

How to Seminar

AG Softwaretechnik

October 9, 2017

Abstract

This document can be used as a template for your seminar paper. Just remove the existing sections and adapt the title and author information above.

The original version of this document was created by Raphael Reitzig and Sebastian Wild.

1 How to install L^AT_EX

Linux Your distribution will most likely offer a Latex package. For Ubuntu you can install the `texlive-full` package via:

```
sudo apt install texlive-full
```

Mac Download and install the MacTeX distribution (<https://www.tug.org/mactex/>).

Windows You can install TeX Live (<https://www.tug.org/texlive/windows.html>) or MiKTeX (<https://miktex.org/howto/install-miktex>).

Make sure that the `latexmk` tool is also installed. You can find installation instructions for `latexmk` at <http://mg.readthedocs.io/latexmk.html#installation>.

2 How to build this document

In essence, building a L^AT_EX document is as simple as running

```
latexmk -pdf seminar.tex
```

Here `seminar.tex` is the name of the main `tex`-file.

The `latexmk` tool integrates several tools including `pdflatex` and `biber`. It makes sure that the individual tools are run in the right order and often enough. If you (or your editor) run the individual tools remember that it might be necessary to compile multiple times. For example, without `latexmk` this document requires the following sequence of commands in order to get all citations and references right:

```
pdflatex seminar
bibtex seminar
pdflatex seminar
pdflatex seminar
```

Note that as long as no citations of bibliography data change between builds (and you do not delete any of the generated files) two runs of `pdflatex` are enough to rebuild the document.

Makefile This template also contains a `makefile`, so that this document can be compiled by running `make`. Moreover `make watch` can be used to recompile the document whenever it changes. The Makefile is also configured with Synctex, which enables some PDF-viewers to jump from the document to the correct position in the \LaTeX code.

2.1 Editors

There are a number of IDE-like \LaTeX editors (none of which fully satisfies the authors). Most are basically a text editor with some gimmicks, but those can already facilitate writing \LaTeX documents significantly.

TeXStudio standalone editor with built-in pdf viewer (with source \leftrightarrow pdf sync), automatic build, instant preview ...

Kile KDE-user's default choice

TeXlipse Eclipse plugin for \LaTeX which is reasonably powerful.

Emacs with AUCTeX If you like Emacs, this is what you want to use.

TeXShop for MAC users

Your favorite text editor probably has \LaTeX syntax highlighting at least. Look for plugins that add more features.

3 More on \LaTeX

A whole collection of tutorials and further information is collected by the German \TeX user group [2].

We consider the following material especially useful:

- “The Not So Short Introduction to $\text{\LaTeX} 2_{\epsilon}$ ”, a basic tutorial for \LaTeX .
- The \LaTeX cheat sheet at <http://www.stdout.org/~winston/latex/>.
- Overviews on typesetting mathematics are available on
 - Wikibooks (en.wikibooks.org/wiki/LaTeX/Mathematics),
 - Wikipedia (en.wikipedia.org/wiki/Help:Displaying_a_formula) and

```

1 def sort[T](list: List[T])(implicit ord: Ordering[T]): List[T] = {
2   list match {
3     case Nil => Nil
4     case x :: xs =>
5       // partition list based on pivot element x
6       val (lo, hi) = xs.partition(ord.lt(_, x))
7       sort(lo) ++ (x :: sort(hi))
8   }
9 }

```

Figure 1: An example algorithm. Do you recognize it? Line 6 is crucial!

– Mathematics Stack Exchange (meta.math.stackexchange.com/q/5020).

- Access package documentation by executing `texdoc packagename` on shell (needs TeX Live).
- Find symbols by drawing them with Detexify.
- Command-line program `lacheck` is often better at detecting syntactical errors than `pdflatex` itself; in fact `pdflatex`'s error messages are hardly ever useful.
- A very comprehensive and up-to-date repository of L^AT_EX packages is available on CTAN.
- If you want to use other fonts, visit the L^AT_EX Font Catalogue. However, fonts can be a fickle issue in L^AT_EX, see also tex.stackexchange.com/q/25249.
- For any L^AT_EX question, visit tex.stackexchange.com.

Note that since T_EX and L^AT_EX are decades old and many features have been added by packages over the years, lots of information on the internet are obsolete by now. A famous example are umlauts and other “non-standard” character (äöüßåø) which used to require special commands; nowadays these work out of the box with Unicode source files. See the source of this document to see how it is done.

In case of doubt, we recommend visiting TeX – LaTeX Stack Exchange; not only does the site cover many problems already but the community is very knowledgeable in all things L^AT_EX.

There are also some particular features we consider noteworthy.

- Referencing (things which have a `label`) and citing¹. For instance,
 - “`See Section~\ref{sec:editors}`” compiles to “See Section 2.1” and
 - “`a tutorial~\cite{danteintro}`” compiles to “a tutorial [2]”.

¹Citations are defined in `references.bib` as BibTeX entries. You can find BibTeX entries on most web-pages where you can access papers and articles.

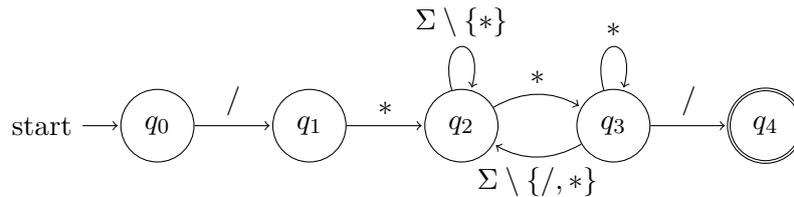


Figure 2: Example for a TikZ image; here using the `automata` library.

There are more commands (and packages) for finer control, e. g. `\eqref`, `\pageref`, `\nameref` and `\textcite`.

The most convincing reason to use these is probably that – since links are computed given the current source – numbering, names and links survive rearrangements of the material.

- You can define custom macros in the preamble. For example, if you include `\newcommand{\N}{\mathbb{N}}` then `“$n \in \N$”` compiles to `“ $n \in \mathbb{N}$ ”` (as would `“$n \in \mathbb{N}$”`).
- (Pseudo-)Code can be typeset by a number of packages, e. g. `listings`. See Listing 1 for an example.
- With the `todonotes` package it is easy to leave little reminders in your document.
- TikZ is a great package for creating all sorts of graphics, see e. g. Figure 2. Find an extensive list of examples (with sources) on texample.net/tikz/examples. Your first graphics in TikZ *will* be time-eaters, but with some practice simple sketches are done in minutes. With a little bit of extra tuning, the same sources produce figures of unsurpassed quality.
- You can use document class `beamer` for creating presentation slides. It introduces some special syntax; best check the documentation and/or search the web.

Should we explain more?

4 General research tools

Google Scholar & backwards search

Google Scholar can be used to find papers and books of all kinds and often has links to PDF versions that are freely available, e. g., on authors’ websites. Moreover, Google Scholar can be used for finding other relevant articles in a field by backward-searching for papers that reference a given paper; see screenshot in Figure 3. Google Scholar has some basic support for exporting entries to BibTeX; the results often need some manual polishing, though.

dblp is a searchable index of computer science bibliography with BibTeX entries for all articles.

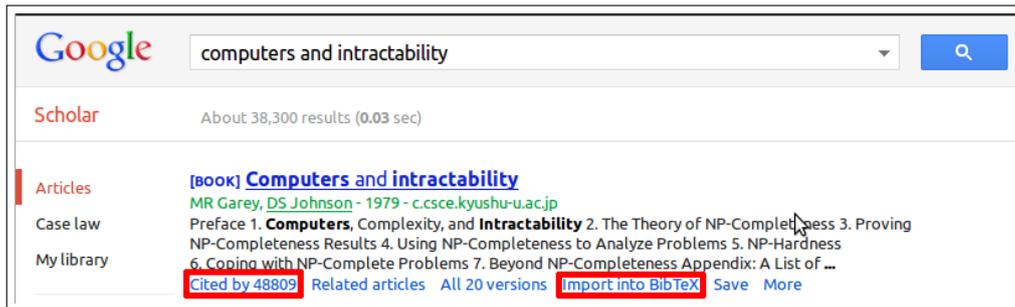


Figure 3: Some Google Scholar features we would like to highlight: back-references (find all papers that cite a given article) and BibTeX export (you may need to enable this in Google Scholar settings).

Mendeley is a (free) web service for storing references (including PDFs).

The collected references can be exported to BibTeX.

aspell or some other spell checker.

As a general rule, typos that can be found by running a spell checker are unacceptable.

Git or some other version control system (like Subversion)

Very useful for keeping old versions around and highly recommended. It works best on text files, but can also handle binary files. A comprehensive introduction is available as free ebook [1].

Git repositories can easily be cloned, e. g., onto an external drive for backup. That way, you get incremental backups of your work basically for free.

As Git (and all other versioning systems) typically operate in a *line-based* way, it is very advisable to break your text into short lines (≈ 100 character). For documents that are edited collaboratively, it has been good practice to start a new line for each sentence (or even sub-clause)².

References

- [1] Scott Chacon. *Pro Git (Expert's Voice in Software Development)*. Apress, 2009.
- [2] Deutschsprachige Anwendervereinigung TeX e. V. *Erste schritte mit tex, latex und co*, 2014.

²Recall that a newline is treated by L^AT_EX like an ordinary space. Only empty lines indicate a new paragraph.