

Typesetting a Thesis/ Dissertation Using Mateti's L^AT_EX Example

(Not true. Delete this line.)

A thesis submitted in partial fulfilment
of the requirements for the degree of
Master of Science in Computer Engineering

By

PRABHAKER MATETI

B.S., Osmania University, 1970

M.Tech., Indian Institute of Technology at Kanpur, 1972

Ph.D., University of Illinois at Urbana-Champaign, 1976

2004

Wright State University

Dayton, Ohio 45435-0001

WRIGHT STATE UNIVERSITY
SCHOOL OF GRADUATE STUDIES

March 30, 2003

I HEREBY RECOMMEND THAT THE THESIS PREPARED UNDER MY SUPERVISION
BY Sripriya Subramanian ENTITLED Sniffing the Ethernet with High Quality Tools BE
ACCEPTED IN PARTIAL FULFILLMENT OF THE REQUIREMENTS FOR THE DEGREE OF
Master of Science in Computer Science.

Prabhaker Mateti, Ph. D.
Thesis Director

Forouzan Golshani, Ph. D.
Department Chair

Committee on
Final Examination

Prabhaker Mateti, Ph. D.

Professor B, Ph. D.

Professor C, Ph. D.

Joseph F. Thomas, Jr., Ph.D.
Dean, School of Graduate Studies

ABSTRACT

Subramanian, Sripriya. M.S. Department of Computer Science and Engineering, Wright State University, 2003. Sniffing the Ethernet with High Quality Tools.

An abstract is typically about one page. The following is an example.

This thesis is a study of software quality in the limited context of a class of network software tools called *sniffers*. Sniffers are network monitoring tools used in the administration of network security. We analyzed five of the hundreds of the existing sniffers to determine the causes of poor quality and the methods to eliminate the problems. We subjected the five selected sniffers to both manual analysis and analysis by software quality assessment tools. We classify the 1000+ errors so discovered in these sniffers. Based on these results, we designed and implemented a new sniffer that is intended as a model of high quality sniffer. The methods applied to analyze and enhance the quality of software are studied.

Quality assessment software tools fall into two categories: 1) Static checkers and 2) Dynamic checkers. Static checkers analyze the source code. Dynamic checkers stand as guards during runtime. We have collected and used exhaustively the checker tools that are in the open source archives. In the course of our use, the quality of the software quality tools is itself analyzed.

This thesis contributes several case studies of open source software projects. It also contributes a new distributed sniffer to the open source. The distributed sniffer can monitor multiple networks and output desired packet details. We designed our sniffer to include a MySQL based collector program, packet capture programs and viewer programs. Our sniffer supports multiple capture and viewer programs. We applied software engineering methods to eliminate the quality issues associated with software. We use the results of our analysis of existing sniffers in avoiding poor quality in our sniffer. We prepared our sniffer for manual audit using documentation tools. This new sniffer is an example of *literate programming* that is worthy of study by software engineering students.

ACKNOWLEDGEMENTS

It is typical to acknowledge a few people who helped you in the thesis work. Include such acknowledgements here.

Contents

1	Introduction	1
1.1	Advice	1
1.2	“I say so.”	1
2	Mateti’s \LaTeX Thesis Example	2
2.1	Contents of the tar File	2
2.2	Your Use of these Files	3
3	Technical Report Writing	4
3.1	Structure	4
3.2	URLs	4
3.3	Acronyms	5
3.4	References	5
4	Typewriting Conventions	6
5	Learning \TeX and \LaTeX	7
6	Routine Use of \LaTeX	8
6.1	Document Structure	8
6.2	\LaTeX -ing Twice?	8
6.3	Lists	9
6.4	Tables	9
6.5	Inclusion of Source Code of a PL	9
6.6	Figures	10
6.6.1	<code>xfig</code> Generated \LaTeX Drawings	10
6.6.2	MetaPost	10
6.6.3	Encapsulated PostScript	11

7	Recommended Software Tools	13
7.1	On Linux, Unix and Windows	13
7.2	On Linux/ Unix Systems	13
7.3	On Windows NT and above	14
8	American English	15
8.1	Spelling	15
8.2	Misused Words and phrases	15

1

Introduction

I want all my students to typeset their work using L^AT_EX. To reduce their effort in doing this, I give this article and the accompanying L^AT_EX files as an example to learn from. The details of L^AT_EX as used in these example files is sufficient to most of their documents.

1.1 Advice

“Down with MS Word. Long live LaTeX.”

The above is the underlying sentiment in this article.

I recommend various things below without justifying them. This keeps the article short.

1.2 “I say so.”

Besides, I have some dictatorial powers over my students “;-)”. And, these are computer science/engineering students. They are supposed to be able to learn new languages and tools fast! No?

2

Mateti's L^AT_EX Thesis Example

Download link: <http://www.cs.wright.edu/~pmateti/thesis.tbz>

I recommend that you use my collection of thesis files that you can download from the link given above. My own contribution here is not original. It is in pulling together various needed pieces in trying to reduce the LaTeX-learning time for my students.

2.1 Contents of the tar File

The downloaded file is a tar archive compressed with bzip2. You can untar it as in

```
tar xvfj thesis.tbz
```

Your current directory will then have a new subdirectory called Thesis/ with the following files and subdirectories in it (the dates and sizes shown may not match).

```
% ls -l Thesis
total 36
drwxr-xr-x  2 pmateti  pmateti    4096 May 24 13:07 Chapters/
drwxr-xr-x  2 pmateti  pmateti    4096 May 24 11:44 Figures/
drwxr-xr-x  2 pmateti  pmateti    4096 May 24 11:44 LaTeX/
-rw-r--r--  1 pmateti  pmateti     713 May 24 11:47 Makefile
drwxr-xr-x  2 pmateti  pmateti    4096 May 24 13:09 Tables/
-rw-r--r--  1 pmateti  pmateti   8513 May 24 11:44 thesis.bib
-rw-r--r--  1 pmateti  pmateti   1158 May 24 11:47 thesis.tex
drwxr-xr-x  2 pmateti  pmateti    4096 May 24 11:44 WSU/
```

2.2 Your Use of these Files

- The files in the directory `LaTeX` are to be left as they are; edit these only if you are fluent in LaTeX.
- Edit the files in the `WSU` directory to insert your own name etc. There should be no need to introduce additional files.
- Replace all the files in the `Chapters/`, `Figures/`, `Tables/` directories with yours – as many as you wish.
- Edit the `thesis.tex` to input (i.e., `\input{}`) the files in the `Chapters/`, `Figures/`, and `Tables/` directories.
- Use the given Makefile as a template and adjust.
- The file named `thesis.bib` should contain the references you collected in the BibTeX syntax. I have included a few references to illustrate this syntax; you can leave them in this file. In general, it can contain many more references than what you cite in the body of your thesis. The `bibtex` program examines the actual citations made, and pulls together the cited references into the text file named `thesis.bbl`.

3

Technical Report Writing

A thesis is a technical report. Read and follow a good style book (e.g., Chicago) on technical report writing. It is a good exercise to mark *this* article up! Additionally, every university has guidelines regarding layouts and required pages such as a title page an approval page.

(This article is itself poorly and hastily written. Think of it as a collection of notes. “Do as I say, not as I do! ;-”))

3.1 Structure

When you are moving from one depth to a deeper one there is always an introductory paragraph. E.g., going from a chapter heading to its first section, and from a section heading to its first subsection.

Sometimes we like to write a paragraph or two at the end of a chapter that does not belong to the last section. This is often done by introducing a vertical white space after the end of the last section that is longer than the typical spacing between consecutive sections.

There are no particular size requirements. But, a one page chapter and a two line section do look silly.

3.2 URLs

URLs are now common in CS/CEG theses. Older style guides do not cover these. URLs should be in a typewriter (`\tt`) font.

3.3 Acronyms

Acronyms are written in all-upper case. Examples: GNU, BCPL, KDE.

Linux, Unix, Pascal, Ada, Java, ... are not acronyms. LINUX should be Linux, no boldface, L in caps, the rest in lower case.

3.4 References

Bibliographic citations are done in many ways. I require my students to use the `acmtrans.bst` style. Other acceptable choices are those that include author(s) last name and year of publication in the body of the text that cites, and the end of the article list of references are sorted alphabetically by the first author's last name. Styles that produce a bracketed number, as in [5], are not acceptable to *me*. I much prefer [Hoare 1978].

4

Typewriting Conventions

You will be working with the source files of LaTeX a lot. So, it is almost as important how the text is laid out in these files as how good the printed result looks.

Sentences should be separated by two blanks. Emacs can then do sentence-level editing.

Separate the paragraphs with an empty line.

Don't let any line be longer than around 70 chars.

Section headings should appear by themselves on a line. Leave a blank line above and below these.

The following are special characters in TeX: | [] {}#&_%\$. If you must use them, you have to enclose them in either in the verbatim mode as in `\verb|#&_%$[]{}|` or provide escape sequences that you should look up.

It is a good idea to end each of your file with the line
`%-eof-`.

5

Learning T_EX and L^AT_EX

Of course, you *should* learn T_EX and L^AT_EX! But, more importantly you should produce a well-written thesis in the time you have.

The files of `thesis.tbz` should be enough if you can learn via examples.

There are many tutorials of varying length, from a few pages to hundred of pages, on the web. The authoritative books are by [Knuth 1970] and [Lamport 1970].

6

Routine Use of L^AT_EX

This article is not even a beginners introduction to L^AT_EX. Many of our students are too afraid to learn it. In exasperation, I wrote these files up as a cook book recipe.

6.1 Document Structure

The macros `\chapter`, `\section`, `\subsection`, ... are self-explanatory. These also introduce a line in the table of contents (`.toc`) file. The `.toc` file is generated at the end of L^AT_EX processing. So, to include a table of contents properly at the beginning of the document, we must latex the files twice.

The depth of sections that show up in the TOC is set in the `pnthesis.cls` file to 3. You can change this, to say 9, by including the line `\setcounter{secnumbookdepth}{9}` in `thesis.tex`.

6.2 L^AT_EX-ing Twice?

Most documents contain forward references to bookmarks. So, often we must latex the files twice. The first run will produce `*.aux` files, which are processed by other tools, including L^AT_EX in the second run.

The file named `thesis.log` contains a detailed listing of all warnings and errors that L^AT_EX has detected.

Most of us, being not as fussy as L^AT_EX is about appearance, can ignore the “underfull” and “overfull” messages.

At the top of the `thesis.tex` file, there are these two lines:

```
\ExecuteOptions{draft,openright} %% comment out this or the one below
%\ExecuteOptions{final,openright}
```

The `draft` option prints black rectangles when a line sticks out to bring your attention to it. Re-adjust your \LaTeX files so that this does not happen. When all revisions are done, you can turn off `draft`, and turn on `final` option.

6.3 Lists

LaTeX has three kinds of lists. The `description` list is often better than `enumerate` which is better than `itemize`.

Braces `\item {...}` The braces are not necessary.

Brackets `\item []` The brackets are not necessary.

6.4 Tables

CATEGORY	TCPDUMP	SNIFFIT	HUNT	DSNIFF	ETHERREAL
Word Count	218455	30938	45568	66599	1352372
Lines of Code	36085	3005	8817	9680	280068
Binary Size (Kb)	488	53	384	482	500
Cyclomatic complexity	5392	610	2029	1517	na

Table 6.1: Size Summaries of the Selected Sniffers

6.5 Inclusion of Source Code of a PL

Source code can be included in the verbatim mode. In this mode, the special characters of \TeX are produced as-is. However, the TAB characters are ignored. Indentation should therefore use just blanks.

```
#!/bin/bash
# kppp does not correctly set the PPP up. Here is the fix.
#
fix-ppp() {
    route del -net 0.0.0.0
    route add default gw 130.108.108.8
```

```

    route -n
}

```

Here is a Makefile rule that will pretty print the `.c` source code inserted into a `verbatim` mode into files that now have an extension of `.src`.

```

%.src: %.c
indent -kr -i2 -nut $<
echo \\singleSpacing \\begin\\{verbatim\\} > $@
cat $< >> $@
echo \\end\\{verbatim\\}\\doubleSpacing >> $@

```

Consider “doing” literate programming [Knuth 1984].

6.6 Figures

Figures and tables float, often ending up at the top of a page. Also, neither figures nor tables can be longer than a single page.

6.6.1 xfig Generated L^AT_EX Drawings

```

\\begin{figure}[htb]
\\centering
\\input{Figures/trie.tex}
\\caption{The trie.tex}\\label{trie}
\\end{figure}

```

The Figure 6.1 was interactively drawn using `xfig` and then exported as a `.tex` file from it. Typically, we label the figures so we can reference it, as we did it here with the label `trie`, in the text of the document.

6.6.2 MetaPost

An excellent language for drawings is `metapost`. This is like writing a program that draws. MetaPost can generate PostScript and other output. MetaPost is typically included in Linux distributions. On other platforms, you may have to ask your sysadmin to install it. <http://cm.bell-labs.com/who/hobby/MetaPost.html> The language and its implementation are in the public domain.

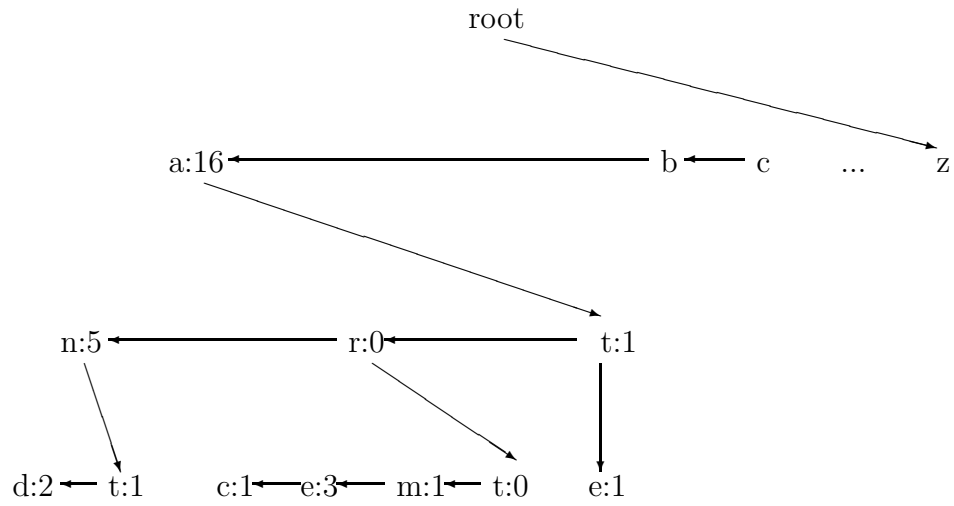


Figure 6.1: The trie.tex

6.6.3 Encapsulated PostScript

Here is an example of an .eps file. Input the epsf file just once.

```



```



Figure 6.2: The tiger.eps from GhostScript

7

Recommended Software Tools

No links are given below because they have been changing too rapidly. Search the web using google or similar.

7.1 On Linux, Unix and Windows

The tools mentioned in this section are available for all the major OS.

Use `emacs`/`xemacs` for editing TeX files. There are style files for TeX. Make sure they are loading when you begin editing a file with the extension `.tex`.

`pdflatex` can directly produce `.pdf` files. However, often it is more convenient to produce `.dvi`, then `.ps`, and finally `.pdf`. DVI-to-PS is by `dvips`, PS to PDF is by `ps2pdf`, dvi to pdf is by `dvipdf`.

OpenOffice has a vector drawing component that is quite good.

`latex2html` (<http://www.latex2html.org/>) and `TTH` (<http://hutchinson.belmont.ma.us/tth/>) can generate a HTML files from your LaTeX files.

7.2 On Linux/ Unix Systems

There are multiple distributions of TeX and LaTeX. The distribution known as the `tetex` is recommended.

Use either `kdvi` or `xdvi` for previewing.

`lyx` is a frontend to LaTeX giving a WYSIWIG view – not recommended.

`texmacs` (<http://www.texmacs.org/>) is excellent. It also is a frontend to LaTeX giving a WYSIWIG view and combines `emacs` with `TEX`

`xfig` for vector-based drawing.

7.3 On Windows NT and above

MiKTeX is excellent, and on par with Unix/Linux versions of L^AT_EX.

8

American English

I strongly urge you to read and apply the book [Strunk and White 1970] The following are collected from the theses drafts of my students.

8.1 Spelling

We use American spelling. Watch for: -ise in verbs (should be -ize).

8.2 Misused Words and phrases

A comma (almost) always follows the e.g., whose Latin expansion means “for example”. The other common abbreviation that we often use is “i.e.” whose Latin expansion means “that is.” Do not use i.e. where e.g. is appropriate.

The following words are almost always unnecessary: “mainly”, “basically”, “very”. **Grep** for each occurrence and remove.

The word “like” should be replaced with “such as”.

Softwares: software is itself plural.

Data is both singular and plural.

“very less” should be either “very little” or “very few.” (Should “very” be deleted?) Less is used for “amorphous” quantities (i.e., real numbers), few is used for discrete items (i.e., integers).

Here are some common phrases and their meanings.

“from the top of head” == unthinking.

Don’t say “implemented” when you mean “used” or “installed”.

latter v. later. Latter == the second one. Later == (time wise, or ...) afterwards.

References

KNUTH, D. 1984. Literate Programming. *The Computer Journal* 27, 2, 97–111.

KNUTH, D. E. 1970. *TEX*.

LAMPORT, L. 1970. *L^AT_EX*.

STRUNK AND WHITE. 1970. *Elements of Style*.