

Introducción a L^AT_EX

Fernando Oleo Blanco

fernando.oleo@alu.comillas.edu

[github.com/Irvis/Documents](https://github.com/Irvis/ Documents)

16 de marzo de 2020

(ICAI - LinuxEC) DEP

Índice

Historia

Instalación y recursos

Comparativa con Word

Estructura del documento

`documentclass` y preámbulo

Manejo del texto

Entornos comunes

Referencias y bibliografía

Escritura científica

Resumen y otros recursos

1. Diseño del documento
2. Configuración del documento
3. Estructuración del texto
4. Herramientas para el trabajo de texto
5. Entornos útiles
6. Referencias y bibliografía
7. Escritura científica
8. Recursos extra

Historia



Figura 1: Donald Ervin Knuth. Creador de T_EX

Un pequeño cuento

¿Quién es Knuth?

Americano. Profesor de Stanford, ya retirado. Matemático, físico, informático y teólogo. Actualmente escribe la serie de libros *The Art of Computer Programming*, precursora del nacimiento de $\text{T}_{\text{E}}\text{X}$. Considerado uno de los padres de la informática moderna

$\text{T}_{\text{E}}\text{X}$

Después de crear el segundo volumen y empezar el tercero se dio cuenta que la tipografía carecía calidad. Buscó soluciones y decidió estudiar tipografía para crearse su propio sistema. $\text{T}_{\text{E}}\text{X}$ es el entorno de programación, \LaTeX es $\text{T}_{\text{E}}\text{X}$ y unos paquetes para agilizar su escritura

"Si una herramienta que uso la utilizan muchas personas, seguramente pensaría que estoy haciendo algo mal"

Instalación y recursos

Instalación

TexStudio, IDE

- **TeXStudio:** Download → busca tu plataforma. Instálalo como solo tú sabes

L^AT_EX 2_ε

- **"Compilador"**

Windows: usad o MikTeX o TeXLive. Texlive es el tradicional

Mac: instalad MacTeX y listo

Linux: buscad TeXLive en vuestra distribución

Recursos on-line

Accesibles desde el link anterior. Es una buena idea tener una copia en la nube. Recomiendo Overleaf, recientemente fusionado con ShareL^AT_EX

Recursos recomendados

Lectura

- *The not so Short Introduction to \LaTeX* por Tobias Oetiker
- *\LaTeX Wikibook*: Libro escrito por y para Wikipedia. El 99 % de vuestras dudas tienen solución aquí
- *More Math Into \LaTeX* por George Grätzner (esta es una buena muestra)

Internet

- Cualquier servicio con plantillas (**Latextemplates** por ejemplo)
- **Tug**: Centro de recursos *oficiales*
- Foros (**Overleaf-learn**), "puntos de información", etc
- Google

Comparativa con Word

Diferencias notables

Microsoft Word

- Intuitivo, fácil de usar
 - Ya conocido
 - Imágenes, tablas, etc se hacen solas
-

- ¿Bibliografía?
- ¿Índice?
- ¿Referencias?

L^AT_EX

- Complicado, tedioso
 - Con un error, ya nada funciona
 - Escribirlo todo manualmente...
-

- Estructura automática
- Texto de calidad sin esfuerzo
- No da problemas a dos semanas antes de la entrega

Buenas prácticas generales

1. Cuando algo falla, **leed el mensaje de error**
2. Nunca, nunca, nunca empecéis desde cero.
¡Usad plantillas!
3. Sed organizados
4. **Haced las cosas sencillas**, si no es obvio, no lo hagas
5. Buscad ayuda (en mi o en los recursos mencionados)
6. ¡Comentad lo que hacéis! % Comentario
7. **Leed la documentación**
8. **Usad bien el IDE.** ¡Ahorra un montón de

Estructura del documento

Estructura de archivos para el trabajo

Estructura de archivos recomendada. Es la que usa la plantilla

```
/
├── main.tex
├── biblio.bib
├── ...
├── Cap1
│   ├── cap1.tex
│   ├── img.png
│   ├── main.pdf
│   └── ...
└── Cap2
    └── ...
```

En \LaTeX podemos, y se recomienda, dividir nuestro archivo en partes pequeñas y en carpetas. Esto permite estructurar mucho mejor el documento, mantener los archivos ordenados, y trabajar con textos menores.

Estructura general de los comandos i

Comando simple

Comienzan con `\`, seguido del comando. Si este comando recibe algún argumento (o algunos), estos van entre llaves. Si reciben opciones, van entre corchetes antes del argumento. Ejemplos:

`\hrulefill` → _____

`\textit{Hola}` → *Hola*

`\textcolor{blue}{azul}` → azul

Estructura general de los comandos ii

Entornos

Como comandos normales, pero cuya función es más extensa y compleja; tienen la estructura:

```
\begin{entorno}[opciones]{argumento}  
  content...  
\end{entorno}
```

En IDE Ctrl + e

Comienzo de nuestro documento

```
\documentclass
```

Nuestra primera línea. Define la naturaleza de nuestro documento. Ejemplo:

```
\documentclass[12pt, a4paper, ...]{book}
```

Argumentos

- article
- book
- letter
- beamer
- :

Opciones

- Tamaño letra: 10pt
- Orientación: landscape
- Columnas: twocolumn
- Centrado: twoside
- : draft, openright...

Importación de herramientas, `\usepackage{}`

En \LaTeX se expande la funcionalidad mediante paquetes, algunos son muy necesarios. Esta sección debería ir justo debajo del `documentclass`

```
\usepackage{geometry} % Ajusta geometrías
\usepackage[spanish]{babel} % Formato en castellano
"{graphicx} % Imágenes, pdfs, etc
"{hyperref} % Referencias como tienen que ser
"[utf8]{inputenc} % Tildes y otros caracteres
"{amsmath, amssymb} % Escritura científica
```

Ver también: `makeidx` (índices avanzados), `fancyhdr` (cabeceras y pie de página), `multicol` (columnas personalizadas), `booktabs` (para tablas preciosas)

Datos previos al documento escrito, preámbulo

Como \LaTeX hará un buen número de cosas automatizadas, le damos unos datos generales en el preámbulo para que el los trate como deba.

Información del autor y texto

```
\author{Fernando ... \and Miguel \thanks{...}}  
\title{Título}  
\date{\today} % 0 en blanco si no se quiere
```

Secuencias de diseño o configuración

Si estuviéramos usando fancyhrd, makeidx o similares tendríamos que escribir en el preámbulo su diseño o configuración.

Comienzo: `documentclass`

Paquetes: `usepackage`

Preámbulo: configuraciones generales

Seccionamiento del texto

En la clase book, se tienen principalmente tres niveles

```
\chapter[short title]{title}
```

```
\section[short title]{text}
```

```
\subsection[short title]{text}
```

```
-\subsubsection[short title]{title}-
```

[short title] es lo que aparecería en el índice y en el encabezado.

Si no se quiere numerado ni en el índice: `\section*{title}`

Para escribir párrafos, dejar una línea en blanco entre ellos.

Para romper una línea usar `\\`

Estilos de texto

C: control, S: shift

Los más comunes y recomendados

Negrita/Boldface `\textbf{text}` **text**. En IDE C + b

Cursiva/Énfasis `\emph{text}` *text*. En IDE C + S + e

Subrayado `\underline{text}` text.

SmallCaps `\textsc{text}` TEXT. En IDE C + S + c

Typewriter `\texttt{text}` text. En IDE C + S + t

Medidas y espaciados. No los deberíais necesitar

- `\hfill` rellena espacio horizontal.
- `\vfill` ídem, pero en vertical.
- `\hspace{text}` espaciado horizontal (usar em como medida). Tienen versiones forzadas.
- `\vspace{text}` ídem pero en vertical. Ambos permiten valores negativos.
- `\hrulefill`

Cont. medidas “programáticas”

- `\textwidth` ancho del texto disponible (permite operaciones matemáticas). `\columnwidth` es el ancho de la columna.
- `\textheight` altura de la zona de texto.
- `\linewidth` como `\textwidth` pero relativo al entorno de trabajo

Estas son muy útiles para su uso con figuras o en tablas

Bloques (*boxes*)

Hay una buena ristra. El más importante, que puede que necesitéis, es `\mbox{text}`. Este forma un bloque único inseparable (útil para, por ejemplo, nombres propios o números).

Notas a pie de página

`\footnote{text}`. Las notas a pie de página van integradas en el texto y su formato es automático. Por ejemplo¹.

Por ejemplo `\footnote{Damos una aclaración}`.

¹Damos una aclaración

Tablas i

Entorno tabular/array básico

Esto es una introducción básica, pero suficiente, cubrirá vuestras necesidades. El IDE tiene una herramienta para hacer tablas *a la* Excel.

```
\begin{tabular}[opciones]{alineacion}
```

contenido

```
\end{tabular}
```

p, m, b sirven para hacer párrafos (top, middle, bottom)

Ejemplo:

11	12	13
hola	hola	hola
adiós querida	adiós	Sayonara Baby

Tablas ii

Código anterior (el espaciado lo da el editor)

```
\begin{tabular}{l|l|c|r}  
11          & 12      &          13 \\  
\hline \hline  
hola          & hola    &          hola \\  
\hline  
adiós querida & adiós   & Sayonara & Baby  
\end{tabular}
```

El & es bien importante, es el símbolo de separación y alineación.

Nota: ver booktabs, (ejemplo)

Items, enumeraciones y descripciones/listas i

Items

- Para `\item` automático en
IDE C + S + i

Newww Ejemplo bastante largo
para que se vean las
diferencias

- Otro item

```
\begin{itemize}  
\item Para ...  
\item[Newww] E...  
\item Otro item  
\end{itemize}
```

Enumeraciones

1. Ejemplo

```
\begin{enumerate}
```

```
\item Para ...
```

```
\item Cont.
```

2. Cont.

```
\begin{enumerate}
```

2.1 Anidados

```
\item Anidados
```

```
\end{enumerate}
```

```
\end{enumerate}
```

Descripciones/listas

label muy largo

Ejemplo de
texto un
tanto largo
para que se
vean las
diferencias

```
\begin{description}  
\item[label] Ejem...  
\item[Nombre] Cont.  
\end{description}
```

Nombre muy largo

Descripción
del texto

Imágenes u otros elementos gráficos (pdfs)

La imagen al inicio de la presentación

```
\begin{figure}[h] % Opciones h, t, b, c
\centering
\includegraphics[width=0.75\linewidth]{Donald...}
\caption{Donald Ervin Knuth. Creador de \TeX}
\label{fig:donald-knuth-stanford-computer-science}
\end{figure}
```

`includegraphics` nos da opciones para el control de la altura, ancho y escala. Sirve para un buen número de formatos, incluido `.pdf`. `\caption[short title]{text}` es el texto que aparece debajo de la imagen y en la tof. **Usad el *wizard* que trae el IDE.**

Programas y fragmentos de código

Se usa `\usepackage{listings}`. Es personalizable hasta el final, desde color del fondo, esquemas de color para el código, reconoce docenas de lenguajes, etc. **Por favor**, miraros la documentación y copiad ejemplos.

```
\begin{lstlisting}[language=Pascal]
for i:=maxint to 0 do
begin
{ do nothing }
end;
Write('Case insensitive ');
Write('Pascal keywords.');
```

```
for i:=maxint to 0 do
begin
{ do nothing }
end;
Write('Case insensitive ');
Write('Pascal keywords. ');
\end{lstlisting}
```


Labels, etiquetado

`\label{key}` nos permite etiquetar lo que deseemos referenciar (anterior o posteriormente). Ejemplos:

- `\label{eq:maxwell}` ecuación de Maxwell
- `\label{fig:imagen}` alguna imagen
- `\label{tab:tabla}` alguna tabla
- `\label{sec:appendixa}` apéndice A
- Etcétera

Usadla/Indicádla a continuación de lo que queráis citar, dentro del entorno.

Referencias, citas

`\autoref{key}` nos generará la referencia de manera automática. Tendrá en cuenta el entorno usado, sección, etc. Ver Figura 1.

La forma tradicional sería con `\ref{text}`:
`\ref{fig:donald}`. Lo que daría solo "1", por lo que tendríamos que indicar qué estamos referenciando.

Bibliografía, programas externos i

No son necesarios para trabajar en \LaTeX como veremos. Pero son muy útiles para el manejo de bibliografías grandes y complicadas. Además de traer muchas herramientas de búsqueda y formato de gran ayuda.

Recordad que hay servicios bibliográficos, como **Google Scholar**, donde podemos buscar la información de las referencias. Además, todos estos servicios sacan formato Bib \TeX .

Nota, hay varios procesadores internos de bibliografía, nosotros usaremos el más flexible y sencillo: Bib \LaTeX .

Comentar configuración en \TeX Studio

Programas

Zotero Multiplataforma y exporta tanto a \LaTeX como a Word. Todas las herramientas necesarias están incluidas excepto un motor de búsqueda con texto (puede ISBNs, DOIs, etc), aunque tiene integración con Firefox y Safari.

JavRef Multiplataforma y también exporta a Word. Completo y avanzado. También tiene integración con Firefox.

KBibTeX Solo Linux. Muy simple pero sencillo de usar y flexible, además de traer varios motores de búsqueda.

Bibliografía en L^AT_EX, uso en el documento i

Citas

Para citar una obra simplemente se hace `\cite{bibid}` donde se quiera la referencia. El bibid es el identificador de nuestra referencia.

Inclusión en el documento y estilos

```
\section{Bibliografía}
\bibliographystyle{style} % plain, abbrev, alpha...
\bibliography{bib1,bib2,bib...} % Añadimos
% archivo(s), sin espacios ni extensión
\end{document} % si queremos terminar
```

Estructura del archivo .bib

Todas las entradas empiezan con una @ y su identificador (article, journal, book, etc); esto sirve para darles formato. A continuación se abren llaves.

Entre las llaves se escribirá la información separada por comas. Lo más importante es la primera palabra que pondremos, esa será la identificación para el comando `\cite{bibid}`. A continuación rellenaremos tantos campos como necesitemos año, título, autor, url, editor, etc, tal y como está indicado en el ejemplo.

El archivo .bib

El archivo .bib, que se recomienda que esté junto con el documento .tex principal, es nuestra *base de datos* con las referencias. Un ejemplo sería:

```
@BOOK{White201501,  
title={Fluid Mechanics},  
author={Frank M. White},  
publisher={McGraw-Hill Education},  
year={2015},  
edition={8},  
isbn={9780073398273},  
totalpages={864},  
timestamp={2018.10.29},  
}
```

Escritura científica

Lógica de la escritura científica en L^AT_EX

Desarrollo

- L^AT_EX se creó para permitir una fácil y rápida creación de textos, aunque parezca poco intuitivo al principio.

Regla de la mano derecha: si algo es muy utilizado y básico en el mundo de las matemáticas y de las ciencias, está acortado, simplificado. El resto son los nombres descriptivos.

Ejemplo: la integral cerrada se usa mucho → está simplificada:

$$\oint = \text{\texttt{\$}\backslash oint\text{\texttt{\$}}}$$

La doble integral cerrada sigue su desarrollo, pero no viene en Amsmath: `\oiint`. La flecha a la derecha no es un símbolo matemático muy querido → no se abrevia `\rightarrow`

Ejemplos de lógica

Expresivos

- ϕ `\phi`
- φ `\varphi`
- Φ `\Phi`
- \downarrow `\downarrow`
- \Downarrow `\Downarrow`

Abreviados

- \prod `\prod`
- \neq `\neq`
- \in `\in`
- \ni `\ni`

Ejercicio, adivinad los símbolos

- \sim
- Π
- \pm
- \mp
- \rightarrow

Ejemplos de lógica

Expresivos

- ϕ `\phi`
- φ `\varphi`
- Φ `\Phi`
- \downarrow `\downarrow`
- \Downarrow `\Downarrow`

Abreviados

- \prod `\prod`
- \neq `\neq`
- \in `\in`
- \ni `\ni`

Ejercicio, adivinad los símbolos

- \sim
- Π
- \pm
- \mp
- \rightarrow

Respuestas

- `\sim`
- `\Pi`
- `\pm`
- `\mp`
- `\rightarrow`

Un par de fórmulas para que os familiaricéis

Ecuación de Bernoulli

$$\left(\frac{p}{\rho g} + \frac{\alpha}{2g} V^2 + z \right)_{ent} = \left(\frac{p}{\rho g} + \frac{\alpha}{2g} V^2 + z \right)_{sal} + h_{tur} + h_{fr} - h_{bom}$$

$$\left[\left(\frac{p}{\rho g} + \frac{\alpha}{2g} V^2 + z \right)_{ent} \right. \\ \left. = \left(\frac{p}{\rho g} + \frac{\alpha}{2g} V^2 + z \right)_{sal} \right. \\ \left. + h_{tur} + h_{fr} - h_{bom} \right]$$

Hermosas mates

$$\zeta(s) = \sum_{n=1}^{\infty} \frac{1}{n^s} \quad \Gamma(z) = \int_0^{\infty} t^{z-1} e^{-t} dt$$

$$\left[\zeta(s) = \sum\limits_{n=1}^{\infty} \frac{1}{n^s} \right. \\ \left. \Gamma(z) = \int_{0}^{\infty} t^{z-1} e^{-t} dt \right]$$

Escritura científica, dos formas

En \LaTeX existen dos formas de escribir fórmulas matemáticas. La razón es simple, estilo y formato.

Inline

Traducido al español: en línea. Se usa para meter símbolos y fórmulas **dentro del texto**. Este modo respetará el formato que posea el texto. Se accede con el signo del dólar. Ejemplo: $\frac{2^2}{4} = 1$
`\frac{2^2}{4} = 1`, `\(...\)` también se permite.

Escritura científica, dos formas

En \LaTeX existen dos formas de escribir fórmulas matemáticas. La razón es simple, estilo y formato.

Inline

Traducido al español: en línea. Se usa para meter símbolos y fórmulas **dentro del texto**. Este modo respetará el formato que posea el texto. Se accede con el signo del dólar. Ejemplo: $\frac{2^2}{4} = 1$
`\frac{2^2}{4} = 1`, `\(...\)` también se permite.

Display (En TF*, usad equation o similar)

Se utiliza para la escritura a parte de la expresión matemática. Genera un espacio nuevo para la fórmula. Se accede con `\[...\]`.

$$\frac{2^{23}}{4} = 2^{21}$$

`\[\frac{2^{\{23\}}{4} = 2^{\{21\}}\]`.

Matrices

Arrays/Tablas

Funciona igual que el entorno tabular pero se usa dentro del entorno de escritura matemática. Juntando esto con el `\left(...\right)` o cualquier otro símbolo podemos hacer matrices.

Matrices

Arrays/Tablas

Funciona igual que el entorno tabular pero se usa dentro del entorno de escritura matemática. Juntando esto con el `\left(...\right)` o cualquier otro símbolo podemos hacer matrices.

Pero \LaTeX es bien eficiente

Las matrices son una herramienta muy usada, por lo que hay una forma sencilla. `\begin{*matrix}...\end{*matrix}`. No requiere de opciones de alineación. *****: significa el tipo de puntuación a usar: **p**: paréntesis; **v**: vertical; **b**: corchetes; **B**: llaves. **Ejemplo:**

$$\begin{pmatrix} 1 & 2 \\ 3 & 4 \end{pmatrix} \rightarrow \begin{vmatrix} 1 & 2 \\ 3 & 4 \end{vmatrix} = -2 \Rightarrow \begin{bmatrix} 1 & 2 \\ 3 & 4 \end{bmatrix}^2 = \begin{Bmatrix} 7 & 10 \\ 15 & 22 \end{Bmatrix}$$

El ejemplo de las matrices

```
\[ \begin{pmatrix}
1 & 2 \\
3 & 4
\end{pmatrix} \rightarrow
\begin{vmatrix}
1 & 2 \\
3 & 4
\end{vmatrix} = -2 \rightarrow
\begin{bmatrix}
1 & 2 \\
3 & 4
\end{bmatrix}^2 =
\begin{Bmatrix}
7 & 10 \\
15 & 22
\end{Bmatrix} \]
```

Entorno equation(*)

La base de la estructuración

`\begin{equation}...\end{equation}` tiene los mismos efectos que `\[...\]` sin embargo, para ecuaciones formales, importantes y largas, es preferido. Da mayor claridad al código, otros entornos se pueden usar dentro suyo y permite la **referencia y numeración**. Para evitar la numeración se le pone un `*` al final de su declaración.

Entorno equation(*)

La base de la estructuración

`\begin{equation}...\end{equation}` tiene los mismos efectos que `\[...\]` sin embargo, para ecuaciones formales, importantes y largas, es preferido. Da mayor claridad al código, otros entornos se pueden usar dentro suyo y permite la **referencia y numeración**. Para evitar la numeración se le pone un `*` al final de su declaración.

Ejemplo

$$f(x) = (x + a)(x + b) \tag{1}$$

Como se puede ver en la Ecuación 1, \LaTeX es bien sencillo.

```
\begin{equation} \label{ec:ejemplo1}
```

```
f(x)=(x+a)(x+b)
```

```
\end{equation}
```

Ecuaciones alineadas

Existen varios entornos, todos con sus peculiaridades, pero prácticamente idénticos. Los más conocidos son `split`, `aligned` & `align`. `aligned` subenumera cada salto de línea por defecto.

$$\begin{aligned} a &= c + d \\ &= e + f \end{aligned}$$

(2)

```
\begin{equation}
\begin{split}
a &= c + d \\
&= e + f
\end{split}
\end{equation}
```

Paréntesis, corchetes y demás puntuaciones

L^AT_EX automáticamente ajusta el tamaño de los símbolos de puntuación, **pero hay que indicárselo**. El comando es `\left*...\right*`, con `*` siendo el símbolo.

Ejemplo

$$3 = \left(\frac{3}{1^{100}} \right)$$

```
\begin{equation*}
3 = \left(\frac{3}{1^{100}}\right)
\end{equation*}
```

Si solo se quiere usar uno (el de apertura o cierre), el que no se desee se indicará con `\left.` o `\right.`.

Más flexibilidad en la escritura

Anotaciones sobre fórmulas

- `\overset{}{} y \underset{}{} Nos permiten poner unos símbolos encima de otros. Ejemplo:`

$$A \overset{!}{=} B; A \underset{!}{=} B \rightarrow$$

$$\$A \overset{!}{=} B; A \stackrel{!}{=} B\$.$$

Más flexibilidad en la escritura

Anotaciones sobre fórmulas

- `\overset{}{} y \underset{}{} Nos permiten poner unos símbolos encima de otros. Ejemplo:`

$$A \overset{!}{=} B; A \underset{!}{=} B \rightarrow$$

$$\$A \overset{!}{=} B; A \stackrel{!}{=} B\$.$$

- `\overbrace{} y \underbrace{} Nos permiten coger`

trozos de ecuaciones. Ejemplo: $z = \overbrace{\underbrace{x}_{\text{real}} + \underbrace{iy}_{\text{imaginario}}}^{\text{número complejo}} \rightarrow$

$$\$z = \overbrace{\underbrace{x}_{\text{real}} + \underbrace{iy}_{\text{imaginario}}}^{\text{número complejo}}\$$$

Tipografía

Para introducir texto normal se usa `text`

- `\mathbb{}`: SOLO MAYÚSCULAS

- `\mathbf{}` **1234 text**

- `\mathfrak{}`

1234 text $\Re, \Im, \mathfrak{L}, \mathfrak{F}, \mathfrak{N}$

- `\mathrm{}` 1234 text

- `\mathcal{}` *SOLO MAYÚSCULAS*

- `\mathrm{}` 1234 text

Resumen y otros recursos

1. Diseño del documento
2. Configuración del documento
3. Estructuración del texto
4. Herramientas para el trabajo de texto
5. Entornos útiles
6. Referencias y bibliografía
7. Escritura científica
8. Recursos extra

Temas no tratados

Recursos que no se han mencionado pero que son de gran ayuda (dentro o fuera de $\text{TF}(G,M)$)

1. **MACROS**: permiten hacer comandos especializados. Muy útil cuando se tienen estructuras repetitivas.
2. **Beamer**: entorno de presentaciones más usado, por ejemplo, esta misma.
3. **Lua \LaTeX** , **Xe \LaTeX** : procesadores más modernos y con más herramientas, automatización y programables.
4. Preprocesadores de bibliografía como **biber**.
5. Distintas tipografías (hay algunas preciosas).
6. Más paquetes que os sean de ayuda, pero tenéis los suficientes.
7. Aprended bien el editor (mi propuesta **\TeX Studio**).

¡FIN Y GRACIAS!

Contacto fernando.oleo@alu.comillas.edu

Plantillas La de la universidad o la que tendréis en el mismo link que esta presentación

Thanks DANIEL ANDRÉS ARCONES,
AURELIO GARCÍA CERRADA
y la gente de ICAI 3D