HAPPINESS THROUGH \LaTeX FOR MICROSOFT WORD USERS

M. Behr
Chair for Computational Analysis of Technical Systems
RWTH Aachen University
52056 Aachen, Germany
March 22, 2014

Abstract

This short document is supposed to help in a transition from e.g. Microsoft Word to \LaTeX as a paper production tool. Apart from providing hints on \LaTeX usage, it also serves as an example of a \LaTeX document in general, and a scientific paper in particular. By changing its contents but retaining the general structure, the creation of your first \LaTeX publication should not be so hard. Source files for this document are available at http://www.cats.rwth-aachen.de/library/publishing/tex. The command-line part needs updating...

1. Introduction

This section is typeset from its \TeX source – \texttt{s-intro.tex}. That source file is a plain text file, containing mostly document text. It also contains commands instructing \TeX to do something or other. Such commands appear in the source file as \texttt{\textit{command-name}}. For example \texttt{\texttt{\textit{pagebreak}}} causes \TeX to break the page at the current line, while \texttt{L} produces – a typical Polish character. Whenever \% is encountered, the rest of the line is ignored. This is the way to include comments in the source file.

It doesn’t matter how you arrange the text in the source file – whether you type one word per line or fill the whole line with words, \TeX will take care of breaking them into lines. It will not rest until the lines are perfectly justified and have the perfect spacing between individual words, even if it means hyphenating some of them. Note that automatic hyphenation can make the- rapist out of therapist, so be careful. It is also unimportant how many spaces you put between words, because \TeX always treats them as one. The only white space that matters is an empty line, which means a beginning of a new paragraph.

Braces {...} are used to limit the effect of some command to just a certain group of words. For example the \textbf{boldface} is achieved through \texttt{\textbf{boldface}} and \textit{italic} with \texttt{\textit{italic}}. Similar commands change font sizes, alignment etc. Additional commands are typically defined via \textit{style files}. For example this document uses the \texttt{fine} style to redefine certain standard \LaTeX commands, and \texttt{graphicx} to add commands for including PostScript graphics.

\footnotesize
\textsuperscript{†}This research would be sponsored by NASA-Johnson Space Center under grant NAG 9-449 and by NSF under grant MSM-8796352 (if they were smart!).

1
So why \TeX?

- For me its equation typesetting syntax is actually easier to remember and apply than MS Word. And the final result looks much more acceptable. More about equations later.

- \LaTeX{}, especially when coupled with Bibtex, has a powerful system for resolving equation numbers, references etc. Equation numbers are computed automatically, so they can change during editing. But each equation can be assigned an optional label, say e-nav-stok-mom, and later referred to with \texttt{(\ref{e-nav-stok-mom})}. When \LaTeX{} is run, it will substitute the real equation number inside the parenthesis.

- For references, you create a bibliographic database containing all of the references you ever care to refer to, and put it in, e.g., \texttt{main.bib} file. Then you tell \LaTeX{} to search it by including \texttt{\bibliography{main}} at the end of the document. After that you refer to individual papers with \texttt{\cite{Behr92a}}, where Behr92a is a label used in \texttt{main.bib} to describe one of my excellent papers. \LaTeX{} will pull all the references whose labels appear in the text from the \texttt{main.bib} file, number them, arrange them in a neat list at the end of the paper, and replace all \texttt{\cite{...}} commands with the proper reference number in a bracket.

- Including properly aligned and captioned Postscript figures, like the ones produced with \texttt{http://www.cats.rwth-aachen.de/software/visualization/pager} or \texttt{gnuplot}, is quite easy.

- Easy macro definition saves typing. For example if you are sick and tired of typing “Hughes and Brooks”, you can define a macro \texttt{\newcommand{\hb}{Hughes and Brooks}}, and the names magically appear every time you type \texttt{\hb}.

And why not?

- You don’t see immediately how the equation will look like. It might not even be correct. Nevertheless you can be certain that it will look better than if it were typeset with MS Word.

- Running \LaTeX{} is like compiling a program, and error messages are completely incomprehensible.

- There is no easy way to move back and forth between MS Word and \LaTeX{} right now.

- Keynote on Mac is a good alternative for slides, and can be used with \LaTeX{} equations using the \LaTeX{}It utility.
2. Operation

The \LaTeX\ system consists of several parts.

1. \TeX\ program by Donald Knuth, which does the actual work. It is invoked with tex command.

2. \LaTeX\ program by Leslie Lamport, together with a bunch of macro and format files, forms a shell around plain \TeX\. It is the preferred program for scientific reports. \LaTeX\ macros make \TeX\ much easier to use. \LaTeX\ is invoked with latex command.

3. Bibtex bibliography management system.

4. Graphicx system for including Postscript graphics in \LaTeX\ documents.

5. Various postprocessors and previewers.

All this software is public domain and is installed on CATS Macs There are also GUI front ends like TeXShop, which usually come preconfigured, and no setup is really necessary. The following refers to the more difficult task of getting \LaTeX\ to run in a Unix environment.

Things to do before using \LaTeX\ for the first time:

1. Add following lines:

   ```
   setenv TEXINPUTS .:$HOME/lib/tex/inputs:/usr/local/lib/tex/inputs
   setenv BIBINPUTS .:/bib:$HOME/lib/bib:~behr/lib/bib
   ```

   to your .cshrc file. Variable TEXINPUTS tells \LaTeX\ where to look for document style and macro definition files. Variable BIBINPUTS tells Bibtex where to look for bibliography files. For now you don’t have to create $HOME/lib/tex/inputs or $HOME/lib/bib, but in the future you will want to have some common place for your often used macro or bibliography files. You must either do source .cshrc or login anew for this change to take effect.

2. Copy the antique go script from ~behr/bin/scripts to your own binary directory. If you want to use it in the same login session, do rehash.

Things to do when creating brand new \LaTeX\ document:

1. Create a new directory to hold the new document sources. It is a good practice (essential if you want to use the amazing go script) to have a separate directory for each document you are working on.

2. In the new directory, create all the sources using vi or some such. A useful vi feature is the margin wrap which automatically inserts carriage returns as your type approaches right margin - set it by having a line set wm=5 in a .exrc file in your home directory. Let’s assume from now on that the main source file of your document is named blah.tex. Note that all source files should have .tex suffix.
3. Make sure that all the .sty and .tex files used by blah.tex, if any, are in one of the directories specified in the TEXINPUTS variable. For example the document you are reading, main.tex (which may be found in behr/tex/intro), uses files s-*.tex and fine.sty. Similarly, place all Bibtex files specified in the \bibliography command in one of the directories specified in the BIBINPUTS variable. If this is your first try, just put all the stuff in the current directory.

4. Initialize the fantastic go script. Do this by typing go -i blah.tex. This creates a file .gorc in the current directory, full of magic information. You can think of go as a make command specially designed to handle \LaTeX documents, and of .gorc as a Makefile.

Things to do after modifying \LaTeX document:

1. Once the go is initialized, then all you have to do after making changes to your document is to type go. The script will check whether your document source file, or any of the files it depends upon, has changed, and run appropriate programs. To resolve all references it may be necessary for go to run \LaTeX several times. If you use a bibliography in your document, go will also run Bibtex. The final outcome should be a typeset ”Device Independent” binary file called blah.dvi.

2. To make sure that the DVI file is correct, and save some printer paper, you can use a previewer to see what it looks like. On UNIX machines there is one previewer available to do this job – xdvi. It requires an X windows display. Running xdvi blah.dvi will show you a window with the first page of your document. To navigate through pages or quit use on-screen buttons. See man xdvi for more information.

3. To print the document on a Postscript printer use a dvips converter. A command:

   dvips blah.dvi

   will print the whole file, while

   dvips -p 2 -l 4 -o blah.ps blah.dvi

   will convert only pages 2, 3 and 4 and leave them in blah.ps. See man dvips for more information.

4. Convert to PDF with pstopdf blah.ps.

5. Gawk at the neat printout, or edit the source files again.
3. Mathematical expressions

We consider a viscous, incompressible fluid occupying at an instant \( t \in (0, T) \) a bounded region \( \Omega_t \subset \mathbb{R}^{n_{sd}} \), with boundary \( \Gamma_t \), where \( n_{sd} \) is the number of space dimensions. The velocity and pressure, \( \mathbf{u}(\mathbf{x}, t) \) and \( p(\mathbf{x}, t) \), are governed by the Navier-Stokes equations:

\[
\rho \left( \frac{\partial \mathbf{u}}{\partial t} + \mathbf{u} \cdot \nabla \mathbf{u} \right) - \nabla \cdot \mathbf{\sigma} = 0 \quad \text{on } \Omega_t, \quad \forall t \in (0, T) \quad (1)
\]

\[
\nabla \cdot \mathbf{u} = 0 \quad \text{on } \Omega_t, \quad \forall t \in (0, T) \quad (2)
\]

where \( \rho \) is the fluid density, and \( \mathbf{\sigma} \) is the stress tensor. For a fluid with viscosity \( \mu \), this tensor can be decomposed into the isotropic and deviatoric parts:

\[
\mathbf{\sigma} = -p \mathbf{I} + \mathbf{T}, \quad \mathbf{T} = \mu \left( \nabla \mathbf{u} + (\nabla \mathbf{u})^T \right). \quad (3)
\]

Here we stop considering the equations – they were just a sample of mathematical formulas typeset in \TeX. By looking at the source of this document you will understand how they were made. It would take ages to completely describe the math typesetting in \TeX. Interested reader is referred to [1], which is a very good manual, and is available in the TAFSM Lab. In the remainder I will just give some very introductory stuff.

All mathematical expressions appearing inline with the text are enclosed in dollar signs \( \$ \ldots \$ \). For example \( a + b = c \) was created using \( \$a+b=c\$ \). Subscripts and superscripts are made using special characters \( _{-} \) and \( ^{-} \), e.g. \( u^i_{\text{df}} \) was created with \( \$u_{i_{\text{df}}}^*\$ \). All greek letters and symbols are defined in \TeX as macros with clearly designed names, for example \( \sigma \) is typed in as \( \$\sigma\$ \). Equations which appear in their own paragraph space, usually numbered, are called ”displayed math”. They are obtained by enclosing mathematical expressions in \begin{equation} ... \end{equation} or \begin{eqnarray} ... \end{eqnarray} See chapter 3 of [1] for more information. You can also learn a lot by looking at how this section was typeset, or by looking at source files of some real paper typeset in \LaTeX.

And for dessert, here is the space-time formulation in \LaTeX: given \((\mathbf{u}^h)^-_n\), find \( \mathbf{u}^h \in (\mathcal{S}^h_u)^n \) and \( p^h \in (\mathcal{S}^h_p)^n \) such that

\[
\int_{Q^n} \mathbf{w}^h \cdot \rho \left( \frac{\partial \mathbf{u}^h}{\partial t} + \mathbf{u}^h \cdot \nabla \mathbf{u}^h \right) dQ + \int_{Q^n} \nabla \mathbf{w}^h : \mathbf{\sigma}(p^h, \mathbf{u}^h) dQ \\
+ \sum_{n=1}^{(na)_n} \int_{Q^n} \tau \left[ \rho \left( \frac{\partial \mathbf{u}^h}{\partial t} + \mathbf{u}^h \cdot \nabla \mathbf{u}^h \right) - \nabla \cdot \mathbf{\sigma}(q^n, \mathbf{w}^h) \right] \\
\quad \cdot \left[ \rho \left( \frac{\partial \mathbf{u}^h}{\partial t} + \mathbf{u}^h \cdot \nabla \mathbf{u}^h \right) - \nabla \cdot \mathbf{\sigma}(p^n, \mathbf{u}^h) \right] dQ \\
+ \int_{Q^n} q^n \rho \nabla \cdot \mathbf{u}^h dQ + \int_{Q^n} \delta \nabla \cdot \mathbf{w}^h \rho \nabla \cdot \mathbf{u}^h dQ + \int_{\Omega^n} (\mathbf{w}^h)^+ : \rho ((\mathbf{u}^h)^+_n - (\mathbf{u}^h)^-_n) dQ \\
= \int_{(P^n)_h} \mathbf{w}^h \cdot \mathbf{h}^h dP, \quad \forall \mathbf{w}^h \in (\mathcal{V}^h_u)^n, \quad \forall q^n \in (\mathcal{V}^h_p)^n, \quad (4)
\]
4. Bibliographies

The bibliography file has its own difficult syntax. The bib file for this document (main.bib) contains examples of common reference types. Wow, isn’t this a short section, or what?

5. PostScript Figures

It is a straightforward task to include PostScript figures in a \LaTeX document using the \texttt{graphicx} macro package. The basic usage is: \texttt{\includegraphics[width=0.1in]{h.ps}} which produces \texttt{h.ps}.

When using \texttt{graphicx} and \texttt{dvips}, the PostScript graphics can be stored in the compressed form, and automatically decompressed when needed by \texttt{dvips}. A script which compresses an image and extracts the bounding box information is in:
\texttt{~behr/bin/scripts/pscompress}
Starting with file \texttt{blah.ps}, the command \texttt{pscompress blah.ps} will produce \texttt{blah.ps.Z} and \texttt{blah.ps.bb} files needed by \texttt{dvips}.

Recent versions of \LaTeX are more at home including PDF, PNG and other figures, using the \texttt{pdftex} driver rather than the sequence \texttt{latex} and \texttt{dvips}. The trouble with that is that all figures in /\texttt{usr/local/ps} are in the PostScript format, and the newer driver cannot handle them without extra steps.

6. Final remarks

Initial investment in learning \LaTeX is not small, but the result is much more professional looking reports and papers. The \LaTeX user guide \cite{1} is an extremely well written and well organized book. I got most of my information from there. We have one copy in Marek’s library. There are now two editions, the original one, and one revised for the \LaTeXe version\footnote{In a flight of mathematical fancy, that probably seemed amusing at the time, but now makes life difficult for millions, Lamport declared that \LaTeX should not be developed beyond version number 2. It should reach instead, in the limit, version number \( e = 2.71828... \)}. The original one is now considered obsolete. For a guide to the underlying \TeX{} system, see the classic \TeX{}book \cite{2}. Of course I am always available for consultations as a village \TeX{}pert.

References

\begin{enumerate}
\item D. Knuth, \textit{The \TeX{}book}. Addison-Wesley Publishing Company, Reading, Massachusetts, 1984.
\end{enumerate}