7. LaTeX Tools

Tex is a powerful macro-based text formatter written by Donald Knuth, popular in the scientific community. Knuth began work on TeX because of problems he experienced with traditional typesetting in volumes I—III of his monumental Art of Computer Programming. with features designed for the production of technical and scientific documentation. LaTeX is the *de facto* standard for the communication and publication of scientific documents.

Well-known extensions to TeX include:

- BibTeX for bibliographies (distributed with LaTeX),

- LaTeX: Lamport TeX - incorporates document styles, A variant of the TeX text formatting program using a set of macro commands useful for medium-to-large technical or scientific documents, but it can be used for almost any form of publishing.

- PiCTeX for pictures, and

- SliTeX for slides (distributed with LaTeX).

LaTeX contains features for:

- Typesetting journal articles, technical reports, books, and slide presentations.

- Control over large documents containing sectioning, cross-references, tables and figures.

- Typesetting of complex mathematical formulae.

- Advanced typesetting of mathematics with AMS-LaTeX.

- Automatic generation of bibliographies and indexes.

- Multi-lingual typesetting.

- Inclusion of artwork, and process or spot colour.

- Using PostScript or Metafont fonts.

LaTeX is based on Donald E. Knuth's *TeX* typesetting language. LaTeX was first developed in 1985 by Leslie Lamport, and is now being maintained and developed by the LaTeX3 Project. The best source for news on TeX and LaTeX is the TeX Users Group.

LaTeX: Fonts

In LaTeX, there are generally three styles within any font family, which are distinguished as font family, font shape, and font series. The commands for family, shape, and series are commutative, so they can be combined as with the command:

{\bfseries\itshape\sfffamily text you want in bold, italics, and sans-serif}

The above command would make the type in bold, italic, and sans-serif. The commands for font can only be combined as long as they aren't contradictory. Contradictory font commands would be trying to get slanted italics, or attempting to combine different font families.

LaTeX expects three font families as defaults.

Font Family	Code	Command
Roman (serif, with tails on the uprights) as the default	rm\	textrm{text}
Sans-serif, with no tails on the uprights	sf	\textsf{text}
Monospace (fixed-width or typewriter)	tt	\texttt{text}

For more on the fonts available with a typical LaTeX installation, please see the documentation in the IMAGE Lab (page 75 in *A Beginner's Guide to Typsetting with LaTeX*). Also, please note that there are many more fonts available for download.

Latex in Indian Languages (iLaTeX)

iLaTeX facilitates the use of LaTeX for preparing documents in Indian Languages. It supports the full featured LaTeX with additional support for the Indian Scripts using the following LaTeX commands.

\devanagari, \bengali, \tamil, \telugu, \gujarati, \kannada, \punjabi, \assamese, \oriya and \malayalam.

The text in corresponding Indian Language is written using the ISCII character set. Under Linux, such texts can be typed using standard text editors such as vi, used in ISCII capable terminal softwares such as iterm. (see the Installation section below).

The text along with other English text and LaTeX commands is processed by LaTeX after converting it to glyph coded strings.

An example of the multilingual text file structure is the following.

\documentclass{article}
\usepackage{indian}
\begin{document}
{\devanagari <<ISCII coded Text in Devanagri>>}
Further text in regular English.
{\bengali <<ISCII coded text in Bengali>>}
\end{document}

CDAC supplied fonts are used for the text printed using iLaTeX.

The following packages should also be installed in order to prepare text in Indian Lanugages.

- **libiscii**

  provides a library package to support conversion from ISCII codes to glyph codes, ISCII codes to Unicode, Unicoe to ISCII, etc. Various fonts are supported by this package)

- **Xiscii**

  Implements the support of Indian Scripts in X windows. Needed to run any X window application that uses Indian Scripts. This package also uses libiscii package.

- **XKB**

  provides the X windows keyboard drivers to handle keyboard layouts for Indian Scripts. It supports three layouts. Regular US layout is used to implement the English keyboard. The Roman-phonetic keyboard layout is used for the Indian Scripts where the Indian Scripts can be typed using the phonetic layout (for example type k m l keys to get Kamal). The Inscript layout (as provided in the IS standard) is implemented to type the Indian Lanugage text. The machine powers on in the US layout and by successive press of key, it moves from one layout to another. The current layout is indicated by the and LEDs on the keyboard. Layout OFF OFF US-English ON OFF US-English (capslock pressed) OFF ON

Roman Phonetic Indian layout ON ON INSCRIPT layout

- **iterm**

  provides the terminal software that allows text handling in Indian Lanugages. It uses Xiscii and XKB support.

- **iLaTeX**

  provides the latex document translation mechanism to glyph codes. The converted text can be compiled using standard LaTeX. Uses libiscii support.

Installation and configuration:

Installation Steps

STEP 1:

Install the following rpms in the given order (rpms are available on website www.cse.iitk.ac.in/users/isciig under downloads link)

1. libiscii
2. Xiscii
3. XKB
4. iterm
5. iLaTeX

(ex. $ rpm -ivh libiscii-1.0-1.i686.rpm)

STEP 2:

Edit /etc/X11/XF86Config-4 file. In the XkbLayout entry replace "us" by "in". This enables the Indian Keyboard (all the three modes as described earlier can be used).

STEP 3:

Restart the X windows.

An example of use:

1. Start iterm. Use "iterm" command for that.
2. Create a file by using vi editor. (ex. $ vi test.tex).
3. Enter the following lines.

\documentclass{article}
\usepackage{indian}
\begin{document}
{\devanagari

4. Press "Scroll Lock" key to change keyboard layout in Phonetic.

5. Type some text in Devanagari script in file. It can be typed in phonetic layout.

6. Press "Scroll Lock" key twice to change keyboard layout to US english.

7. Type the following. {\end{document}

8. Save and close the file.

9. Run the command

$ ilatex test.tex -o gtest.tex

This command converts the text into the glyph coded text.

10. Run the command

$ latex gtest.tex

This command creates the dvi file gtest.dvi.

11. Print the dvi file using the dvips.

$ dvips gtest.dvi -o test.ps

12. The dvi file can be viewed using the dvi viewers. Similarly the postscript file can be viewed using ghostview. This file can also be printed.

Basic Latex Usage

Unix (teTex package)

Edit *filename.tex* with your favorite editor, e.g. gedit under gnome, or emacs.
Create *filename.dvi* by the invocation
latex filename
Create postscript *filename.ps* by the invocation
dvips filename -o filename.ps
Create *filename.pdf* by the invocation
pdflatex filename
Print to the default printer with the invocation
dvips filename
Create html with the invovation
latex2html filename

Windows (Winedt/miktex)

Edit *filename.tex* with winedt, from the *start* menu.

Invoke latex by pushing the latex button or using the accessories menu.

Preview with yap from the accesories menu.

Print from the file menu or create pdf from the pdf submenu of the accesories menu.

Winedt has many buttons and menus for inserting elements of latex syntax into your files.

Mac (OzTex)

Edit *filename.tex* with alpha or bbedit lite.

The program alpha has many buttons and menus for inserting elements of latex syntax into your files. Process, print and preview with OzTex.

Why do we need LaTeX?

LaTeX allows user to be the author and the book designer, rather than just the author. LaTeX acts as typesetter who follows your commands as the book designer and author to create professional documents with relative ease. LaTeX allows one to control how document looks and to understand why it looks the way it does. LaTeX, like any markup language, will also help to have a sense of organization, flow, and structure within work. As interdisciplinary programs grow, technical-format requirements for publication increase. LaTeX offers a one step (or close to one step) solution for documents which need to be in multiple formats. LaTeX documents are all plaintext files. This means one can edit them with any editor, and transfer them to any computer system running LaTex and they will format exactly the same. This is unlike Microsoft Word and many other programs which will alter based on the system, version of software and system, and other variables. Nothing is hidden in LaTeX and nothing will go out of date. LaTeX will never have the gooblety-gook problems that converting documents from Word Perfect to Word to StarOffice incur. In LaTeX, users type the text of their documents along with the accompanying markup (also referred to as commands or control sequences, with control sequences being the proper TeX term). The markup identifies parts of the document by name, such as "title" and "section." LaTeX does the formatting by using the user's markup to guide its internal rules and external stylesheets for typesetting.

Information on some packages, and other typesetting systems based on (La)TEX.

Babel The Babel package by Johannes Braams provides multilingual support for LATEX.

ConTEXt ConTEXt (by Hans Hagen) is a system for typesetting documents. Visit the PRAGMA (Advanced Document Engineering) page for more information.

fontinst The fontinst package originally written by Alan Jeffrey simplifies the installation of PostScript or True Type fonts for use with (La)TEX. Click here to go to the fontinst homepage.

hyperref The hyperref package by Sebastian Rahtz extends the functionality of all LATEX cross-referencing commands so that those can be turned into hypertext links. You can download the package from your favorite CTAN mirror.

keyval The keyval package by David Carlisle implements a system of defining and using a set of parameters, which are set using the syntax $<key> = <value>$.

Metapost MetaPost is a graphics language by John D.Hobby. It is very similar to METAFONT except that it ouptuts PostScript commands instead of bitmaps. Click here to go directly to the MetaPost home.

Omega Omega is an extension of TEX developed by John Plaice and Yannis Haralambous. It aims primarily at improving TEX's multilingual abilities. Visit the Omega project home page for more information.

PSTricks PSTricks (by Timothy van Zandt) is a collection of PostScript-based TEX macros that are compatible with most TEX macro packages.

PPower4 The PPower4 package by Klaus Guntermann for making dynamic presentations (requires Java). This is the homepage of PPower4.

pdfTEX By Hàn Thê Thành, Sebastian Rahtz & Hans Hagen. Creates pdf directly from TEX source files. Click here to go the pdfTEX support page. This is the location for Thành's source of pdfTEX.

pdfTricks The pdfTricks package by Radhakrishnan CV and Rajagopal CV enables PSTricks macros to be used with pdfTEX. The package can be found here.

pdfscreen The pdfscreen package by Radhakrishnan CV helps to redesign the pdf output of normal documents fit to be read in a computer monitor while retaining the freedom of conventional printing. You can get the pdfscreen package here.

pdfslide The pdfslide package by Radhakrishnan CV provides a easy way to make technical presentations with easily configurable options. The package can be obtained from CTAN.

TEXpower A package by Stephan Lehmke for creating dynamic online presentations with LATEX. The latest version of the package can be found here.

Xy-pic Xy-pic (by Kristoffer Høgsbro Rose & Ross Moore) is a package for typesetting graphs and diagrams. The package can be obtained from its home!

Courtesy/Source:

http://www.latex-project.org/intro.html
http://www.cse.iitk.ac.in/users/isciig/documents/user_ilatex.html
http://mathlab.cit.cornell.edu/support/latex/latex.html
http://www.image.ufl.edu/help/latex/
http://www.tug.org.in/packages.html