The University of York Department of Computer Science

Submitted in part fulfilment for the degree of MEng.

An evaluation of indoor localisation techniques in disaster environments

Miguel d. Boland

15th November 2016

Supervisor: Leandro M. Soares

Number of words = 8832, as counted by `wc -w`.
This includes the body of the report only.

Abstract

Abstract

Dedication

Acknowledgements

Acknowledgements

Contents

List of Figures

List of Tables

Part I

Preliminaries

1 Introduction

In each taught course, undergraduate or postgraduate, there is a compulsory large project.[1] By far the largest component of the assessment of the project is a written report. There are various appropriate technologies for producing reports. Among these is Lamport's LaTeX [**?**].

This user guide describes a LaTeX 2ε class, |UoYCSproject|, to help in the type-setting of project reports; it is (inappropriately) written using that document class. The division into parts, chapters and so on is too heavy for a brief introduction and user guide, but appropriate for a project report. The source code for this document is available through the CSW web site; you are welcome to use it as a template.

1.1 What is LaTeX?

LaTeX, or more strictly, LaTeX 2ε, is a notation for describing document structure (much as HTML or XML applications) [**?**]. It is very different from WYSIWYG, which has been characterised as "What you see is all you've got"[2]

LaTeX 2ε is built on top of Donald Knuth's TeX [**?**]. TeX is a notation for describing type-set pages *plus* a macro language. LaTeX 2ε is a collection of TeX macros that allows for extensions and modifications using the class and package mechanisms. Thus a LaTeX 2ε description of a document can be turned into print by processing it with a suitable program.

Output is available as the original Device Independent (DVI) format (by using |latex| to process the document), PostScript (by converting from DVI) or PDF (by using |pdflatex| to process the document).

TeX itself was developed by Donald Knuth for type-setting his books, particularly his multi-part work on algorithms [**?** **?** **?**]; take a look at

[1] Except for the three teaching-year joint degrees with mathematics, where a computer science project is optional.

[2] **?** , p7, Footnote 1] says that "Brian Reid attributed this phrase to himself and/or Brian Kernighan".

them to see what is possible. He also developed a font design program to accompany TEX, .[3]

1.2 Advantages of TEX

TEX has a very sophisticated text type-setting algorithm; its implementation is proved optimal (Donald Knuth did more or less found the theory of algorithms). The PDFTEX engine extends the algorithm to include hanging punctuation, for even better results. (See the TEX showcase for several examples; it lives at http://www.tug.org/texshowcase/.)

TEX has a very sophisticated mathematics type-setting algorithm.

TEX also has a Turing-equivalent macro language so that you can program substructures in your document.

1.3 Advantages of LATEX 2ε

LATEX 2ε provides a pre-defined set of document structures (using the TEX macro language), and hooks for integrating further structures.

LATEX 2ε simplifies the task of writing TEX macros (unless you need something very sophisticated).

1.4 Advantages of a programmable mark-up language

I consider the ability to write definitions the greatest advantage of TEX-like systems.

Such a facility enables its users to design a collection of macros that reflect the abstract syntax of important structures in the document (later we will see an example of part of a collection of macros for describing cryptographic protocols). Just doing this will help you ask the right questions about your project, even if you end up using some other document processing system. The fact that LATEX 2ε also lets you associate type-setting commands with each element of the abstract syntax is an

[3] An illustration of how Donald Knuth's mind works. The current version of TEX is 3.141592; the next version, should there be one, will be numbered 3.1415926, and the one after that 3.14159265. On his death the source code is to be amended to print out 'Version \Pi', and no further changes will be allowed. Similarly, version numbers are converging on *e*; currently it is Version 2.71828.

added bonus, and one that gives you consistent type-setting across the document, and between documents.

2 Useful references

2.1 Books

?] The original source. It has a reasonable reference manual, but can be terse. It does not cover package and class writing, nor does it cover more than a handful of useful packages. It does describe the and index making programs.

?] A comprehensive reference; it covers everything except the many add-on packages. Most people use this as their primary reference.

?] A guide to many of the most useful add-on packages and classes.

?] A slightly dated guide to packages for graphics.

?] A slightly dated guide to packages for adding hyperlinks, and producing PDF and HTML from LaTeX 2ε.

2.2 Papers

There are many papers describing LaTeX 2ε and its associated packages. They are available on-line, usually through the Comprehensive TeX Archive Network.[1]. They are usually also available on the TeX Live distribution,[2] which the department uses.[3]

The useful *general* papers are:

The Not So Short Introduction to LaTeX 2ε [?]

Available in several languages.

Be warned that this paper describes the standard classes. There are a few differences in the class options and declarations bewteen the standard classes and |UoYCSProject|.

[1], http://www.ctan.org/.

[2]http://www.tug.org/texlive/

[3]Departmental Linux users should look under file:///usr/local/pkg/ for the current TeX Live distribution, and under that for the various |doc| directories; documentation is usually in |pdf| or |dvi| files.

Math mode [?] A detailed explanation of typesetting mathematics in LaTeX 2ε.

The Comprehensive LaTeX Symbol List [?] An enormous list of symbols and how to make them.[4]

Packages in the 'graphics' bundle [?] A bit out of date (it does not describe PDF extensions), but a useful introduction.

Hypertext marks in LaTeX [?] Access to hypertext features via the |hyperref| package.

Most of the effects happen automatically on loading the package.

It works best in combination with the hypcap package.

|UoYCSproject| loads these packages for you, and sets some of the manual things to sensible values.

The KOMA-Script bundle [?]

|UoYCSproject| is based on the KOMA-Script |scrreprt| class. The manual will tell you about several extra facilities available to you (but you should not change layout, and such things).

2.3 Web resources

The Comprehensive TeX Archive Network <http://www.ctan.org/>

What it says on the label. Almost everything you need in the way of TeX and friends can be found here. Also known as .

The TeX Users Group (TUG) <http://www.tug.org/> A useful web site.

TUG members get the TeX Live distribution as part of their subscription.

TeX FAQ <http://faq.tug.org/>

An extremely useful first port of call for solving common problems, hosted by TUG.

TeXdoc Online: TeX and LaTeX Documentation <http://texdoc.net/> A fast way of finding documentation for TeX-related things.

[4] An experimental web application for finding symbols can be found at http://detexify.kirelabs.org/.

The PracTEX Journal <http://tug.org/pracjourn/>

An on-line journal of TEX practice, including a Q&A section.

The LATEX Project <http://www.latex-project.org/>

The centre of the LATEX project.

The TEX newsgroup <news:comp.text.tex> If asked politely, questions not in the FAQ or standard sources of documentation will usually be answered by gurus. *Minimal* examples of problems, together with the versions of TEX, LATEX and all classes and packages used in the example must be given.

Motivation <http://www.slideshare.net/LianTzeLim/latex-more-than-just-academic-papers-and-theses>
An excellent set of slides introducing what LATEX can do for you, by Lian Tze Lim.

The beauty of LATEX <http://nitens.org/taraborelli/latex> A page describing typographic advantages of TEX-based systems over common competitors.

A wiki for LATEX 2ε <http://en.wikibooks.org/wiki/LaTeX> A relatively new resource; as good or as bad as a Wiki can be.

A Visual FAQ for LATEX 2ε <http://www.tex.ac.uk/tex-archive/info/visualFAQ/visualFAQ.pdf>
The associated README file for this resource says:

> Having trouble finding the answer to a LaTeX question? The Visual LaTeX FAQ is an innovative new search interface that presents over a hundred typeset samples of frequently requested document formatting. Simply click on a hyperlinked piece of text and the Visual LaTeX FAQ will send your Web browser to the appropriate page in the UK TeX FAQ.

MathTran instant preview <http://www.mathtran.org/toys/jfine/editor2.html>
A web-based application to let you try out small pieces of TEX (*not* LATEX 2ε) source code (especially mathematical source code) to see what the type-set version looks like.

2.4 Video

Marc van Dongen's Introduction to LATEX <http://youtu.be/s04nZHtl8jg> Fifteen minutes and 50 seconds of advertisement and gentle introduction to the working model and basic features.

3 The LaTeX edit-process cycle

The standard books on LaTeX describe the process by which you turn your document description into ink. Most describe this process using |latex|, which produces DVI format, and a DVI viewer, such as |xdvi|.

Since these books were written it has become more convenient to use |pdflatex|, which produces PDF format, and a PDF viewer such as |xpdf| or |acroread| (|xpdf| is slightly more convenient than |acroread|, although it does not support all the features that |acroread| does, nor does it have as good rendering).

LaTeX 2ε source may be created using any editor. Several editors have support for TeX and LaTeX 2ε, including managing the edit-create cycle. I like |emacs| with the AUCTeX enhancements to the TeX modes; Windows users often use |WinEDT|.

The perfect edit-process cycle goes like this:

1. Create a LaTeX 2ε source file, and any others needed, such as a file, figures, and so on.

2. Run |pdflatex|. (This creates PDF output with place-holders for missing information and auxiliary files with information about the table of contents, cross references, name of file(s) containing the bibliographic database, and so on.)

3. Run . (This creates a file containing the references.)

4. Run |pdflatex|. (This recreates PDF output with place-holders for missing information and auxiliary files with information about the table of contents, cross references, name of file(s) containing the bibliographic database, and so on, but this time also with bibliographic citations.)

5. Run |pdflatex|. (This will create PDF output which is complete.)

Imperfections in this cycle creep in when you make errors in the files, add new citations, and so on. Further recompilation is necessary; rerunning

is only necessary if new citations are inserted or if an entry in the bibliographic database changes.

Tools such as AUCTeX/ |emacs| and |WinEDT| can manage the process for you.

A brief guide to using LaTeX 2ε on (some of) the department's systems is given in **??**.

Part II

Concepts of LaTeX 2ε

In this part of the document I briefly review some of the main concepts of LaTeX 2ε documents.
This is *not* a comprehensive guide to LaTeX 2ε, but a list of useful concepts, together with a few hints and tips. Consult the main references for full details.

4 The anatomy of a LaTeX 2ε source file

The layout of a normal LaTeX 2ε document description is given in **??**. [caption=The anatomy of a LaTeX 2ε file,label=lst:anatomy,float,morekeywords=document,maketitle,numbers
[class options]class name preamble (definitions and declarations)

The University of York Department of Computer Science

Submitted in part fulfilment for the degree of MEng.

An evaluation of indoor localisation techniques in disaster environments

Miguel d. Boland

15th November 2016

Supervisor: Leandro M. Soares

Number of words = 8832, as counted by `wc -w`.
This includes the body of the report only.

Abstract

Abstract

Dedication

Acknowledgements

Acknowledgements

Contents