

Guide on Using umalayathesis

version 1.3

Lim Lian Tze

liantze@gmail.com

8 September, 2017

umalayathesis is a \LaTeX class for authoring theses that fulfil formatting specifications required by Universiti Malaya (UM), Malaysia. The thesis preparation guide can be accessed at <http://bit.ly/2xaYpzN>.

A sample `thesis.tex` is included in the package, which I recommend you modify for your own thesis write-up. (You can rename it, but I'll stick with the file name '`thesis.tex`' throughout this guide.)

1 Before You Start

1.1 Pre-requisite Packages

To streamline things, install the following packages first before using umalayathesis or compiling the example `thesis.tex`:

- | | | |
|--------------|----------------|------------------|
| • memoir | • longtable | • etoolbox |
| • apacite | • supertabular | • enumitem |
| • glossaries | • hyperref | • lipsum (for |
| • babel | • xfor | thesis.tex only) |

1.2 Files

Here's a quick list of the files required when writing your thesis with the umalayathesis class. Easiest way to go about things is to put all the files in the same directory. (See Section 2 for more details.)

- **umalayathesis.cls**, the \LaTeX class file implementing the UM thesis formatting requirements.
- A “main driver” `.tex` file of your thesis, analogous to `int main()` or `public static void main(String[])`. You can name this file anything you like; it is known as `thesis.tex` in this guide. (See Section 2.)

This is the *only* file that you should run the processing tools on!

- A `.tex` file containing your thesis abstract. (See Section 2.5.)
- `.tex` files containing your thesis chapters and appendices, one chapter per file. (See Section 2.7 and Section 2.8.)

- A `.bib` file containing your references and publications. (See Section 2.9).
- A `.tex` file containing your glossary. (See Section 2.11).

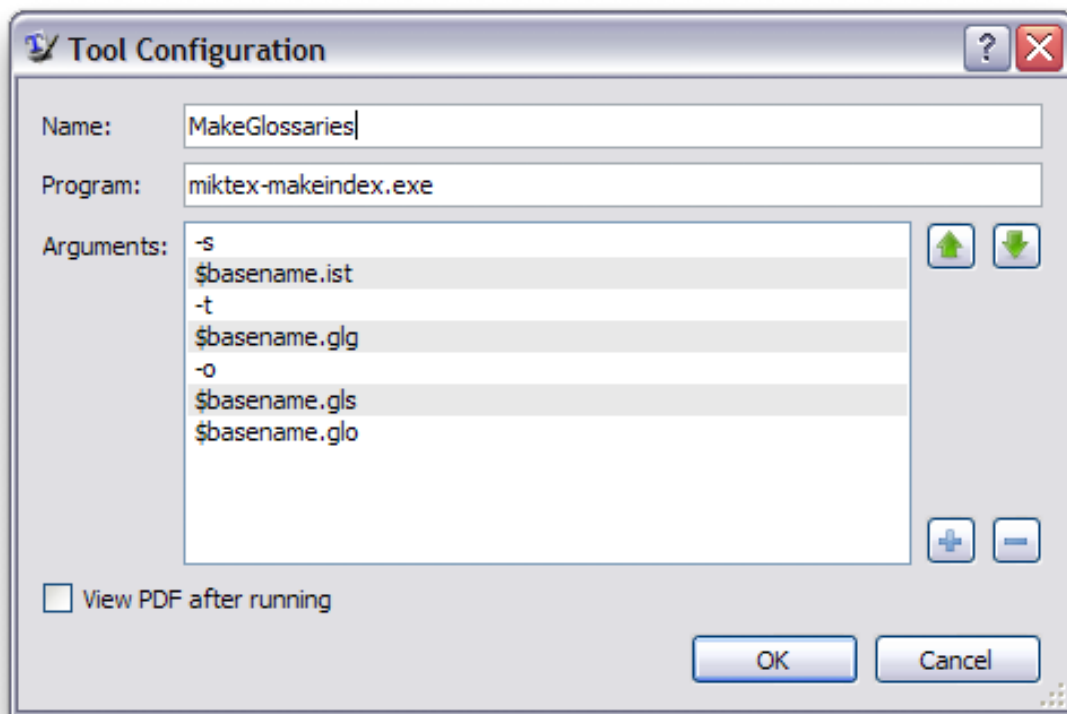
1.3 TeXworks Configuration

Assuming TeXworks is your \LaTeX editor of choice on Windows, you will probably want to configure it so that you can process your glossary and list of own publications from within TeXworks.

(You can always, of course, opt to run the relevant commands from the command line prompt, or adapt these configurations for other editors and operating systems: I have tested on Windows XP/7, Ubuntu and Mac OS X.)

1.3.1 Tool Configuration for Generating the Glossary

Access the TeXworks menu Edit \triangleright Preferences... \triangleright Typesetting. Add a new processing tool called “MakeGlossaries”. Configure it as shown below:



Now **repeat the above step** for another similar tool called “MakeAcronyms”, but replace “thesis.glg” with “thesis.alg”; “thesis.gls” with “thesis.acr”; “thesis.glo” with “thesis.acn”.

On Linux and Mac systems, these are equivalent to the command lines

```
makeindex -s <base>.ist -t <base>.glg -o <base>.gls <base>.glo
```

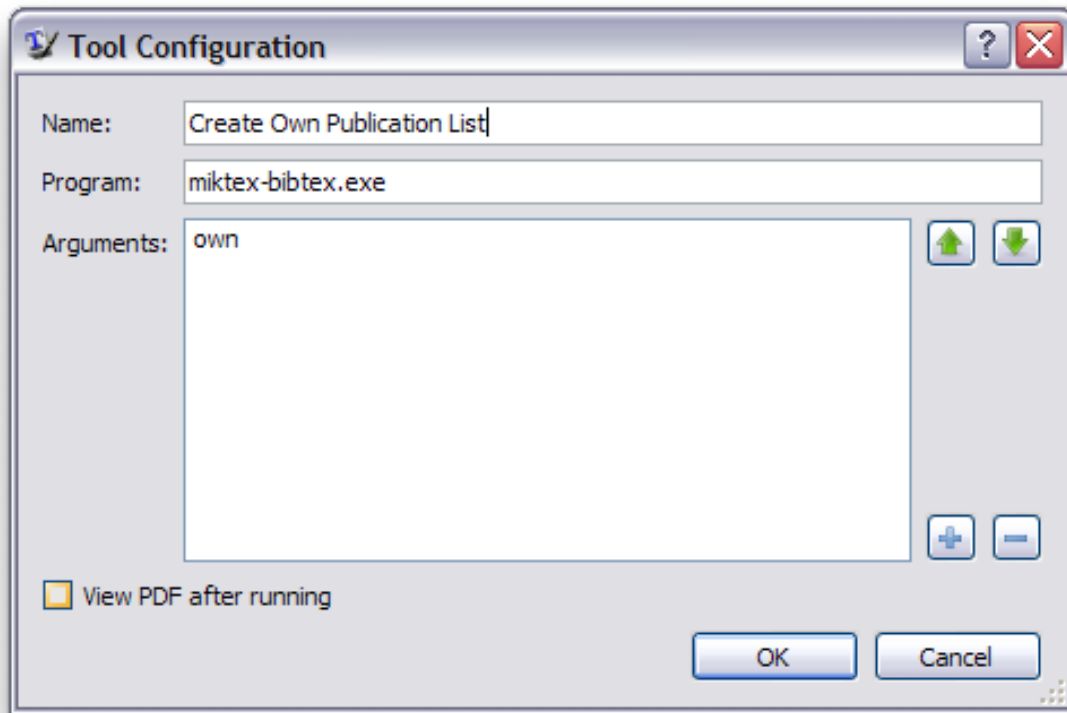
If you have Perl installed (likely if you’re using Linux or Mac), you can just run

```
makeglossaries <base>
```

and it’ll process both the acronyms and the glossaries.

1.3.2 Tool Configuration for Generating the List of Publications

Now add a new processing tool called “Create Own Publication List” (or some other name). Configure it as shown below:



On Linux and Mac systems, these are equivalent to the command lines

```
bibtex own
```

1.3.3 Compiling thesis.tex

The processing tools should be run on `thesis.tex` in the following sequence:

1. `pdflatex + bibtex + makeindex`
2. `MakeGlossaries + MakeAcronyms + Create Own Publication List`
3. `pdflatex + bibtex + makeindex` (twice)

You will need to run ‘MakeGlossaries’ and/or ‘MakeAcronyms’ again if you add and use a *new glossary or acronym entry*.

1.4 Printing from Acrobat Reader

Remember to set the **paper size** to **A4** and **page scaling** to **None** in the Print dialog, otherwise the margins would be incorrect.

2 Using the umalayathesis Class

2.1 Activation

To ‘activate’ the class, make sure your main document file (e.g. `thesis.tex`) starts off with `\documentclass{umthesis}`:

```
\documentclass{umalayathesis}  
\usepackage{graphicx}  
\usepackage{... other packages you need}
```

This will set up the page margins, paragraph spacing, indents, page numbers, font face and size, citation and bibliography format, amongst other things.

(There is an experimental `bahasam` document class option that sets up everything for Bahasa Malaysia; but at present `apacite` has not yet been localised to Bahasa Malaysia yet.)

2.2 Author Information

You need to provide some author information in the preamble. Example lines from `thesis.tex`:

```
\author{Lim Lian Tze}  
\title{My Ground-breaking Research}  
\submissionyear{2012}  
\faculty{Faculty of Amazing Research}  
\degree{Doctor of Philosophy}}
```

These information are needed to generate the preliminary pages.

2.3 Preliminary Pages

Once in the main document body, `\frontmatter` sets up the, well, front matter. This include setting the page numbers to lower-case Roman numerals.

`umalayathesis` can generate the cover page, title page and original literary work declaration page with the following lines (included in `thesis.tex`):

```
% \makecoverandtitlepage{\mastercoursework}  
% \makecoverandtitlepage{\mastermixedmode}  
% \makecoverandtitlepage{\masterresearch}  
\makecoverandtitlepage{\doctoralresearch}  
% \makecoverandtitlepage{\doctoralmixedmode}  
\declarationpage
```

Please *uncomment* the correct `\makecoverandtitle` line to generate the correct statement on the title page.

2.4 Acknowledgements

This is provided using `\acknowledgements`:

```
\acknowledgements{I would like to thank my parents, my family, my  
supervisor...}
```

2.5 Abstract

Write your abstracts in separate files (`sample-abstract.tex` for the English abstract and `sample-msabstract.tex` for the Malay abstract in this example), and include them in `thesis.tex` like this:

```
\abstractfromfile{sample-abstract}
\msabstractfromfile{sample-msabstract}
```

2.6 Table of contents, List of figures and tables

These are auto-generated by the following lines (included in `thesis.tex`):

```
{\clearpage\SingleSpacing
\tableofcontents\clearpage
\listoffigures\clearpage
\listoftables\clearpage}
```

2.7 Main Chapters

I highly recommend that each chapter be written in a separate file. For example, `chap-intro.tex` has the contents

```
%!TEX ROOT=thesis.tex
\chapter{Introduction}
This is the introduction chapter.

\section{Problem Background}
We study the...
```

And `chap-litreview.tex`:

```
%!TEX ROOT=thesis.tex
\chapter{Literature Review}
We review the state of the art in...

\section{Early Approach}
Researchers first attempted to...
```

In `thesis.tex`, these chapter files are included with the following lines:

```
\mainmatter % signal start of main chapters
\input{chap-intro} % no .tex extension!
\input{chap-litreview}
\input{...}
```

The `%!TEX ROOT=thesis.tex` indicates to TeXworks (and also TeXshop on the Mac) that `chap-xxx.tex` are ‘sub-files’ of `thesis.tex`. This means if you hit `Ctrl` + `T` when you are editing `chap-xxx.tex`, `thesis.tex` will get compiled instead. Neat eh?

2.8 Appendices

Again, I recommend keeping each appendix chapter in its own file e.g. `app-uml diagram.tex`:

```
%!TEX ROOT=thesis.tex
\chapter{UML Diagrams}
...
```

And in `thesis.tex`:

```
\appendix % signal start of appendices
\input{app-uml diagram}
\input{...}
```

2.9 Citations and Bibliography

umalayathesis uses the apacite package to format citations and bibliography in the APA style.

Here are some useful variants of the `\cite` command; see the apacite manual for full list.

- `\cite{Lim:2009}` → (Lim, 2009)
- `\citeA{Lim:2009}` → Lim (2009)
- `\citeNP{Lim:2009}` → Lim, 2009 (no parenthesis)
- `\cite<see>[p.~7]{Lim:2009}` → (see Lim, 2009, p. 7)
- `\citeauthor{Lim:2009}` → Lim
- `\citeyear{Lim:2009}` → (2009)

In `thesis.tex`, these lines will print the bibliography list:

```
\backmatter % signal start of back matter
\bibliography{bibfile} % bibliography file name without .bib extension
```

2.10 List of Publications

First, make sure that you enter details about your own publications in your `.bib` file. Then in `thesis.tex`, search for the following line:

```
\nociteown{Lim:2009}
```

Replace the BibTeX key between the curly braces with that of your own publication. If you have more than one publications, simply separate them with commas inside the curly braces, like this:

```
\nociteown{lim:tang:2004,Lim:2009}
```

If you add more publications to `\nociteown` later, you will need to run `pdflatex` and the Create Own Publication List tool that you configured earlier, then 2 more runs of `pdflatex` for the list to appear properly.

2.11 Glossary

You can maintain a consistent glossary and acronym list using the glossaries package. It also supports acronym expansion on first mention! ☺

First, define your acronyms and terms in a separate file e.g. `myacronyms.tex`:

```
% \newglossaryentry{label}{name={term},description={explanation}}
\newglossaryentry{lexicon}{
name={lexicon},
description={The vocabulary of a language, including its words and
expressions. More formally, it is a language's inventory of lexemes}
}

% \newacronym[description={explanation}]{label}{abbrv}{full form}
\newacronym
[description={single word or words that are grouped in a language's lexicon}]
{LI}{LI}{lexical item}

\newacronym[description={The application of computational linguistics
principles to problems}]
{NLP}{NLP}{Natural Language Processing}

% when the plural form is irregular, specify firstplural and plural
\newacronym
[firstplural={parts of speech}, plural={POS},
description={linguistic category of lexical items}]
{POS}{POS}{part of speech}
```

Loading the glossary and acronym list, and later printing the list of acronyms and glossary in `thesis.tex`:

```
% Must be loaded BEFORE \begin{document}!
\loadglsentries{myacronyms}
\begin{document}
...
% List of acronyms is between list of tables and list of appendices
\listofacronyms\clearpage
...
\bibliography{bibfile}
% Glossaries is placed AFTER the bibliography
% (only entries that are actually used in the text will be listed)
\printglossary
...
```

To mention them in the text (i.e. `chap-xxx.tex` etc):

Let's talk about `\acp{LI}` and `\acp{POS}` in `\ac{NLP}`. I mention again `\acp{LI}`. We will also talk about `\glsplural{lexicon}`.

Notice how the acronyms are expanded on first use, as well as the use of `\glsplural` and `\acp` for plurals:

Let's talk about lexical items (LIs) and parts of speech (POS) in Natural Language Processing (NLP). I mention again LIs. We will also talk about lexicons.

You will need to run `pdflatex`, the `MakeGlossaries` and `MakeAcronyms` tool that you configured earlier, then 2 more runs of `pdflatex` for the glossaries to appear properly.

Use `\Gls`, `\Glsplural`, `\Ac`, `\Acp` etc. if you need to capitalise the first letter of your terms at the beginning of sentences.