



Manual for \LaTeX class ctuthesis

Tomáš Hejda *et al.*

Acknowledgements

We thank the CTU in Prague for being a very good *alma mater*.

Declaration

I declare that this work is all my own work and I have cited all sources I have used in the bibliography.

Prague, April 16, 2015

Prohlašuji, že jsem předloženou práci vypracoval samostatně, a že jsem uvedl veškerou použitou literaturu.

V Praze, 16. dubna 2015

Abstract

This manual shows how to use the ctuthesis L^AT_EX class, what are the requirements, etc.

Keywords: manual, degree project, L^AT_EX

Abstrakt

Tento manuál představuje L^AT_EXovou třídu ctuthesis, její použití, požadavky na systém atd.

Klíčová slova: manuál, závěrečná práce, L^AT_EX

Překlad názvu: Manuál ke třídě ctuthesis pro L^AT_EX

Contents



Figures

Tables



Chapter 1

Introduction

In this manual, we want to explain how to use the class `ctuthesis`. The manual consists of three main parts:

- basic user guide;
- reference manual;
- guide for template authors.

Students and other ‘users’ of the class should carefully read the basic user guide (Chapter ??), and visit the reference when needed (Chapter ??).

Why was this project created

The `ctuthesis` L^AT_EX template was written by Tomáš Hejda because there is wider demand for it at CTU. The aim has been to provide a L^AT_EX tool to students and their advisors for writing qualification theses (*e.g.*, bachelor, master, Ph.D.). L^AT_EX as a tool for professional typesetting has had a significant group of users at CTU. The `ctuthesis` template can be selected and used voluntarily, *i.e.*, if the student and the advisor agree on this.

There were several more local attempts in creating L^AT_EX templates for qualification theses at CTU. Let us give only two examples. First, the package `ctu`, <https://github.com/kyrcore/ctu-latex>¹. Its development seems to be interrupted since 2013. Second, the package `CMPthesis` written and maintained up to now by Vít Zýka, which is provided only internally at <http://www-cmp.felk.cvut.cz/proceedings/TeX/texmf-cmp/tex/latex/cmpthesis/cmpthesis.pdf>.

Petr Olšák from Department of Mathematics FEE came with the initiative to create a T_EX package for qualification theses and offered it to the whole CTU in 2013, see <http://petr.olsak.net/ctustyle.html> (in Czech). The package is less spread because there are fewer T_EX users than L^AT_EX users at CTU.

¹All URLs in this section were retrived on March 15, 2015.

The `ctuthesis` L^AT_EX package, described in this document, was inspired by P. Olšák's T_EX package. The aim is to preserve very similar graphical look. T. Hejda volunteered to write a L^AT_EX package with similar properties as P. Olšák's `ctustyle` package. P. Olšák has supported this initiative.

The small team around the `ctuthesis` package (Tomáš Hejda, Václav Hlaváč, Petr Olšák) has attempted persuading CTU leaders to support this package and help offering it at university-wide scale for the voluntary use. As many students and supervisors use other publishing/typesetting tools, such as Microsoft Word, the mentioned team encourages someone to write a template, maybe with similar outer look to P. Olšák's T_EX template `ctustyle` and T. Hejda's L^AT_EX template `ctuthesis`.

Chapter 2

User Guide

2.1 Installing the class

The class comprises the files listed in Table ?? . The class is build over the L^AT_EX3 bundle, which is a project under development and new programming features are added regularly. This means that the class works as is only with T_EX Live 2014. If you have an eariler version of L^AT_EX distribution, you have basically two options:

File name	Description
<code>ctuthesis.cls</code>	Class file itself
<code>ctu_logo_blue.pdf</code>	Logo of the university; this file is subject to different license and copyright than the rest of the class files
<code>ctuthesis.ist</code>	Nicely looking style for <code>makeindex</code>
<code>latex3-tl2014.zip</code>	ZIP file containing the “good” version of L ^A T _E X3
<code>ctuman.pdf</code>	This manual
<code>ctuman-*.tex</code>	Source code to this manual
<code>ctuth-core.tex</code>	Core definitions for the class; users do not need to ever load this file manually
<code>ctuth-pkg.tex</code>	Package loading code for the class; users do not need to ever load this file manually
<code>ctuth-names.tex</code>	Language-dependent part of the class; users should not need to modify this file or load it manually
<code>ctuth-templates.tex</code>	Templates for various parts of the document; users may look inside this file and see how things are done; this is discussed in more detail in Chapter ??

Table 2.1: The files of the class `ctuthesis`.

1. Install T_EX Live 2014 or newer; it is not difficult and it will save you all the troubles. The installing guide is on TUG's website at <http://tug.org/texlive/>.
2. Install just the new version of L^AT_EX3. This is tested to work on T_EX Live from version 2010. The bundle is attached as a ZIP file, it is sufficient to extract all the files in the working directory.

2.2 Minimal example

The minimal example of a Bachelor project at the Nuclear Faculty looks like this:

```
\documentclass{ctuthesis}

\ctusetup{
  xdoctype = B,
  xfaculty = F4,
  mainlanguage = english,
  titlelanguage = english,
  title-english = {Planting Uranium},
  title-czech = {Sázení uranu},
  department-english = {Department of Mathematics},
  author = {Lolek Pilný},
  supervisor = {Prof. Krutoš Spravedlivý, CSc.},
  supervisor-address = {Pěstitelský ústav, \\ Zářivá 232, \\
    12000 Praha 2},
  month = 5,
  year = 2330,
}

\ctuprocess

\begin{abstract-english}
We develop \ldots
\end{abstract-english}

\begin{abstract-czech}
Rozvíjíme \ldots
\end{abstract-czech}

\begin{document}

\maketitle

\chapter{Introduction}
```

```

Foo bar

\chapter{Conclusion}

Lorep ipsum \cite{doe}

\begin{thebibliography}{1}

\bibitem{doe} J. Doe. \emph{Book on foobar.} Publisher X,
2300.

\end{thebibliography}

\end{document}

```

At the first sight, it may look complicated, but the basic structure is pretty simple:

```

\documentclass{ctuthesis}
\ctusetup{
  < key = value settings >
}
\ctuprocess

< abstracts and other frontmatter text here,
  and standard preamble contents >

\begin{document}

\maketitle

< the whole text here >

\end{document}

```

2.3 Setting up the class

This section lists all various settings that can be made to the class. The ones marked ‘✖’ are *always mandatory*. The ones marked ‘⊠’ are *mandatory in almost all cases, with some small exceptions*. For most of them, no errors are issued if they are missing, though.

2.3.1 Options to \documentclass

The class `ctuthesis` itself takes an optional argument `draft` that sets some draft options (the ones common in \LaTeX). It also takes the option `oneside`

`documentclass`
`draft`
`oneside`

which switches to one-sided mode; however, *we strongly advice against using this option.*

2.3.2 Parameters to \ctusetup

ctusetup The parameters can be divided into several groups.

Text fields. Some of the fields — like the theses title — are language-dependent. They are marked `key-<lang>` below, and it means that multiple keys exist: `key-english`, `key-czech` and `key-slovak`.

author author — author's name, including titles.

supervisor supervisor — supervisor's name, including titles.

supervisor-address supervisor-address — supervisor’s affiliation and address; use \\ to format it with linebreaks.

supervisor **supervisor-specialist** — supervisor-specialist’s name, including titles.

year   year,  month, day — numerical values of the date of theses publishing.

month

day

day

■ title-<lang> — title of the thesis.

title-<lang>

subtitle-<lang> ■ **subtitle-<lang>** — subtitle.

university-<lang> ■ **university-<lang>** — name of the university (is preset for CTU).



university-endl-<lang> ■ **university-endl-<lang>** — name of the univesity broken in 4 lines using \\.

faculty-<lang> ■ faculty-<lang> — name of the faculty (set up automatically, see xfaculty below).

`facultynum` ■ **facultynum** — standard number of the faculty (set up automatically, see `xfaculty` below).

`doctype-<lang>` ■ `doctype-<lang>` — type of the document/thesis (set up automatically, see `xdoctype` below).

department-<lang> department-<lang> — name of the department.

fieldofstudy-<lang>   fieldofstudy-<lang>, subfieldofstudy-<lang> — field of study.

keywords-<lang> keywords-<lang> — comma-separated list of keywords.

General fields.

mainlanguage	■ ✗ mainlanguage — main language of the thesis; the choices are <code>czech</code> , <code>english</code> and <code>slovak</code> .	
titlelanguage	■ titlelanguage — language in which the title pages are typeset (defaults to <code>mainlanguage</code>).	
secondlanguage	■ secondlanguage — language of the second abstract (set up automatically so that two abstracts appear: one in English and one in Czech, unless the thesis is in Slovakian, in which case the two abstracts are in English and in Slovakian).	
	■ otherlanguages — allows more languages to be loaded by <code>babel</code> package, accepts a comma-separated list.	otherlanguages
	■ ✗ xfaculty — takes values <code>F1</code> , <code>F2</code> , ..., <code>F8</code> and sets up the faculty name and number automatically.	xfaculty
	■ ✗ xdoctype — takes values <code>B</code> (bachelor), <code>M</code> (master) or <code>D</code> (dissertation) and sets up the document type automatically; for non-thesis-like documents, use <code>doctype-<lang></code> .	xdoctype
	■ ✗ specification-file — the path to the specification file (usually something like <code>zadani.pdf</code> , the current directory is searched for the file of course).	specification-file
	■ front-specification — if set to <code>true</code> , the specification will appear just after the titlepage (defaults to <code>false</code>).	front-specification
	■ front-list-of-figures , front-list-of-tables — says whether the list of figures/tables will be set in the frontmatter (default to <code>true</code>)	front-list-of-figures front-list-of-tables
	■ monochrome — switches to optimized monochrome setup, in case the work will not be printed in colour.	monochrome
	■ savetoner — disables background and other too dark things to save toner while printing drafts.	savetoner

Default package loading. The class attempts to set up some packages nicely if the user wants to use them. So for instance, the package `listings` can be pre-set to include the blue background of all listings. In order to leave the user the chance to mess with things “his own way”, options are offered to enable or disable certain packages. The format of the option name is always `pkg-<packagename>` and it accepts either `true` or `false`

- **pkg-hyperref** (default: `true`) — sets up all the fields of the PDF document.

`\subsection`, `\subsubsection`. These ought to follow the hierarchy from top to bottom. There is also `\paragraph` and `\subparagraph` which can be used as lowest-level sectioning commands. Long works can be divided into parts using `\part`; its numbering is independent of the other ones. All these commands take standard optional argument and star; however, we advice against using starred sections.

The appendices are introduced by command `\appendix`. Then, each appendix is a chapter, so the standard `\chapter` command ought to be used. In appendix, even chapters started by `\chapter*` such as the bibliography and the index are numbered. (In a document without appendices, these should be unnumbered. Once appendices are used, all indexes and alike should be standard numbered appendices.) For unnumbered appendices, `\chapter**` can be used.

If the document is divided into parts, the appendices are introduced by a part-like title “Appendices”, both in the document and in the table of contents. This can be suppressed by using `\appendix*` (keeps the line in the table of contents) or `\appendix**`.

If the project specification is to be included as an appendix, it can be done by `\ctutemplate{specification as chapter}`.

There is no true backmatter behaviour defined in the class. The only thing that would belong to the backmatter is the colophon (*‘tiráž’*). Users are advised to use `\cleardoublepage*` `\thispagestyle{empty}` and then typeset the colophon to their liking.

2.4.3 Bibliography

There are three ways how to include the bibliography:

1. Using Biblatex. In that case, setting up `pkg-biblatex = true` is recommended. The bibliography is then included using `\addbibresource` and `\printbibliography`.
2. Using BIBTEX. No special setting is needed, and the bibliography is included using `\bibliographystyle` and `\bibliography`.
3. By direct input. The standard environment `{thebibliography}` can be used for this.

We kindly ask users to consult the manuals to the specific packages for their usage.

2.4.4 Index

To generate the index, use `pkg-makeidx = true`, and put `\printindex` where the index should appear. We provide an in-house style for the index called `ctuthesis.ist`. The index generation is dependent on calling the external tool `makeindex`, in a similar way in which the bibliography generation depends on `bibtex` or `biber`. The proper way how to call this tool is the following:

```
makeindex -s ctuthesis.ist <filename>
```

ctuthesis.ist where <filename> is the root name of your main project file (without `.tex`).

2.4.5 List of notation. Nomenclature

We do not preset any special environment for typesetting appendices such as list of notation or nomenclature. One of the possible standard way how to do it is with the help of the packages `tabularx`, `array` and `booktabs` as follows:

tabularx
array
booktabs

```
\documentclass{ctuthesis}
\ctusetup{ mainlanguage = english }
\ctuprocess
\usepackage{tabularx, array, booktabs}
\begin{document}

\appendix

\chapter*{List of Notation}

\noindent
\begin{tabularx}{\linewidth}{
  { 1 >{\raggedright\arraybackslash}X }
\bfseries Symbol & \bfseries Meaning \\ \Midrule
$\alpha$ & The angle of attack \\
$\mathbb{R}$ & The real numbers \\
\end{tabularx}

\end{document}
```

ltxtable In the case the notation is longer than one page, the package `ltxtable` or
ltablex `ltablex` can be used.

2.4.6 Floating objects: Figures and tables

figure Floats are input either as `{figure}` or `{table}` environment. All floats are
table automatically typeset centered, so a typical code for a figure is the following:

```
\begin{figure}
\includegraphics[width=0.8\linewidth]{mygraphicfile.pdf}
\caption{We depict a foo-bar here.}
\label{fig:foobar}
\end{figure}
```

For a table, it is the following:

```
\begin{table}
\begin{ctucolortab}
\begin{tabular}{cc}
```



```
\bfseries Foo & \bfseries Bar \\\Midrule
foo1 & bar1 \\\
foo2 & bar2
\end{tabular}
\end{ctucolortab}
\caption{Table of foo-bar.}
\label{tab:foobar}
\end{table}
```

Note the usage of the environment `{ctucolortab}` to make the table get a blue background, and the usage of `\Midrule` to get a horizontal line in the table (a thinner line can be obtained using `\midrule`). Also note the deliberate absence of vertical rules.

ctucolortab
Midrule
midrule

2.4.7 Mathematics

All standard concepts for typesetting mathematics are working as in any other \LaTeX class. We only emphasize how the class behaves in relation to theorem-like environments and proofs. The default behaviour is that the package `amsthm` is loaded and theorem styles `plain`, `definition` and `note`. Theorems (plain style) are by default slanted rather than italic, should italic theorems be preferred, use `plainit`. However, no theorem-like environments are predefined, everybody can do this to their liking. The preamble declarations could look like this:

```
\theoremstyle{plain} % not necessary, it is the default
\newtheorem{theorem}[Theorem][chapter] % per-chapter numbering
\newtheorem{lemma}[theorem]{Lemma} % use the same counter for
% all of them
\newtheorem{proposition}[theorem]{Proposition}

\theoremstyle{definition}
\newtheorem{definition}[theorem]{Definition}
\newtheorem{example}[theorem]{Example}

\theoremstyle{note}
\newtheorem{remark}[theorem]{Remark} % numbered remark
\newtheorem*{remark*}{Remark} % unnumbered remark*
```

The environment `{proof}` is provided by default. All theorem handling can be disabled by `\ctusetup{ pkg-amsthm = false }`.

2.4.8 Colors

The class uses internally many named colors. For the general use, two non-standard colors are provided:

■ `ctubblue` `ctubblue`;

ctubblue

■ `ctulightblue` `ctulightblue`.

`ctulightblue`

The colors are used for the headers, rules in the titles, title texts, backgrounds for listings and tables etc.

■ 2.5 Features not described in this manual. Troubleshooting

I may happen that some of the features of the class are not described in this manual. If the users struggle with anything, they can contact the class authors at tohecz@gmail.com or via the GitHub repository github.com/tohecz/ctuthesis.



Chapter 3

Creating and modifying templates

We apologize, this chapter of the manual is still under construction.