Interactive PDFs with LaTeX

LaTeX is a document preparation system. The basic version **TeX** is a sophisticated program designed to produce high-quality typesetting, especially for scientific and mathematical text. It takes a computer file, prepared according to the "rules" of TeX, and converts it to a form that may be printed on a high-quality printer, such as a laser writer, to produce a printed document of publication quality. The program is extendable by **packages** for a wide field of application. Which a few of this pakages it is possible to create interactive PDF files.

What is an interactive PDF file

An interactive PDF file is a document which includes directly

- embedded (you can directly act on it) or
- implemented (you can download it out of the PDF file and open it)

interactive elements (e.g. movie, quiz, animation). You can easily

- print the content of a interactive PDF file
- act on interactive elements
- hand it out

The technology to generate a interactive PDF file is based on LaTeX.

What you need first

1. TeX Distribution (basic version)
There are several TeX distributions for different operating systems available. Almost of them are freeware, e.g. one of this

- **TeXLive** (http://www.tug.org/texlive/) for Windows
- **MiKTeX** (http://www.miktex.org) for Windows
- **TeTeX** (http://www.tug.org/teTeX/) for Unix
- **MacTeX** (http://www.tug.org/mactex/) for Mac

In principle you can start to generate Postscript or PDF files now at the level of the command line. For a more comfortable handling of TeX/LaTeX there are some development environments and editors available which have a graphical user interface.

2. Development Environment / Editor
Almost of the practical development environments and editors are also freeware, eg. one of this

- **WinShell** (http://www.winshell.de/) for Windows
- **TeXnicCenter** (http://www.texniccenter.org) for Windows
- **Kile** (for KDE) (http://kile.sourceforge.net) for Unix
- **AUCTeX** (http://www.gnu.org/software/auctex/) for Windows, Unix, Mac
- **TeXShop** (http://www.uoregon.edu/~koch/texshop/index.html) for Mac

3. PDF Reader
To view the generated PDF file you need also the newest version of a PDF reader.

- **Adobe Reader** (http://get.adobe.com/de/reader/) for Windows, Unix, MacOS

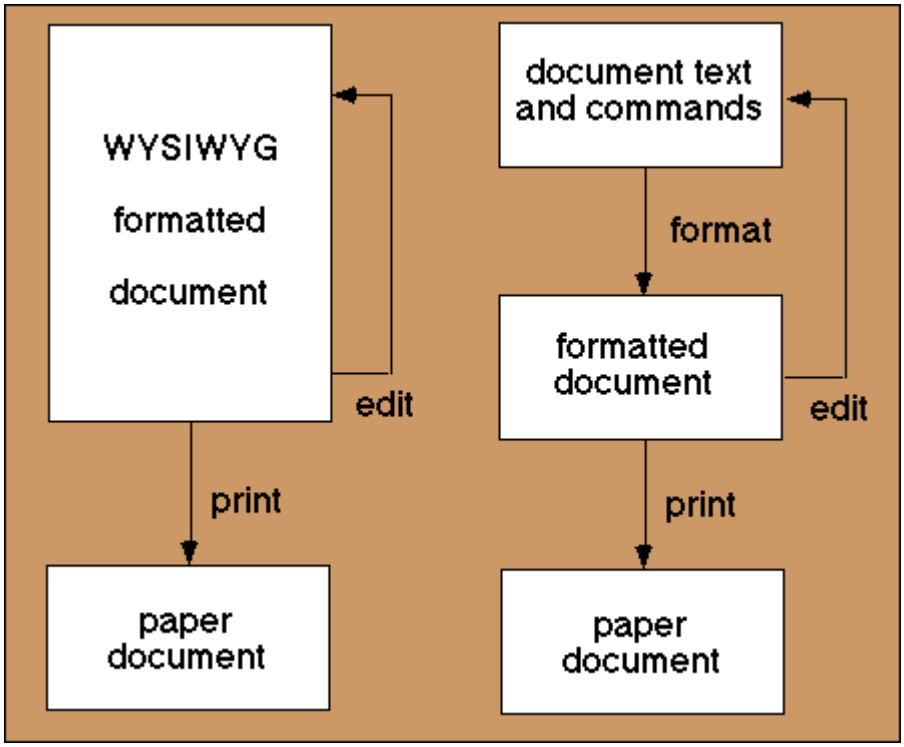
How LaTeX Works

In typical "WYSIWYG" text processors, such as Microsoft Word or Word Perfect:

- Formatting commands are invisible.
- The file shows pretty much the final result.

LaTeX, on the other hand, is a **formatter** rather than a text processor:

- The file includes commands that define structure.
- The formatting commands are visible.
- The process requires a compiler to format the final result.



(from http://www.cs.cornell.edu/info/Misc/LaTeX-Tutorial/Introduction.html)

How to use LaTeX

Here are the commands you should use to format LaTeX files from the command line. But `development environments and editors` actually provides a special “tex-mode”' with commands tailored to editing LaTeX files. Some of these commands are also given here.

- Using a `development environments or a editor`, **create a file** containing the text of your document. (See below for a sample). Give the file a name that ends in “.tex”', for example `first.tex.` **Save** your file before going on to the next step.
- To **format** your document:
  `<path-to-workspace>\`**`latex first.tex`**
  This creates a file called `first.dvi,` which is a binary file containing the formated version of your document. (It also creates some other files of no immediate interest).
- [Optional] To **view** your document on an X terminal
  `<path-to-workspace>\`**`xdvi first.dvi`**
  or on Windows
  `<path-to-workspace>\`**`windvi first.dvi`**
- To **convert** your document to a **PostScript** file:
  `<path-to-workspace>\dvips first.dvi`
  This creates a file called `first.ps.`
- To **print** your document:
  `<path-to-workspace>\lpr first.ps`
  (Use `lpr -Paries first.ps` if necessary).
- If you make **changes** your document (the `.tex` file), you must do all the above steps again.
- HINT: You don't need to type the file suffix for any of the above commands, except emacs; for example, you can view your file just by typing
  `<path-to-workspace>\xdvi first`
  I included the suffixes above to indicate which file each command operates on.

(adapted from http://www.ling.upenn.edu/advice/latex/starting.html)

A simple test file

Type the following file exactly as it appears (or, you may prefer to cut-and paste it). Do not confuse forward with backward slashes, or capital with lowercase letters. (It is ok to have additional spaces where there are some, or to start a new line anywhere you see a space). If you can get it to format, display on the screen, and print, you have mastered the process of formatting a LaTeX document.

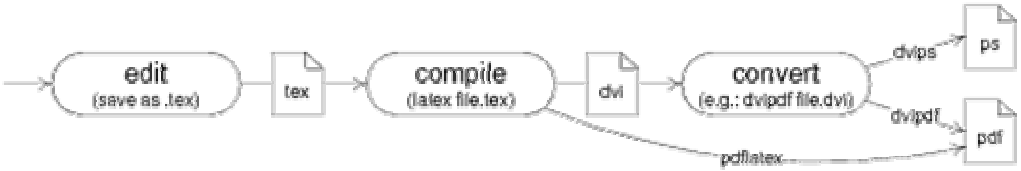
```
\documentclass{article}

%Preamble
% - declarations
% - formatting instructions
% - definitions
% - ... etc.

\begin{document}

%Comments and Text
This is a test \LaTeX\ document.   %further Comments

\end{document}
```

A more sophisticated test file

```
\documentclass[12pt]{article} % define class of document

% - declarations
 \usepackage[german]{babel}   % use package german with option babel
 \author{V.~Schmidt}          % author of the article
 \title{"Uber kurz oder lang} % title of the article

% - formatting instructions
 \setlength{\parindent}{0pt} % indent 1st row of a paragraph
 \setlength{\parskip}{5pt plus 2pt minus 1pt} % spacing between paragraphs
 \frenchspacing              % not to insert extra space at the end of sentences
 \sloppy                     % makes TeX less fussy about line breaking

\begin{document}             % begin document

 \maketitle % include title page

 \begin{abstract}            % begin abstract of article
  Beispiel f"ur einen Artikel in deutscher
  Sprache
 \end{abstract}              % end abstract of article

 \tableofcontents % include table of content

 \section{Start}             % section with title “Start”
  Hier beginnt der erste Abschnitt\dots

 \input{filename.tex} % include extern TeX files

 \section{End}               % section with title “End”
  \dots{} und hier endet es.

\end{document}               % end document
```

How to create interactive PDFs

LaTeX is a markup language for predefined TeX commands. PDFs can be created with the help of Latex. To generate a interactive PDF file use the command pdflatex instead of generating dvi first.



(from http://franz.kollmann.in/latex/latex.html)

A document needs the following mininum basic structure:

```
\documentclass[a4paper]{article}
\begin{document}
%Here is the code which Latex is supposed to interpret.
\end{document}
```

With this small example it is already possible to create your first Latex document and to compile it without an error message. In most cases several libraries are needed for Latex to be able to interpret other commands. Useful packages for the daily use are:

```
%insert graphic
\usepackage{graphicx}
% show umlaute (german)
\usepackage{ngerman}
\usepackage[T1]{fontenc}
\usepackage[latin1]{inputenc}
```

In order to be able to insert graphics, the package graphicx is required. It will be included if you indicate the following sequence of commands.

```
\usepackage{graphicx}
```

Make sure that all document-specific data is in front of

```
\begin{document}
```

Packages can only be declared before and only here it is possible to indicate the margin for the whole document. Example:

```
\documentclass[a4paper]{article}
%insert graphic
\usepackage{graphicx}
% show umlaute
\usepackage{ngerman}
\usepackage[T1]{fontenc}
\usepackage[latin1]{inputenc}
\begin{document}
%Here is the code which Latex is supposed to interpret.
\end{document}
```

Latex ignores lines which start with % as this command is reserved for comments.

For a interactive PDF the following structure is intended to be used:

```
\documentclass[a4paper]{article}

\usepackage{...}

\begin{document}

% include title page
\input{Titelseite.tex} % extern file
\newpage

% include impresssum / technical requirements
\input{impressum.tex} % extern file
\input{technik.tex} % extern file
\newpage

% include table of content
\tableofcontents
\newpage

% *** Learning Activity (Learning Scenario)
% include title page (includes short description, author, keywords, and link to further ressources, e.g. PPT-document embedded in PDF by package attachfile or a URL to the ressource in the COSMOS portal by package hyperref)
\section{Stundenbild: Sonnensystem}

\textbf{Dauer: }3x50 Minuten \\
\textbf{Keywords: }Sonnensystem, Sonne, Planeten, Flypen \\
\textbf{Tools und Materialien: }FlyPen, Software und Arbeitsblatt für FlyPen-Anwendung, PC mit Internetverbindung.\\
\textbf{Ziele und Zielsetzungen: }die Studierenden sollen die grundlegende Daten über die Sonne und die Planeten erfahren, sowie selbständig Daten über die Himmelskörper gewinnen und vergleichen.\\
\noindent
\textbf{Autor: }Alexander Nischelwitzer, Christian Reimers\\

File extern öffnen: \attachfile{stundenbild_sonnensystem_flypen.pdf} % attach the original file

\newpage % begin new page

% include Learning Activity
% e.g. including a PDF file
\includepdf[pages={1-9}, nup=1x2]{stundenbild_sonnensystem_flypen.pdf}

% ******************************************************************

% Learning Object 1 (Educational Content)
% include title and title page (includes short description, author, keywords, and link to further ressources, e.g. DOC-document embedded in PDF by package attachfile or a URL to the ressource in the COSMOS portal by package hyperref), e.g.
\section{Mondphasen - Teil 2}

Im nachfolgenden Video zeigt sich ein Ausschnitt über Mondphasen. Zum Starten des Videos in das Video klicken.\\

\textbf{Keywords: }Mond, Mondphasen, Video\\[1cm]
\textbf{Autor: }Mag. Wilfried Hartmann \\

File extern öffnen: \attachfile{Cosmos_Szenario_Erdmond_Mondphasen-02.mov } % attach the original file

% include Learning Object (Educational Content)
% e.g. including a movie
\includemovie[poster,text=(Mondphasen-02.mov),mouse]{1\linewidth}{.75\linewidth}{Cosmos_Szenario_Erdmond_Mondphasen-02.mov}
```

```
% ****************************************************************

% Learning Object 2 (Educational Content)

% ****************************************************************

% Learning Object 3 (Educational Content)

% ****************************************************************

% Learning Object 4 (Educational Content)

% ****************************************************************

\end{document}
```

As we now know the basic structure of a document, I will briefly explain the applicable packages for interactive PDFs.

Packages applicable for interactive PDFs

In the following chapters I will address in detail the packages which were necessary for establishing tutorials in order to produce interactive PDFs.

attachfile

Attachfile is a Latex package in order to send various other documents enclosed. Example: Movie15 Manual. To be able to use this package, it has to be included.

```
\usepackage{attachfile}
```

In order to include a file, the following command is used:

```
\attachfile [<options>] {<filename>}
```

I hardly used options, therefore I refer you again to the manual which you will find enclosed at the end of this chapter. In order to attach a file, the Latex document reads as follows:

```
\documentclass[a4paper]{article}
\usepackage{attachfile}
\begin{document}
\attachfile{attachfile.pdf}
\end{document}
```

It should be mentioned to no archive or executable files (.exe) can be attached. This was excluded for security reasons.

Manuals for package attachfile:

http://www.ctan.org/tex-archive/macros/latex/contrib/attachfile/

http://www.ctan.org/tex-archive/help/Catalogue/entries/attachfile.html

Manuals for package attachfile2:

http://www.ctan.org/tex-archive/help/Catalogue/entries/attachfile2.html

movie15

Movie15 is a package which imports multimedia content (films, audio, …). First, the package has to be imported with

```
\usepackage{movie15}
```

This package can be indicated with options. In order to be able to use options, these have to be written in [] of the package:

```
\usepackage[<package options>]{movie15}
```

The following options are possible:

- 3D: makes it possible to include three-dimensional files.
- draft: no multimedia file is inserted. Only the file name, as well as height and width are indicated.
- final: the opposite of final. Can be used to describe the global draft option. The exact command reference to include a multimedia file is:

```
\includemovie[<options>]{<width>}{<height>}{<media file>}
```

Not all fields are mandatory fields. Thus, <width> and <height> and <options> can be left blank. In most cased I filled the fields <width> and <height> to make sure the width-to-height ratio of the included files does not get lost. Options which can be indicated for including a multimedia file differ from the options used for including packages. In order to see a list of all options, please be referred to the manual which you will find enclosed at the end of this chapter. The options I used were:

- autoplay: starts the automatic playing of a file as soon as the page is opened.
- mouse[=true|false]: allows interaction with the mouse such as start and stop of the content.
- text: text which is shown in the middle of the included file during loading.

In order to include the following video, the Latex document reads as follows:

```
\documentclass[a4paper]{article}
%insert movie15
\usepackage{movie15}
\begin{document}
```

```
\includemovie[mouse, autoplay]{0.5\linewidth}{.375\linewidth}{fsoe.MPG}
\end{document}
```

Manuals for package movie15

http://www.ctan.org/tex-archive/macros/latex/contrib/movie15/

http://www.ctan.org/tex-archive/help/Catalogue/entries/movie15.html

AcroTeX eDucation Bundle

AcroTex is a very big package in order to insert quizzes into a PDF.

To import the package, it has to be included with:

```
\usepackage[<options>]{exerquiz}
```

In the following you will find a description and an example of a quiz in Latex

```
\usepackage[noquizsolutions]{exerquiz}
```

imports the exerquiz package with the option that no results are included in the latexfile.

```
 \useBeginQuizButton \useEndQuizButton
```

defines that buttons will be used for starting and ending the test

```
\begin{quiz*}{sonnenfinsternis}
```

defines that a quiz environment is created with the name ‚Sonnenfinsternis'.

```
\begin{questions}
```

indicates that the questions and answers of the quiz starts

```
\item Text of the Question
```

indicates a new question.

```
\begin{answers}*{2}
```

indicates that the possible answers to a question start and that these are broken down in 2 columns

```
\bChoices \eChoices
```

defines a segment for the answers. In between are all possible answers to a question :

```
\Ans0 Text of an Answer \eAns   % This answer is wrong.

\Ans1 Text of an Answer \eAns   % This answer is right.
```

```
\end{answers}
```

indicates that the possible answers to a question ends

```
\end{questions}
```

indicates that the questions and answers of the quiz ends

```
\end{quiz*}
```

indicates that a quiz environment ends

```
\ScoreField\currQuiz
```

allows the user to view the number of correct results for exactly this test after clicking the ‚Test End Button'.

```
\eqButton\currQuiz
```

allows the user to view all results for exactly this test after clicking the ‚Test End Button'.

Manual for package Acrotex

http://www.acrotex.net/aeb_index.php?lang=de&page=what_

http://www.math.uakron.edu/~dpstory/webeq.html

http://www.ctan.org/tex-archive/macros/latex/contrib/acrotex/

http://www.ctan.org/tex-archive/help/Catalogue/entries/acrotex.html

pdfpages

pdfpages includes a file directly into the PDF which was created with Latex. The user does not have to open the file again, but can see it directly attached. In order to be able to use this package, it has to be included.

```
\usepackage[hoptionsi]{pdfpages}
```

In order to attach a file, the following command is used:

```
\includepdf[<option>]{<filename>}
```

Be careful with the file name. It must not include space characters! Options for attaching a file can be found in the manual, I used the following ones several times:

- pages: indicates the pages which ought to be included
- nup: includes several pages of the file to be included on one page of the PDF
- landscape: specifies the format. Is turned by 90 degrees (e.g. portrait format vs. landscape format)

On the following pages all options from the manual for pdfpages were included. The Latex document is as follows:

```
\documentclass[a4paper]{article}
% integrate pdfs
\usepackage{pdfpages}
\begin{document}
\includepdf[pages={2-8}, nup=1x1]{pdfpages.pdf}
\end{document}
```

When producing DVI output pdfpages cannot insert pages of a PDF documents.
But instead of interrupting execution pdfpages will insert empty pages.
This feature is important when using packages like pst-pdf, which need to produce DVI output at the first run.

Manuals for package pdfpages

http://www.ctan.org/tex-archive/help/Catalogue/entries/pdfpages.html

http://www.ctan.org/tex-archive/macros/latex/contrib/pdfpages/pdfpages.pdf

hyperref

The hyperref package activates links (e.g. to internet addresses and within the document). In order to be able to use this package, it has to be included

```
\usepackage[Optionen]{hyperref}
```

The following options are possible:

- breaklinks=true/false: indicates whether links may be broken.
- linktocpage=true/false: in the table of contents only the page numbers are links, but not the text.
- colorlinks=true/false: links are coloured (colours are defined with linkcolor, anchorcolor ...)
- linkcolor=colour: colour of the linked text, links within the document
- citecolor=colour: colour of the linked text, links to the bibliography
- lecolor= colour: colour of the linked text, links to local files
- pagecolor= colour: colour of the linked text, links to other pages
- urlcolor= colour: colour of the linked text, external URLs
- frenchlinks=true/false: links are shown as smallcaps instead of coloured
- breaklinks=true/false: indicates whether links may be broken.

In order to insert a link, there are the following possibilities:

\href{URL}{text}: the indicated text is implemented as a link on URL

\url{URL}: the indicated URL is shown and activated as a link

\nolinkurl{URL}: the indicated URL is shown, but not marked as a link.

Example Latex Code:

```
\documentclass[a4paper]{article}
```

```
% Clickable links in PDF file
\usepackage{hyperref}
\begin{document}
\url{http://www.tug.org/applications/hyperref/manual.html}\\
\href{http://www.jkrieger.de/tools/latex/hyperref.html}{Kleines Tutorial}\\
\nolinkurl{http://www.tug.org/applications/hyperref/manual.html}\\
\end{document}
```

Manuals for package hyperref

http://www.ctan.org/tex-archive/macros/latex/contrib/hyperref/

http://www.ctan.org/tex-archive/help/Catalogue/entries/hyperref.html

http://www.tug.org/applications/hyperref/manual.html

http://www.jkrieger.de/tools/latex/hyperref.html

Help to LaTeX/TeX

There are also various tutorials, HowTos and communities available which can provide you with the solution if a problem occurs. In many Languages (bulgarian, chinese, dutch, english, finnish, french, german, italian, japanese, korean, mongolian, polish, portuguese, russian, slovak, slovenian, spanish, thai, turkish, ukrainian, vietnamese) the dosument “The Not So Short Introduction to LaTeX2e” is available for download under http://tug.ctan.org/tex-archive/info/lshort/ .

In German:

LaTeX 2e-Kurzbeschreibung
ftp://ftp.dante.de/tex-archive/info/german/LaTeX2e-Kurzbeschreibung/l2kurz.pdf

Kochbuch für LaTeX
http://www.dante.de/TeX-Service/TSP/tex/cookbook/cookbook.html

LaTeX-Tutorium
http://latex-tutorium.sourceforge.net/latex-tutorium.pdf

LaTeX Tips und Tricks
http://www.fernuni-hagen.de/IT/team/duering/tiptrick.pdf

In English:

Getting Started with LaTeX
http://www.maths.tcd.ie/~dwilkins/LaTeXPrimer/GSWLaTeX.pdf

LaTeX Tutorials
http://www.tug.org/twg/mactex/tutorials/ltxprimer-1.0.pdf

Begin LaTeX – Formatting Information
http://tug.ctan.org/tex-archive/info/beginlatex/

Getting to grips with LaTeX
http://www.andy-roberts.net/misc/latex/