

Generating CAIA Technical Reports and Slide Presentations Using L^AT_EX

Jason But

Centre for Advanced Internet Architectures, Technical Report 070312A
Swinburne University of Technology
Melbourne, Australia
jbut@swin.edu.au

Abstract—CAIA has traditionally provided OpenOffice templates to generate Technical Report formatted papers. This approach has limitations, modifying an existing Technical Report for submission to a conference of journal can be time consuming. L^AT_EX on the other hand provides consistent generation of formatted documents direct to PDF output and simple modification to new formats. In this paper we present the installation instructions for the CAIA Technical Report L^AT_EX Class files on a FreeBSD based system and pointers for generating CAIA Technical Reports.

Index Terms—CAIA, Technical Reports, Templates, L^AT_EX

I. INTRODUCTION

L^AT_EX is a platform independent document formatting and generation tool. In this document I will describe how to install and use the CAIA Technical Report L^AT_EX Class Template.

II. INSTALLING L^AT_EX

L^AT_EX is available for many different operating systems including FreeBSD, Linux, Windows and MacOSX. This section will briefly describe the installation of L^AT_EX under FreeBSD. For other platforms please refer to the links provided in the references.

Initial versions were known as T_EX - a tool for generating and typesetting technical documentation - and provided a macro substitution language to generate consistent output. T_EX was too complex for general purpose use and was subsequently reworked into L^AT_EX with the addition of a set of pre-existing macros and stylesheets.

Many versions of L^AT_EX exist, currently TeT_EX [1] seems to be the most popular choice amongst users. TeT_EX provides not only the traditional command line L^AT_EX compiler, but also tools to generate raw PostScript and PDF output. Given that the traditional processing

steps involved using L^AT_EX to generate DVI output, followed by DVI-PostScript and PostScript-PDF conversion, the option of generating PDF output directly using PDFL^AT_EX is particularly appealing and one we will be concerning ourselves with for the remainder of this report. Amongst other advantages, PDFL^AT_EX allows us to generate PDF files which make full use of the features of the Adobe Acrobat Reader including:

- PDF Document Properties
- PDF Bookmarks
- Hyperlinked tags within the PDF document

A. Installing TeT_EX

TeT_EX can be installed via the `print/TeX` port, this can be done quickly by issuing the command (as **root**)

```
> pkg_add -r TeTeX
```

The `pkg_add` command will fetch and download the precompiled package to your system, uncompress the files and finally install them to the required location. The TeT_EX package is large, and if your `/var` partition does not have enough space for the uncompressed package, the installation may fail. In this case you may wish to temporarily rename `/var/tmp` and create a softlink that points `/var/tmp` to an empty directory on a larger partition during the TeT_EX package install (of course remembering to reset your system after the install).

B. Have You Already Got It?

The TeT_EX package has been pre-installed on the CAIA FreeBSD pre-compiled image. If your workstation has been built from this image, then you can safely ignore the preceding section.

III. INSTALLING THE TEMPLATE CLASSES

The CAIA Technical Report Template is based on the pre-existing IEEE L^AT_EX class file [2]. As such you will need to obtain and install the IEEE template files. At the time of writing, the latest version of the IEEE templates is 1.6c, please check the IEEE website for any updates. This version of the IEEE L^AT_EX template is provided for download on the CAIA ftp server (internal CAIA access only), the provided package also includes the CAIA L^AT_EX class files.

L^AT_EX Class files are typically installed in `<latex.home>/textmf/text` where `<latex.home>` is usually `/usr/local/share` for FreeBSD but maybe elsewhere on other platforms. When new class files are installed, the L^AT_EX compiler needs to have its database updated so that it can find the new template files. It is necessary to execute (as **root**)

```
> texhash
```

A. The IEEE Template Files

The IEEE Template consists of the following files:

- **IEEEtran.cls** - The IEEE L^AT_EX Class File
- **IEEEabrv.bib**, **IEEEfull.bib** - Defines macros for shortened IEEE publication names
- **IEEEtran.bst**, **IEEEtranS.bst** - The IEEE L^AT_EX Bibliography formatting files

The files need to be installed as follows:

- ***.cls** - `<latex.home>/tex/latex/IEEE/.`
- ***.bib** - `<latex.home>/bibtex/bib/IEEE/.`
- ***.bst** - `<latex.home>/bibtex/bst/IEEE/.`

If any of the above directories do not exist then they need to first be created.

B. The CAIA Template Files

The CAIA Template consists of the single file `caia-tr.cls` which must be installed in the `<latex.home>/tex/latex/CAIA/.` directory.

C. Do You Need To Do Any Of This?

Again, version 1.6c of the IEEE L^AT_EX templates, along with version 1.00 of the CAIA L^AT_EX templates have been pre-installed on the CAIA FreeBSD 5.4 pre-compiled image. Further, `texhash` has also been executed. If your workstation has been built from this image, then you can also safely ignore the preceding sections.

IV. THE L^AT_EX PRE-AMBLE

A L^AT_EX document is built of a pre-amble, where the document style and format is chosen, properties (such as title and authors) are set, and a group of packages providing different levels of functionality are included. Following the pre-amble is the actual content of the document itself. Typically, you can completely reformat a document by changing the pre-amble and making few - if any - changes to the content of the document itself.

One possible approach is to produce a document file that has the pre-amble and then *includes* the content from a second source file. This allows a quick switch to a different pre-amble file for reformatting without actually changing the source code file itself.

V. THE CAIA TR PRE-AMBLE

When writing the pre-amble for a CAIA Technical Report there are five commands(macros) that need to be issued prior to beginning the document content. You may also wish to load other packages to provide support such as simple citations, inclusion of figures or anything else.

A. Loading the CAIA TR Template Class

The first line of your pre-amble must indicate which class of document you wish to produce, we wish to generate a CAIA Technical Report so we use:

```
\documentclass{caia-tr}
```

B. Setting the CAIA TR Title

The title of the Technical report is issued with the following command. If a title is not set, then the Template Class will issue a warning during compilation and instead use the default title of "NO TITLE SET"

```
\title{Insert your paper title here}
```

C. Setting the CAIA TR Authors

The authors of the Technical Report are set using the command `\CAIATRAddAuthor`. This macro should be used once for every author of the paper (if you have three authors, you need to call this macro three times). The macro takes three parameters, the first is the name of the author, the second is the authors email address, while the third is a footnote to generate for the author. If no footnote is required the third parameter must specified as an empty parameter (`{}`). If no authors are set, then the Template Class will issue a warning during compilation and use a default author of "NO AUTHOR DETAILS" along with an author email

address of **"no@author.defined"**. Below is presented an example to generate two authors, with only the second author having a footnote.

```
\CAIATRAAddAuthor{Donald Duck}
  {donald@disney.com}{}
\CAIATRAAddAuthor{Mickey Mouse}
  {mickey@disney.com}
  {Mickey has big ears}
```

D. Setting the CAIA TR Report Number

The Technical Report number is configured using the command `\CAIATRSetDetails`. This macro takes four parameters, in order being the year of the paper in the format YY, the month in the format MM, the month in long word format, and the day and index number in the format DDA. If not called, then the Template Class will issue a warning during compilation and use a default TR number of **"YYMMDDA"** and a Report Month in the footer of **"Month 20YY"**. To generate this TR the command sequence below was used.

```
\CAIATRSetDetails{05}{11}{November}
  {01A}
```

E. Setting the CAIA TR Confidential Footer

Some CAIA Technical Reports are confidential - these reports are distinguished by an extra line in the footer of the CAIA Technical Report. The command required to generate this footer is `\CAIACConfidential` and takes one of two formats.

```
\CAIACConfidential{}
```

This form of the macro will install the default Confidential footer at the bottom of the TR.

```
\CAIACConfidential
  {String for the footer}
```

This form of the macro will install the provided string as the Confidential footer. To generate the © symbol you should use the `\copyright` command.

VI. THE CAIA TR CONTENT

A \LaTeX document is encapsulated by the \LaTeX standard `\begin{document}` and `\end{document}` commands. Instructions on how to write \LaTeX documents is beyond the scope of this document, you should consult an online reference. Further, since the CAIA TR Class Template is based on the IEEE template, more information on how to format of your document can be found in the IEEE documentation. This section focusses on CAIA TR specific changes to this information.

A. The Title Block

The first part of the document is the title block, which consists of the Technical Report title, authors, CAIA affiliation details and Technical Report Number and finally the authors email addresses. This block is generated using the command.

```
\CAIATRMaketitle
```

The `\CAIATRMaketitle` macro will:

- Correctly Format the title block area
- Insert footnote marks if an author was added with an accompanying footnote
- Append the footnote text to the bottom of the page
- List the authors email addresses and convert them to hyperlinked tags within the final PDF

Following this command you can begin writing your document. To correctly format a CAIA Technical Report you need an abstract and a set of keywords, these fields are delineated by:

```
\begin{abstract}
...
\end{abstract}
```

```
\begin{keywords}
...
\end{keywords}
```

At the end of your document, you need to provide your references which are formatted using the IEEE abbreviated style. To generate references from a file called **"references.bib"**, issue the commands (at the end of your document):

```
\bibliographystyle{IEEEtran}
\bibliography{IEEEabrv,references}
```

These macros will:

- Put in a **"References"** section header
- Properly format all references for the paper

NOTE: \LaTeX will only include references that have been cited somewhere within the paper. This means that your reference file could contain details of multiple papers but if you have only cited five, then only five will appear in your reference list. This allows you to keep a single reference file for all your papers and saves you from re-generating the information for the same set of papers multiple times.

B. Document Template

To facilitate your task even further, a document template has been generated and is available at the CAIA

Website [3]. This template contains a complete \LaTeX source file to generate an empty Technical Report. You can begin from and edit this document.

VII. \LaTeX GOTCHAS

Some things to be wary of.

A. Image formats

When generating PDF files directly using $\PDF\LaTeX$ all images must be either `*.gif`, `*.jpg`, `*.pdf`. However if generating PostScript output using $\PS\LaTeX$ then image files must be in `*.eps` format.

B. Balancing the Last Page

Balancing the columns on the final page is important for Technical Reports and for papers generated for submission to IEEE Journals or Conferences. Unfortunately, this is one task that \LaTeX does not perform so well at. Indeed, this must be manually accomplished and therefore best left until the Technical Report is finalised so you know exactly where to put the column break.

The IEEE recommendations for balancing columns on the last page are:

1) *newpage*: - The first option is to use `\newpage`. This command is placed within the text where you want the column break to occur, you may need to use a process of trial-and-error to generate the best results.

2) *enlargethispage*: - The second option uses `\enlargethispage{-X.Yin}` placed anywhere within the text of the last page. You specify the amount to shrink the column by. Again this may involve a process of trial-and-error.

3) *triggeratref*: - A different procedure is required if the column break is to be within the references since you don't have access to the text that makes up the references. In this instance you can use the command `\IEEEtriggeratref{ref_no}` to specify which reference number you wish to appear at the top of the next column.

The IEEE template typically uses `\newpage` to achieve this, the command to use can be changed but this should not be necessary unless your references will begin on the second to last page, possibly not even then. If you do not wish to use the default page break command to stop the column, you can change it using the `\IEEEtriggercmd{\enlargethispage{-X.Yin}}`. If this is the case, you would still need to use the `\IEEEtriggeratref{ref_no}` command to actually trigger the column balancing.

C. Balancing pages using the Balance package

An alternative to help balance the packages is to use the \LaTeX `balance` package (if you have it installed). To use this package you need to insert into your document pre-amble:

```
\usepackage{balance}
```

You then need to issue the following command somewhere within the text for the first column on the last page.

```
\balance
```

VIII. BUILDING THE FINAL PDF

One of the more difficult things about \LaTeX to comprehend is the document generation process. Once the source \LaTeX files have been generated, it is necessary to compile them to the final presentation, in our case into a PDF file using $\PDF\LaTeX$. The major issue is the fact that \LaTeX and its variants typically need to be executed several times before the *final* document is generated.

This section covers what steps are required to build a complete document using \LaTeX , it is hoped that the reader will gain an appreciation for why tools such as KILE are so useful in that many of these steps are automated.

A. Compiling a Document With References

When first compiling a \LaTeX source file, \LaTeX will fail because the bibliography file `"*.bbl"` is not present in the directory. In order to generate this file, we need to execute $\Bib\TeX$.

The problem with executing $\Bib\TeX$ is that it will also fail unless the output files generated by the initial run of \LaTeX (`*.out` and `*.aux`) are also present as these files contain information about which file contains the input list of references and which of these references are actually cited.

When executed, $\Bib\TeX$ will generate the expected `"*.bbl"` file which will contain a bibliography list formatted in the style selected in the source \LaTeX document. This list will contain only the cited references and be in a form that can be directly included by \LaTeX the next time it is executed.

Once the bibliography file is generated we can run \LaTeX again which will successfully include the generated bibliography file into the final PDF. However we still do not have a complete document as the citations within your document will not be correctly numbered as \LaTeX is not yet sure of their correct number until after

execution. The citation details will now be stored in the \LaTeX output files ready for another pass.

Finally we need to run \LaTeX one final time to ensure that all the citations can be numbered correctly when generating the final document. The correct compilation chain is:

```
 $\LaTeX$  - Bib $\TeX$  -  $\LaTeX$  -  $\LaTeX$ 
```

The upshot of this process is that when you are editing a document, the following rules apply:

- If you add a new citation to your document that has not been cited in this paper yet, you must re-execute all four steps in the compilation chain in sequence.
- If you modify the details of the references in the input bibliography (*.bib) file, you must re-execute the last three steps in the compilation chain in sequence.
- If you change the contents of the document source, you only need to reexecute \LaTeX once.

B. Ensuring PDF Bookmarks are Complete

When producing the final PDF, each `\section`, `\subsection` and `\subsubsection` will generate a multi-level PDF bookmark in a collapsible tree that can be used with the Adobe Acrobat Reader application. This feature allows for quick navigation through the final document, especially for large documents.

However, as for generating papers with references, this required multiple passes of \LaTeX execution. Executing \LaTeX once will generate the list of bookmarks which will only be inserted into the final PDF when \LaTeX is next executed.

The upshot of this process is that when you are editing a document, the following rules apply:

- If you add change the section layout of your document such that the bookmarks will be different, you must execute \LaTeX twice, once to generate the new bookmark list and once to incorporate it into the final PDF.
- If you modify the document without changing the section layout, you only need to execute \LaTeX once as you can reuse the bookmarks from your previous execution.

C. Simplify the Task

One option is to write a Makefile to automate the task of generating a document, a better approach is to use KILE [4]. KILE is a KDE based \LaTeX editor

that facilitates writing of \LaTeX documents by providing support for:

- \LaTeX sensitive source highlighting
- Automatic \LaTeX macro completion
- Automated \LaTeX build cycles
- \LaTeX project support

KILE can be configured to check \LaTeX output for references and to re-execute \LaTeX as many times as is required to correctly generate the reference list and the citation numbers. KILE does not however re-execute \LaTeX to properly generate PDF bookmarks, this will need to be performed manually.

D. Installing KILE

The current KILE package for FreeBSD can be installed KILE by executing (again as **root**)

```
> pkg_add -r kile
```

KILE can also be installed by building it from source in the ports tree. In this case run (as **root**)

```
> cd /usr/ports/editors/kile
> make
> make install
```

IX. SOME THOUGHTS ON REFORMATTING PAPERS

With \LaTeX changing the underlying format of a paper is a matter of rewriting the pre-amble to take into account the new document class file you wish to use. \LaTeX takes care of the rest of the formatting, including resizing images, and setting fonts.

ACKNOWLEDGEMENTS

Many thanks to Andres Rojas who showed that it was actually possible to use the pre-existing IEEE \LaTeX template with some slight modifications to generate a \LaTeX template that could also be used to generate CAIA Technical Reports.

REFERENCES

- [1] Thomas Esser, "The teTeX Homepage," March 2007, <http://www.tug.org/tetex>.
- [2] IEEE, "IEEE LaTeX Template," March 2007, <http://www.ieee.org/web/publications/authors/transjnl/index.html>.
- [3] CAIA, "Centre for advanced internet architectures," March 2007, <http://caia.swin.edu.au>.
- [4] K. D. Group, "Kile - an integrated latex environment," March 2007, <http://kile.sourceforge.net>.