_{50,\ProSrvlSig} + \ProSrvlSig(t)} \right)
  - K_{\outApop} \cdot \ApopSig(t)
\end{dmath*}
```

$$\begin{aligned}\frac{d}{dt}\mathit{ApopSig}(t) = K_{\mathit{inApop},\mathit{Upsig}} \cdot \mathit{Upsig}(t) \cdot \left(1 - \frac{\mathit{ProSrvlSig}(t)}{IC_{50,\mathit{ProSrvlSig}} + \mathit{ProSrvlSig}(t)}\right)\\ - K_{\mathit{outApop}} \cdot \mathit{ApopSig}(t)\end{aligned}$$

(b) Much better line breaking using the `breqn` package

Figure 2: Equation breaking using `align` vs. `breqn`. (`\ApopSig`, etc., just typesets `\mathit{ApopSig}`).

problem, that of consistency. Typesetting equations has its own set of rules from the spacing between numbers and variables to how superscripts and subscripts are placed. A writer or editor will always have to keep these typesetting rules in mind. A solution to these problems is to utilize the `siunitx` and `mhchem` packages [9, 10].

5.1 SI units

Entering physical quantities in LaTeX can be something of a challenge as the subscript and superscript operators are only available in math mode. Thus,

```
This is my 1^{st} article in a TUG journal.
```

will result in an error and not compile. Of course, one can simply enter math mode to write in superscript but we are then faced with the problem of the text being in (math) italics.

```
This is my 1$^{st}$ article in a TUG journal.
```

becomes

> This is my 1^{st} article in a TUG journal.

This particular case doesn't look *too* bad but the italicization can be a problem when typesetting physical units. To solve this, we can enter math mode, print text and then leave math mode.

```
This is my 1$^{\textrm{st}}$ article in a TUG
     journal
```

becomes

> This is my $1^{\textrm{st}}$ article in a TUG journal.

This looks better and works when typesetting physical quantities. While this may be easy for one or two cases when one needs to typeset physical quantities, it can become difficult to maintain consistency during the writing and editing process for many units and troubleshoot problems when they do arise. Consider:

```
The acceleration due to Earth's gravity, $g$,
is $9.81\, \textrm{m}\,\textrm{s}^{-2}$.
```

becomes

> The acceleration due to Earth's gravity, g, is $9.81\,\textrm{m}\,\textrm{s}^{-2}$.

The `siunitx` package provides users with the tools to typeset numbers and units in a way that is both consistent and easy. Each unit is defined by its own macro. When typing a physical quantity, one enters the information syntactically and this makes for much easier editing without the need to constantly compile to see and verify output.

The `siunitx` package can be loaded with the `\usepackage` command as usual:

```
\usepackage{siunitx}
```

Typesetting SI units then becomes simple. The `\SI` macro allows a writer to enter a number with its physical quantity. Thus:

```
The acceleration due to Earth's gravity, $g$, is
\SI{9.81}{\meter\per\second\squared}.
```

As we can see, this package allows for physical quantities to be written as they are said in day-to-day language. This makes for much easier writing as well as proofreading and editing.

5.2 Chemical equations

Many of the same problems that exist with typesetting and editing physical quantities in LaTeX also exist with chemical equations. One can manually enter math mode, typeset a chemical equation using TeX's mathematical notation and then leave math mode. To solve some of these problems and make editing and proofreading easier, I turned to the mhchem package. This allows for chemical equations and symbols to be typeset easily.

The biggest plus of the mhchem package is that the notation is both easy to read and write. It is much easier to type the chemical notation for phosphoric acid as H3PO4 than H_3PO_4. The former is typically used in email communication when discussing chemical equations and reactions and is thus a more natural form of expression. The mhchem package uses this form of notation. Instead of using mathematical notation to write phosphoric acid, we can write the chemical as:

```
$\ce{H3PO4}$
```

which gives $\mathrm{H_3PO_4}$. As the package does all of the heavy work, this means neither the author nor editor has to think about formatting. This makes the writing and editing process much easier.

The true strength of the mhchem package comes from entering and typesetting chemical equations. In math mode, the chemical equation for photosynthesis would be written as

```
$6\,\mathrm{CO}_2 + 6\,\mathrm{H}_2\mathrm{O} \
    xrightarrow{\textrm{light}} \mathrm{C}_6\mathrm{
    H}_{12}\mathrm{O}_6 + 6\,\mathrm{O}_2$
```

As we can see, this is not only difficult to write but read as well. Instead, we can write

```
\ce{6CO2 + 6H2O ->[\text{light}] C6H12O6 + 6O2}
```

which is a much more natural representation of

$$6\,\mathrm{CO_2} + 6\,\mathrm{H_2O} \xrightarrow{\text{light}} \mathrm{C_6H_{12}O_6} + 6\,\mathrm{O_2}$$

This is much easier to read, edit and check for problems and can be entered in either math mode or text mode.

6 Conclusion

The packages mentioned are by no means a comprehensive list of the packages that a student should use but rather highlight how appropriate choices can make writing and editing a thesis easier and allow for consistent typesetting. This means that a thesis doesn't just read well but also looks good. University thesis and dissertation class files are designed to fit the needs of as many students as possible and it would be impossible to include all the packages that every student could conceivably need. To do so would make a class file unreasonably large and make it more difficult to maintain.

The advantage of a minimalist design or including the packages that most students need is that packages can be added by students depending on their needs. It would be best for students to take the time to acquaint themselves with the various LaTeX packages available and how they can best use and take advantage of them to make the eventual writing and editing process easier. But for first-time LaTeX users, this may be an intimidating task; not only must they learn how to use LaTeX but they must also select the "right" packages they need. If possible, it may be best to seek the advice of a more experienced LaTeX user who can offer advice on how best to prepare the packages to be used in the document's preamble.

In the absence of such an expert, academic departments can help their students by compiling how-to lists of the most commonly used packages to be published on the department website or be given along with the university's formatting requirements. This should help students be better prepared to focus on the task of writing and publishing their research and hopefully, eventually graduating.

Bibliography

[1] Markus Kohn, Jens-Uwe Morawski, Frank Neukam, and Axel Kielhorn. *The Guide: KOMA-Script*, 2012.

[2] Johannes Braams. *Babel, a multilingual package for use with LaTeX's standard document classes*, 2011.

[3] Alan Jeffrey and Frank Mittelbach. *inputenc.sty*, 2008.

[4] Philipp Lehman, Audrey Boruvka, Philip Kime, and Joseph Wright. *The BibLaTeX Package*, 2012.

[5] Simon Spiegel. *Der biblatex-fiwi-Stil*, 2011.

[6] Stephen Few. *Show Me the Numbers: Designing Tables and Graphs to Enlighten.* Analytics Press, Oakland, Calif., 2004.

[7] Simon Fear. *Publication quality tables in LaTeX*, 2005.

[8] The breqn maintainers. *The breqn package*, 2012.

[9] Joseph Wright. *siunitx — A comprehensive (SI) units package*, 2011.

[10] Martin Hensel. *The mhchem bundle*, 2011.

◇ David S. Latchman
texnical dot designs (at) gmail dot com