Typesetting with TEX

Scientific typesetting made easy

TEX (rhymes with "blecchhh!") is a technical typesetting system created by Donald Knuth of Stanford University. It is currently used by most physicists, mathematicians, and computer scientists, and many astronomers.

TEX is capable of correctly typesetting complicated mathematical expresions, with proper alignment of all elements:

$$\nabla \cdot \vec{D} = \frac{\rho}{\epsilon} \tag{1a}$$

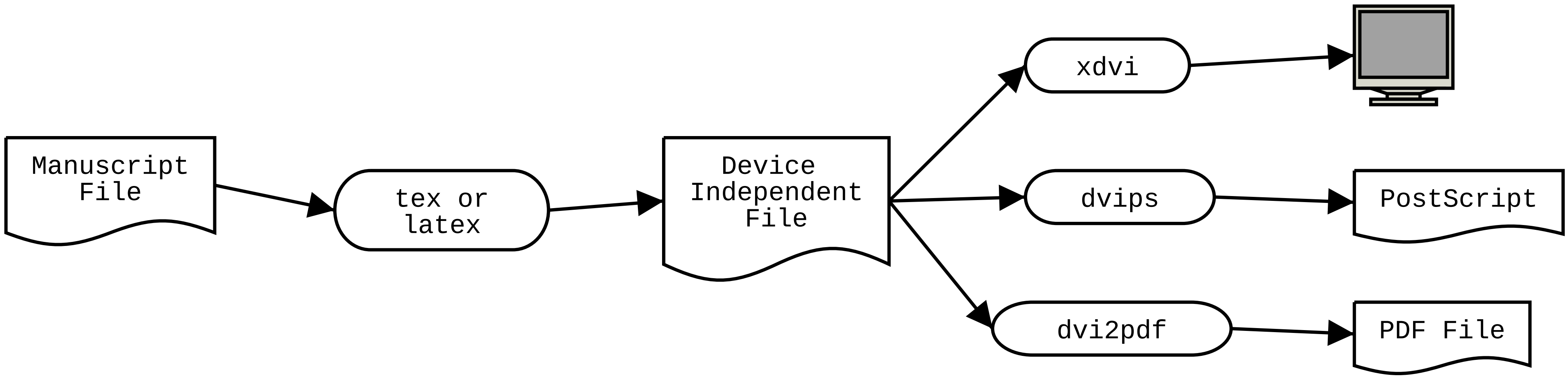
$$\nabla \cdot \vec{B} = 0 \tag{1b}$$

$$\nabla \times \vec{E} = -\frac{1}{c}\frac{\partial \vec{B}}{\partial t} \tag{1c}$$

$$\nabla \times \vec{H} = \frac{\partial \vec{D}}{\partial t} + \vec{J} \tag{1d}$$

This is no mere "processing" of words.

The TEX Process



1. Text, equations, formatting instructions, etc.. are all entered into a "manuscript" file, `myfile.tex`.

2. This is run through `tex` or `latex` (the TEX "engine" and macros) to produce a Device Independent file, `myfile.dvi`

3. The DVI file is processed through an output filter. You can view it on the screen (with `xdvi`), convert it to a PostScript file `myfile.ps` (with `dvips`) or convert it to a PDF file `myfile.pdf` (with `dvi2pdf` or `dvips`→`ps2pdf`)

Typesetting: The Finer Points

Typesetting is more complex than word processing, but for regular text (no equations) typesetting is easy if you keep a few subtle points in mind:

- A new paragraph is signaled in TEX by a *blank line.* The entire paragraph is typeset at one time, with optimal line breaks chosen to make the entire paragraph look good.

- *Quotation marks* are more complicated: You should use left and right single quotes twice to get left and right quotation marks.

- Use a single dash for a short hyphen, "-", a double dash for a regular hyphen, "–", and a triple dash for a long hyphen "—".

- *Diacritical marks* are created with special control characters. For example, typeset the words "façade" and "coördinates" with `fa\c cade` and `co\"ordinates`.

Examples are shown in the file `story.tex`.

Symbols, Subscripts, Superscripts

Mathematics text is typeset differently. Variables are typeset in italics, with different spacing. Mathematics in the body of a text must therefore be enclosed in "math quotes", which are dollar signs, `$`.

Math symbols are indicated by a "control word" name, which begins with a backslash, (eg. `$\alpha$` and `$\Omega$` produce α and Ω).

Subscripts are indicated with an underscore `_`, while superscripts are indicated with a circumflex `^`. Grouping is indicated with curly brackets `{` and `}`.

- To get "$Y_l^m(\theta_1,\phi^\prime)$" you type:

```
$Y_l^m(\theta_1,\phi^\prime)$
```

- To get "$G_{\mu\nu} = g_{\mu\nu} R + {1 \over 4} R_{\mu\nu}$" type:

```
$G_{\mu\nu} = g_{\mu\nu} R + {1 \over 4} R_{\mu\nu}$
```

Displayed Equations

Long equations, or important equations, are set off from the text as "displayed" equations. To get a displayed equation you double the math quotes. To get this:

$$\langle \psi_1\vert \psi_2 \rangle = \int_{-\infty}^\infty { Y_l^m(\theta_1,\phi_1) Y_l^m(\theta_2,\phi_2) \over \sqrt{2\pi} } \, d\Omega$$

you would type this:

```
$$
\langle \psi_1\vert \psi_2 \rangle =
           \int_{-\infty}^\infty {
            Y_l^m(\theta_1,\phi_1) Y_l^m(\theta_2,\phi_2)
           \over \sqrt{2\pi} } \, d\Omega
$$
```

Maxwell's Equations

$$
\begin{aligned}
\nabla \cdot \vec D &= {\rho \over \epsilon} \\
\nabla \cdot \vec B &= 0 \\
\nabla \times \vec E &= -{1 \over c} {\partial \vec B \over \partial t} \\
\nabla \times \vec H &= {\partial \vec D \over \partial t} + \vec J
\end{aligned}
$$

Maxwell's equations, nicely aligned, are produced by typing:

```
$$
\eqalign{
   \nabla \cdot \vec D     &=  {\rho \over \epsilon}       \cr
   \nabla \cdot \vec B     &=  0                     \cr
   \nabla \times \vec E    &=
             - {1 \over c} {\partial \vec B \over \partial t} \cr
   \nabla \times \vec H    &= \phantom{-}
    {\partial \vec D \over \partial t} + \vec J \cr}
$$
```

Macro Definitions

TEX is extendable by defining new control words as "macros".
For example:

```
\def\Sph#1{Y_l^m(\theta_{#1},\phi_{#1})}
```

The argument `#1` is replaced with whatever argument you give to the macro. So you can produce

$$\langle\psi_1|\psi_2\rangle = \int_{-\infty}^\infty Y_l^m(\theta_1,\phi_1) Y_l^m(\theta_2,\phi_2) \, d\Omega$$

by typing:

```
$$
\langle \psi_1\vert \psi_2 \rangle =
            \int_{-\infty}^\infty \Sph{1} \Sph{2} \, d\Omega
$$
```

Large collections of pre-defined macros are called "formats".
REVTEX and TEXsis are examples of special formats for physicists.

EPS Figures

Drawings and figures can be included in the document if they are in an "Encapsulated" PostScript file (ie, an `.eps` file).

- In Plain TEX you need to use the macro file `epsf.tex`, like so:

```
\input epsf.tex
   ⋮
\line{\epsfxsize=\hsize\epsfbox{TeX-Process.eps}}
```

- In LaTEX you use the "`graphix`" style package, like so:

```
\usepackage{graphicx}
   ⋮
\includegraphics[width=\columnwidth]{LAT96Fig2}
```

(There is a bit more to it if you also want captions, figure numbers, etc..., but not much. See the file `modlab.tex` for examples.)

Citations and References

A utility program called BIBTEX makes it easy to manage citations and references:

1. You collect one or more bibliography files (eg. `mylist.bib`) containing a list of fields (title, author, journal, etc..) for each work that you might wish to cite. Each item is identified by a unique 'key'. For example:

```
@book{Bevington1969,
          title = {Data Reduction and Error Analysis
                    for the Physical Sciences},
          author = {Philip R. Bevington},
          edition = {First},
          publisher = {McGraw Hill},
          year = {1969}
}
```

2. In your text, when you wish to cite a work, you simply say `\cite{key}`. Each new work gets a new citation number.

3. At the end of the manuscript file you put

   ```
   \bibliography{mylist}
   ```

   This names the bibliography file (ie, `mylist.bib`) and it is also where the list of references will appear.

4. When you run TEX, the citations are added to an auxiliary file, `myfile.aux`. You then run the BIBTEX program, which collects the citations, selects the references from the bibliography file, and outputs the list of references as `myfile.bbl`.

5. When you run TEX again the list of references (in `myfile.bbl`) is inserted at the end of your document (where you put the `\bibliography`).

You only need to run BIBTEX again if you add, delete, or re-arrange references.

TEX Distributions

- On Windows PC's you want to get MikTEX, which collects together both the TEX engine and all available macros, a DVI displayer, and `dvips`. Go to `www.miktex.org` to get it

  A good front-end is WinEDT.

- On MacOSX get TEXShop for the front end, and `II2.dmg` for the TEX engine and macros. [Better instructions on the web sometime soon.]

- On Linux TEX has been packaged by Thomas Escher as teTEX. This is the TEX engine and macros, output filters (`xdvi`, `dvips`, etc.) and other tools.

  Use your favorite editor (`emacs`, `vi`, `nedit`) as the "front-end". These all come with Red Hat Linux.

  There is also a front-end system called Lyx.

TeX for Physicists

- For LaTeX there is a "class" of macros called REVTeX which is used by the American Institute of Physics to typeset their journals (eg. *Physical Review*, *Physical Review Letters*, and *Reviews of Modern Physics*). Authors are encouraged to submit computer manuscripts using REVTeX.

- For Plain TeX the equivalent is TeXsis (see `www.texsis.org`)

- For the Vassar Journal of Modern Physics there is a style file called `vjmp.sty`.

Data reduction is the transformation of numerical or alphabetical digital information derived empirically or experimentally into a corrected, ordered, and simplified form. The purpose of data reduction can be two-fold: reduce the number of data records by eliminating invalid data or produce summary data and statistics at different aggregation levels for various applications. The presentation is developed from a practical point of view, including enough derivation to justify the results, but emphasizing methods of handling data more than theory. The text provides a variety of numerical and graphical techniques. Computer programs that support these techniques will be available on an accompanying website in both Fortran and C++. 5 Data visualisation and reduction 5.1 Producing a good graph 5.1.1 The independent and dependent variables 5.1.2 Linearising the data 5.1.3 Appropriate scales for the axes 5.1.4 Labelling the axes 5.1.5 Adding data points and error bars to graphs 5.1.6 Adding a t or trend line 5.1.7 Adding. a title or caption.Â The aim of error analysis is to quantify and record the errors associated with the inevitable spread in a set of measurements, and to identify how we may improve the experiment. In the physical sciences experiments are often performed in order to determine the value of a quantity. However, there will always be an error associated with that value due to experimental uncertainties.