

# Introduction to Investigation

## L<sup>A</sup>T<sub>E</sub>X, an introduction

Arno Formella, Juan Carlos González Moreno

Escola Superior de Enxenaría Informática  
Universidade de Vigo

Feb 2010

# Part I

## Program

# Learning results

- know, classify, and analyze methods to study bibliography
- know, classify, and analyze research projects
- know, analyze, and apply methods to handle scientific data
- know and analyze legal, ethical, and social aspects of research

# Contents

- Research principals in computer science
- Writing and reading English articles
- Write and review scientific articles
- Write research projects
- Plan, realize, and control research projects
- Text processors, data analysis, data management
- Visualization tools
- Legal, ethical, and social aspects of research

# Scheduling

- 3 ECTS
- 75 hours
- 15 class room hours with professor
- 5 basic modules/tasks
- close relation with Master's thesis

# Modules/Tasks

- tools to work with
- state-of-the-art of a research field
- study of related articles
- writing of an article
- writing of a research proposal

# Tools to work with

- $\text{\LaTeX}$  (Arno)
- JabRef (Arno)
- scopus, google-scholar, ieee-xplore, springerlink, acmportal, citeseerX, ... (Antón)
- grace, matplotlib, gnuplot (David)

# State-of-the-art + study

- work to start with for your Master's thesis (Tutor)
- search for related articles and research projects or groups
- collect articles and analyze
- look what others do



# Article

- Write down the state-of-the-art
- with annotated bibliography
- in  $\text{\LaTeX}$  according to a predefined conference or journal style
- for your proposal of your Master's project
- (Tutor, Arno, David)

# Research proposal

- how to get financing for research projects
- criteria to consider
- writing a proposal
- (Juan Carlos)

# Inclusion

- Patents
- Author rights
- Software Licenses
- Protection of data
- Reproducible research
- (Floro)

## Part II

## Remark

# Binary or ASCII

- **all** data should be stored in **human readable** format
- with (open) **standard compression** algorithm

# Human readable files

## advantages

- searchable
- formatable
- modifyable
- extendable
- annotable
- portable

# Binary Files

think twice before using them

- there are only two reasons to use binary data
  - efficiency due to repeated or random access
  - you have really binary data (e.g. images)

# Binary or ASCII

an example

writing 100000 double precision floating point numbers

```
1564841 2006-06-22 18:48 ascout.dat
800000 2006-06-29 14:45 binout.dat
487380 2006-06-29 14:45 ascout_10.dat.bz2
475510 2006-06-22 18:37 binout.dat.bz2
469465 2006-06-29 14:45 ascout.dat.bz2
```



# Modular documents

why you should work with small parts

- reusable units
- small file units
- distributed authoring
- fine grain version control

# Part III



# The T<sub>E</sub>X System

what do you need

## The system

### Document Structure

### Fonts

### Math

### Tables

### Graphics

### Floating objects

### What else

### The Beamer Class

### Summary

- a T<sub>E</sub>X L<sup>A</sup>T<sub>E</sub>X, and/or pdfL<sup>A</sup>T<sub>E</sub>X system  
(all files are **human readable**, besides:  
executables and compressed font description)
- your `.tex`-file(s)
- other files (like graphics and bibliography) to include

# The T<sub>E</sub>X System

what is it

## The system

### Document Structure

### Fonts

### Math

### Tables

### Graphics

### Floating objects

### What else

### The Beamer Class

### Summary

- sort of programming language
- works on almost any operating system
- **best** typesetting program around
- you **program** the lookalike of your document
- open source (since approx. 30 years)
- invented and donated by **D. Knuth**

# Document Structure

## structured documents

- first line
  - document class, global settings
- preamble
  - packages to include
  - command and environment definitions
  - layout specifics
- main document
  - all you want to write

# Document Structure

example document

The system

Document  
Structure

Fonts

Math

Tables

Graphics

Floating  
objects

What else

The Beamer  
Class

Summary

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

```
\usepackage{times}
```

```
\title{Advanced Documentation with \LaTeX}
```

```
\author{Arno Formella}
```

```
\begin{document}
```

```
\maketitle
```

```
Hello World
```

```
\end{document}
```

# Programming Conventions

the hard part

The system

Document  
Structure

Fonts

Math

Tables

Graphics

Floating  
objects

What else

The Beamer  
Class

Summary

- all commands of  $\text{\LaTeX}$  start with a  $\backslash$
- all environments are surrounded by a correctly nested pair  
 $\backslash\text{begin}\{\text{name}\} \dots \backslash\text{end}\{\text{name}\}$
- you can form blocks (same as in C/C++ or Java) using  $\{$  and  $\}$  to confine actions of commands (i.e.,  $\text{\LaTeX}$  works like a pushdown automaton)
- you can define your own commands and environments

# Document Structure

the classes

The system

Document  
Structure

Fonts

Math

Tables

Graphics

Floating  
objects

What else

The Beamer  
Class

Summary

- **the classical ones**  
article, book, letter, report, slides
- **some special ones**  
amsart, amsbook, amsproc, **beamer**,  
xatcobeo, lia, ... etc.
- **and journal styles**  
IEEEtran, sgmlmanual, llncs, egpubl,  
elsart, ...



# Document Structure

all you need

The system

Document  
Structure

Fonts

Math

Tables

Graphics

Floating  
objects

What else

The Beamer  
Class

Summary

- **some of the common packages:**  
a4wide, afterpage, amfonts, amssymb,  
array, caption, color, colortbl,  
datetime, dcolumn, fancybox, fancyhdr,  
float, geometry, graphicx, html,  
hyperref ifthen, lastpage, listings  
listings, longtable, lscape, makeidx,  
marginnote, pdfpages, pgf, sectsty,  
tlenc, url, verbatim  
**etc.**

# Document Structure

collection

The system

Document  
Structure

Fonts

Math

Tables

Graphics

Floating  
objects

What else

The Beamer  
Class

Summary

- more than 1000 commands and environments
- huge set of prepared packages
- everything searchable and documented (google, ctan, tex–distributions)

# Document Structure

the internals of a document

The system

Document  
Structure

Fonts

Math

Tables

Graphics

Floating  
objects

What else

The Beamer  
Class

Summary

- different structure depending on document class
- structural document division in  
part, chapter, section, subsection,  
subsubsection, paragraph, subparagraph
- automatic (optional) enumeration
- automatic (optional) table of content (and other tables)

# Document Structure

lists of lists

The system

Document  
Structure

Fonts

Math

Tables

Graphics

Floating  
objects

What else

The Beamer  
Class

Summary

`description` lists different parts with highlighted words

`enumerate` lists different parts using some type of  
enumeration

`itemize` lists simply parts with some type of marks

`list` the base for general lists

# Document Structure

example description

The system

Document  
Structure

Fonts

Math

Tables

Graphics

Floating  
objects

What else

The Beamer  
Class

Summary

```
\newlength{\thiswidth}  
\settowidth{\thiswidth}{  
  {\ttfamily description}}  
\labelwidth1.2\thiswidth  
\begin{description}  
\item[\ttfamily description] lists ...  
\item[\ttfamily enumerate] lists ...  
\item[\ttfamily itemize] lists ...  
\item[\ttfamily list] the base ...  
\end{description}
```

# Document Structure

example enumeration

The system

Document  
Structure

Fonts

Math

Tables

Graphics

Floating  
objects

What else

The Beamer  
Class

Summary

```
\begin{enumerate}
\item
  first item, level 0
  \begin{enumerate}
  \item first item, level 1
  \item second item, level 1
  \item third item, level 1
  \end{enumerate}
\item
  second item, level 0
\item
  third item, level 0
\end{enumerate}
```

# Document Structure

example enumeration

- 1 first item, level 0
  - 1 first item, level 1
  - 2 second item, level 1
  - 3 third item, level 1
- 2 second item, level 0
- 3 third item, level 0

# Document Structure

description, enumeration, and itemize

- you can nest everything
- there exist 4 predefined levels
- there are parameters to control spaces and margins
- there are parameters to modify the symbols



# Font

## Font Size

- the font size is defined implicitly
- the effective size depends on the document class
- commands: `tiny`, `small`, etc.

## Example

`tiny` `scriptsize` `footnotesize` `small` `normalsize` `large`

Large LARGE huge Huge

## Example

tiny scriptsize footnotesize small normalsize large

Large LARGE huge Huge

## Example

*tiny scriptsize footnotesize small normalsize large*

*Large LARGE huge Huge*

## Example

tiny scriptsize footnotesize small normalsize

Large LARGE huge Huge

## Mathematical Mode

writing formulas is easy

$$P_{rec} = \left| \sqrt{\frac{\epsilon D_0}{2\eta} \cdot \frac{\lambda^2}{4\pi} \sum_{\forall \Theta, \Phi} E(\Theta, \Phi) \sqrt{F(\Theta, \Phi)}} \right|^2$$

```
\[
  P_{rec}=
  \left| \sqrt{
    \frac{\epsilon D_0}{2\eta} \cdot
    \frac{\lambda^2}{4\pi}
    \sum_{\forall \Theta, \Phi}
    E(\Theta, \Phi) \sqrt{F(\Theta, \Phi)}
  } \right|^2
\]
```

$$P_{rec} = \left| \sqrt{\frac{\epsilon D_0}{2\eta} \cdot \frac{\lambda^2}{4\pi} \sum_{\forall \Theta, \Phi} E(\Theta, \Phi) \sqrt{F(\Theta, \Phi)}} \right|^2$$

```
\[
  P_{rec}=
  \left| \sqrt{
    \frac{\epsilon D_0}{2\eta} \cdot \frac{\lambda^2}{4\pi}
    \sum_{\forall \Theta, \Phi}
    E(\Theta, \Phi) \sqrt{F(\Theta, \Phi)}
  }
  \right|^2
\]
```

# Mathematical Mode

writing formulas is easy

$$\begin{aligned}
 |E_{\Psi}(\Theta, \Phi)| &= \sqrt{\frac{\eta D_0 \epsilon P_t}{2\pi}} \sqrt{F(\Theta, \Phi)} \frac{|e^{-jkr}|}{r} \\
 &= \sqrt{\frac{2\eta \epsilon P_t}{\int_0^{2\pi} \int_0^{\pi} F(\Theta, \Phi) \sin \Theta \, d\Theta d\Phi}} \\
 &\quad \cdot \sqrt{F(\Theta, \Phi)} \frac{|e^{-jkr}|}{r}
 \end{aligned}$$

## Mathematical Mode

writing formulas is easy

$$\sqrt{1 + \sqrt{1 + \sqrt{1 + \sqrt{1 + \sqrt{1 + \sqrt{1 + \sqrt{x}}}}}}}$$

```
\[
  \sqrt{1+\sqrt{1+\sqrt{1+\sqrt{
    1+\sqrt{1+\sqrt{1+\sqrt{x}}}}}}
  }}}}
```

## Mathematical Mode

writing formulas is easy

$$\sqrt{1 + \sqrt{1 + \sqrt{1 + \sqrt{1 + \sqrt{1 + \sqrt{1 + \sqrt{x}}}}}}}$$

```
\[
  \sqrt{1+\sqrt{1+\sqrt{1+\sqrt{
    1+\sqrt{1+\sqrt{1+\sqrt{x}}}}}}
  }}}}
```

# Mathematical Mode

complete math typesetting

The system

Document  
Structure

Fonts

Math

Tables

Graphics

Floating  
objects

What else

The Beamer  
Class

Summary

- (almost) all mathematical symbols are available
- different types of characters  
(e.g., bold, roman, etc.)
- (almost) all common structures  
(e.g., fractions, indices, matrices, roots etc.)
- you can use macros



# Mathematical Mode

macro math for the lazy

The system

Document  
Structure

Fonts

**Math**

Tables

Graphics

Floating  
objects

What else

The Beamer  
Class

Summary

$$(a_0, a_1, a_2), (b_0, b_1, b_2), (c_0, c_1, c_2)$$

$$(A_0, A_1, A_2)_0, (B_0, B_1, B_2)_1, (C_0, C_1, C_2)_2$$

$$(A_{\alpha 0}, A_{\alpha 1}, A_{\alpha 2}), (B_{\beta 0}, B_{\beta 1}, B_{\beta 2}), (C_{\gamma 0}, C_{\gamma 1}, C_{\gamma 2})$$

# Mathematical Mode

macro math for the lazy

The system

Document  
Structure

Fonts

**Math**

Tables

Graphics

Floating  
objects

What else

The Beamer  
Class

Summary

```
\newcommand{\Vector}[1]{  
  ({#1}_0, {#1}_1, {#1}_2)}
```

```
\[ \Vector a, \Vector b, \Vector c \]  
\[ \Vector A_0, \Vector B_1, \Vector C_2 \]  
\[ \Vector{A_\alpha},  
  \Vector{B_\beta},  
  \Vector{C_\gamma} \]
```

# Tables

writing tables is as easy

|       |       |         |
|-------|-------|---------|
| this  | is    | a       |
| table | with  | several |
| rows  | and   | columns |
| and   | lines | !       |

```
\begin{center}\begin{tabular}{c|l|r}
this & is & a \\\hline
table & with & several \\
rows & and & columns \\
and & lines & !\\
\end{tabular}\end{center}
```

# Tables

writing tables is as easy

|       |       |         |
|-------|-------|---------|
| this  | is    | a       |
| table | with  | several |
| rows  | and   | columns |
| and   | lines | !       |

```
\begin{center}\begin{tabular}{c|l|r}
this & is & a \\\hline
table & with & several \\
rows & and & columns \\
and & lines & !\\
\end{tabular}\end{center}
```

The system

Document  
Structure

Fonts

Math

Tables

Graphics

Floating  
objects

What else

The Beamer  
Class

Summary

# Graphics

which types can you include

- packages to realize simple graphics directly in  $\text{\LaTeX}$
- include graphics in other formats
  - Postscript in normal  $\text{\LaTeX}$
  - PDF and JPEG in  $\text{pdf\LaTeX}$

The system

Document  
Structure

Fonts

Math

Tables

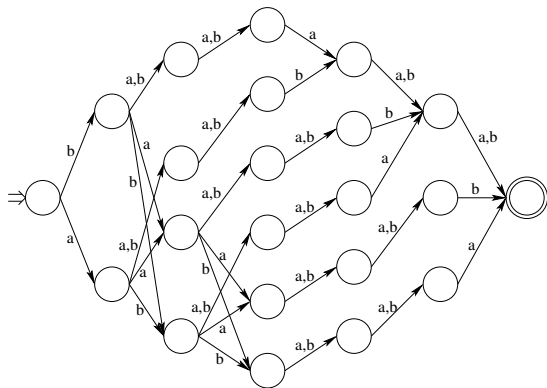
Graphics

Floating  
objects

What else

The Beamer  
Class

Summary



# Graphics

including a PDF

```
\centerline{%  
  \includegraphics [width=.7\textwidth]%  
  {afndln}  
}
```

# Graphics

including a JPEG

The system

Document  
Structure

Fonts

Math

Tables

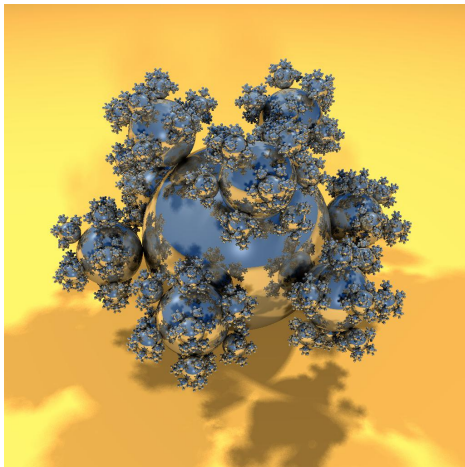
**Graphics**

Floating  
objects

What else

The Beamer  
Class

Summary





# Graphics

including a JPEG

```
\centerline{%  
  \includegraphics [width=.6\textwidth]%  
  {ballsp}  
}
```

# Floating Objects

automatically placed

- floating objects are figures, tables, images etc.
- $\text{\LaTeX}$  puts these objects according to rules (e.g., only at the top of a page)
- permits captions for floating objects
- generates table of figures, table of tables etc.

# Floating Tables

placed where it fits well

|       |       |         |
|-------|-------|---------|
| this  | is    | a       |
| table | with  | several |
| rows  | and   | columns |
| and   | lines | !       |

**Table:** A nicely placed table.

# Floating Tables

easy to define

The system

Document  
Structure

Fonts

Math

Tables

Graphics

**Floating  
objects**

What else

The Beamer  
Class

Summary

```
\begin{table}[h]
\begin{center}\begin{tabular}{c|l|r}
this & is & a \\\hline
table & with & several \\
rows & and & columns \\
and & lines & !\\
\end{tabular}\end{center}
\caption{A nicely placed table.}
\end{table}
```





# What else

there is much more in L<sup>A</sup>T<sub>E</sub>X

- **cross references with `\label \ref` pairs**
- references to a bibliography with `\cite` command (and a corresponding database).
- footnotes
- lines and boxes
- margin text
- head and foot lines
- index generation
- and more and more and ...

# What else

there is much more in  $\text{\LaTeX}$

- cross references with `\label \ref` pairs
- references to a bibliography with `\cite` command (and a corresponding database).
- footnotes
- lines and boxes
- margin text
- head and foot lines
- index generation
- and more and more and ...



# What else

there is much more in  $\text{\LaTeX}$

- cross references with `\label \ref` pairs
- references to a bibliography with `\cite` command (and a corresponding database).
- footnotes
  - lines and boxes
  - margin text
  - head and foot lines
  - index generation
  - and more and more and ...

# What else

there is much more in  $\text{\LaTeX}$

- cross references with `\label \ref` pairs
- references to a bibliography with `\cite` command (and a corresponding database).
- footnotes
- lines and boxes
  - margin text
  - head and foot lines
  - index generation
- and more and more and ...

# What else

there is much more in  $\LaTeX$

- cross references with `\label \ref` pairs
- references to a bibliography with `\cite` command (and a corresponding database).
- footnotes
- lines and boxes
- margin text
- head and foot lines
- index generation
- and more and more and ...

# What else

there is much more in  $\text{\LaTeX}$

- cross references with `\label \ref` pairs
- references to a bibliography with `\cite` command (and a corresponding database).
- footnotes
- lines and boxes
- margin text
- head and foot lines
- index generation
- and more and more and ...

# What else

there is much more in  $\text{\LaTeX}$

- cross references with `\label \ref` pairs
- references to a bibliography with `\cite` command (and a corresponding database).
- footnotes
- lines and boxes
- margin text
- head and foot lines
- index generation
- and more and more and ...

# What else

there is much more in  $\text{\LaTeX}$

- cross references with `\label \ref` pairs
- references to a bibliography with `\cite` command (and a corresponding database).
- footnotes
- lines and boxes
- margin text
- head and foot lines
- index generation
- and more and more and ...

# Personal Comments

you should make up your mind

The system

Document  
Structure

Fonts

Math

Tables

Graphics

Floating  
objects

What else

The Beamer  
Class

Summary

- use macros and environments
- avoid changing established behavior
- think about content, not shape
- write to take advantage of
  - actual text editor (e.g., `gvim` or `emacs`)
  - possible reuse of parts (especially macros)
- generate parts of the document automatically
- use a good writing style (same as with programs)

# Presentations

make your slides as writing text

- this presentation is made with  $\text{L}^{\text{A}}\text{T}_{\text{E}}\text{X}$  and `beamer class`
- has almost everything you need
- automatic styles and themes



# How to use

it might take some time

- read a book on  $\text{\LaTeX}$
- look into files of others
- ask those who know
- google

# How to use

## use makefile

- write your `.tex`-files
- place your `.bib`-files and other dependent files (like graphics)
- run `latex` or `pdflatex` sufficiently often (might not converge in pathological cases!)
- use tools to convert to final format
- or simply use the makefile

# Pros and Cons

it is your choice

nice  
easy  
complete  
powerful  
flexible  
compact  
compatible  
extendible  
coherent  
free  
unixable

no WYSIWYG  
(but there is lyx)  
you need certain level of  
abstraction

# Part IV

## Bibtex

# The BibT<sub>E</sub>X system

- tool and file format
- to describe and process lists of references
- mostly for L<sup>A</sup>T<sub>E</sub>X-documents
- but usable in other environments as well
- BibDesk (Mac OS), Bibshare (MS Word)

# File format

- **definition of strings**  
`@string{ foo = "Anton Foo" }`
- **definition of preambles**  
`@preamble{ This is an preamble. }`
- **definition of entries with field pairs**  
`@entry{ field = value, field = value, ... }`
- **field pairs are, for instance,**  
`author = {Formella, Arno}`  
`title = "My great first written work"`
- **values can be integer numbers**

# Example BibT<sub>E</sub>X file

```
@Book{landru-1921-hundred,  
  author = {Landru, Henri D\'esir\'e},  
  title = {A hundred recipes for you wife},  
  publisher = {Culinary Expert Series},  
  year = 1921  
}
```

## Some remarks

- a comment is everything outside an entry
- `@comment` entries are comments
- entry names and fields are case insensitive
- usage of `{ }` and `"` is tricky  
(maintaining capital letters, use of accents etc.)
- automatic author name parsing  
(I recommend the form: `lastname(s), firstname(s)`)
- defined strings are simply used or concatenated with `#`



## Example usage

```
\documentclass[11pt]{article}  
\usepackage{cite} % optional  
\begin{document}
```

```
\title{Using reference}  
\author{A. Formella}  
\date{Today}  
\maketitle
```

```
This is my first reference  
\cite{formella-2010-first}.
```

```
\bibliography{mybib}{}  
\bibliographystyle{plain}  
\end{document}
```

# Some explanations

- the argument of the `cite` command is the key of the bib-entry
- various citations can be inclosed separated by coma
- the package `cite` sorts, groups etc. for numerical style
- the style defines how the reference entries are typeset
- various bib-files can be given in `bibliography`

# Standard entry types and fields

- article, book, booklet, conference, inbook, incollection, inproceedings, manual, mastersthesis, misc, phdthesis, proceedings, techreport, unpublished
- abstract, address, annote, author, booktitle, chapter, copyright, crossref, edition, editor, eprint, howpublished, institution, ISBN, ISSN, journal, key, month, note, number, organization, pages, publisher, school, series, title, type, url, volume, year, (etc.)
- each entry requires certain fields and allows others as optional (depending on bibtex style used)
- however, you can use any field and even create your own style...

# BibTeX styles

- IEEE, IEEEannot, abbrev, abbrvnat, acm, **alpha**, alphanum, ama, ametsoc, amsalpha, amsplain, annotate, annotation, apalike, apasoft , cell, chicagoa, cj, decsci, development, finplain, humanbio, **ieeetr**, is-abbrev , is-alpha, is-plain, is-unsrt, jas99, jbact, jqt1999, jtb, jtbnew, nar, **nature**, neuron, phjcp, **plain**, plainnat, plainyr, siam, these, **unsrt**, **unsrtnat**, wmaainf

# JabRef: reference management software

- easy-to-use interface for editing BibT<sub>E</sub>X files
- importing data from online scientific databases
- maintained for Windows, Linux and Mac OS X
- on-going development (current Nov 1st, 2009, v2.5)
- Java based, web-start possibility
- there are more systems, take a look at  
[http://en.wikipedia.org/wiki/Comparison\\_of\\_reference\\_management\\_software](http://en.wikipedia.org/wiki/Comparison_of_reference_management_software)

# JabRef: features

- fully BibTeX compliant output
- grouping feature for organizing reference entries
- import filters for 15 reference formats
- direct search and download from PubMed and IEEEXplore, direct download from CiteSeer and ArXiv.
- Links to external resources
- Cite-as-you-write functionality for external applications such as LyX, Kile, Emacs, WinEdt, Vim and L<sup>A</sup>T<sub>E</sub>X-Editor (LEd)
- customizable export filters
- Plug-in system (based on Java Plugin Framework)

# Summary

- Use the right tool at the right time.
- Use whenever possible **human readable** files.
- Learn **English**.