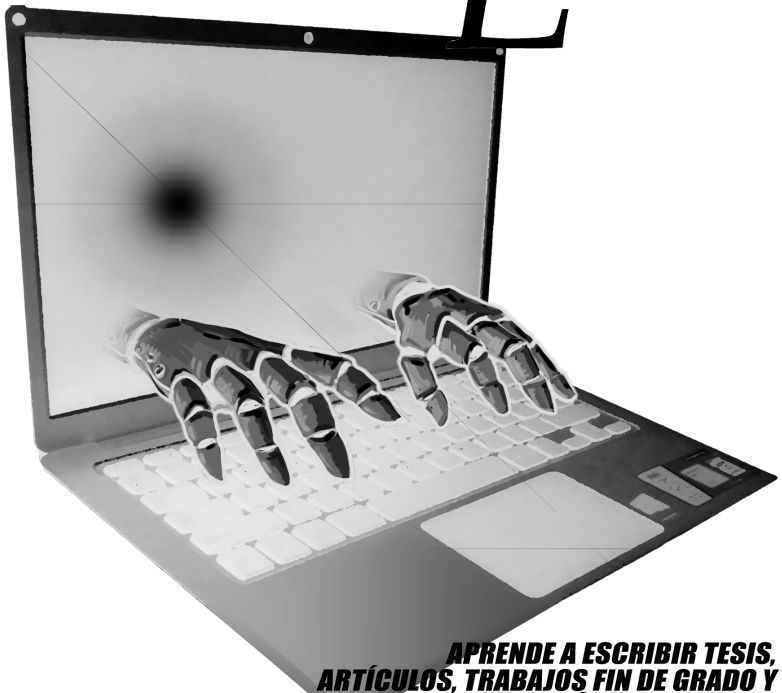


# LA BIBLIA DE L<sup>A</sup>T<sub>E</sub>X



**APRENDE A ESCRIBIR TESIS,  
ARTÍCULOS, TRABAJOS FIN DE GRADO Y  
PRESENTACIONES CON TERMINACIÓN PROFESIONAL.**

**DAVID SANTO ORCERO**

*david@peritoeninformatica.pro*  
*<http://www.ingenieroeninformatica.pro/>*

© David Santo Orcero  
Todos los derechos reservados  
Febrero del 2019

---

*Santo Orcero, David*

**La Biblia de  $\text{\LaTeX}$  2 $\epsilon$**

Ed. Coronado — Impreso por Amazon

Febrero del 2019

**Palabras clave:** latex, latexe, autoedición

**ISBN-10:** 1-7954-0971-1

**ISBN-13:** 978-1795409711

**Número de depósito legal:** MA 319-2019

**BISAC:** • COM022000 COMPUTERS / Desktop Applications / Desktop Publishing • COM058000 COMPUTERS / Desktop Applications / Word Processing • COM085000 COMPUTERS / Documentation & Technical Writing • COM078000 COMPUTERS / Desktop Applications / Presentation Software

**IBIC:** • UGD (Autoedición) • UFD (Procesadores de texto) • UFG (Software para presentaciones gráficas) • 2ADS (En castellano) • 4KMT (Adquisición de competencias TIC)

---

$\text{\LaTeX}$  es un sistema de preparación de documentos desarrollado originariamente por Leslie Lamport, basado en el  $\text{\TeX}$  de Donald E. Knuth, y bajo licencia libre. Debian es una marca registrada de Software in Public Interest, Inc. Los respectivos programas analizados en este libro pueden ser marcas de sus respectivos dueños.

---

Portada diseñada por Daniel Santo Orcero; puedes ver su obra en <http://www.danielsanto.es/>.

---

Contacte con el autor en la dirección [david@peritoeninformatica.pro](mailto:david@peritoeninformatica.pro) si desea descuentos por compras en volumen, o para cursos, ponencias o presentaciones sobre la temática del curso o para informes periciales.

# Índice general

<b>Índice general</b>	<b>I</b>
<b>1 Introducción al L<sup>A</sup>T<sub>E</sub>X 2<sub>ε</sub></b>	<b>1</b>
1.1 La filosofía de la maquetación . . . . .	4
1.2 Origen de L <sup>A</sup> T <sub>E</sub> X 2 <sub>ε</sub> . . . . .	9
1.3 Tópicos sobre LaTeX . . . . .	13
1.4 Estructura de un texto en L <sup>A</sup> T <sub>E</sub> X 2 <sub>ε</sub> . . . . .	18
<b>2 Instalando L<sup>A</sup>T<sub>E</sub>X 2<sub>ε</sub></b>	<b>21</b>
2.1 T <sub>E</sub> Xlive vs. t <sub>E</sub> X . . . . .	21
2.2 Instalando L <sup>A</sup> T <sub>E</sub> X 2 <sub>ε</sub> en Linux . . . . .	22
2.3 Instalando L <sup>A</sup> T <sub>E</sub> X 2 <sub>ε</sub> en Mac OS X . . . . .	23
2.4 Instalando L <sup>A</sup> T <sub>E</sub> X 2 <sub>ε</sub> en Windows . . . . .	25
2.4.1 L <sup>A</sup> T <sub>E</sub> X 2 <sub>ε</sub> en Windows con MiK <sub>T</sub> E <sub>X</sub> . . . . .	25
2.4.2 Instalando Texmaker . . . . .	31
2.4.3 Qué puede salir mal . . . . .	32
2.5 Después de la instalación... . . . .	34

<b>3</b>	<b>Conceptos básicos de L<sup>A</sup>T<sub>E</sub>X 2<sub>ε</sub></b>	<b>37</b>
3.1	Clases en L <sup>A</sup> T <sub>E</sub> X 2 <sub>ε</sub> . . . . .	40
3.2	El texto del documento . . . . .	41
3.2.1	El texto crudo . . . . .	41
3.2.2	Los comandos . . . . .	42
3.2.3	Los caracteres de control . . . . .	45
3.2.4	Espacios y retornos de carro . . . . .	47
3.3	La codificación de caracteres . . . . .	52
3.3.1	Formas de codificar caracteres . . . . .	52
3.3.2	Codificación en UTF-8: la que vamos a usar habitualmente . . . . .	54
3.3.3	Otras codificaciones, ya arcaicas . . . . .	55
3.3.4	La composición tipográfica . . . . .	57
3.3.5	Caracteres vía comando . . . . .	64
3.3.6	El símbolo del euro € . . . . .	65
3.3.7	Signos especiales usados en español . . . . .	66
3.4	Babel y el soporte multilíngue de L <sup>A</sup> T <sub>E</sub> X 2 <sub>ε</sub> . . . . .	67
3.4.1	Qué es Babel . . . . .	67
3.4.2	Manos a la obra con Babel . . . . .	68
3.5	Qué vendrá ahora . . . . .	71
<b>4</b>	<b>Parámetros de clases, entornos, tablas, y otros elementos básicos</b>	<b>73</b>
4.1	Clases de documentos . . . . .	73
4.1.1	Clases más importantes en L <sup>A</sup> T <sub>E</sub> X 2 <sub>ε</sub> . . . . .	74
4.1.2	Un arcaísmo: estilos en L <sup>A</sup> T <sub>E</sub> X . . . . .	76

4.2	La clase <code>book</code> , primera aproximación . . . . .	77
4.3	Opciones de clases . . . . .	78
4.3.1	Opciones de número de caras . . . . .	78
4.3.2	Opciones de número de columnas . . . . .	79
4.3.3	Opciones de tamaño base de los caracteres . . . . .	79
4.3.4	Opciones de tamaño de página . . . . .	80
4.3.5	Formato apaisado . . . . .	81
4.3.6	Colocación de la primera página del capítulo . . . . .	83
4.4	Entornos . . . . .	83
4.4.1	Qué es un entorno . . . . .	83
4.4.2	Entornos más comunes . . . . .	84
4.4.3	Ejemplos con entornos . . . . .	86
4.4.4	Entornos de tamaño . . . . .	92
4.4.5	El entorno matemático . . . . .	95
4.4.6	Comandos y entornos curiosos . . . . .	97
4.5	Listados y enumeraciones . . . . .	97
4.5.1	Listados en L <sup>A</sup> T <sub>E</sub> X 2 <sub>ε</sub> . . . . .	98
4.5.2	Enumeraciones en L <sup>A</sup> T <sub>E</sub> X 2 <sub>ε</sub> . . . . .	102
4.5.3	Listas descriptivas en L <sup>A</sup> T <sub>E</sub> X 2 <sub>ε</sub> . . . . .	105
4.6	Tablas . . . . .	107
4.6.1	<code>tabular</code> y las tablas simples . . . . .	107
4.6.2	El contenido de la tabla . . . . .	108
4.6.3	La apertura del entorno <code>tabular</code> . . . . .	109
4.6.4	Dibujando líneas en <code>tabular</code> . . . . .	110
4.7	Unidades de medida de longitud en L <sup>A</sup> T <sub>E</sub> X 2 <sub>ε</sub> . . . . .	113

4.8	Incluyendo imágenes en L <sup>A</sup> T <sub>E</sub> X 2 <sub>ε</sub> . . . . .	115
4.8.1	Planteamiento previo: imágenes en L <sup>A</sup> T <sub>E</sub> X 2 <sub>ε</sub> . . . . .	115
4.8.2	Incluyendo el paquete <code>graphicx</code> . . . . .	116
4.8.3	El comando <code>includegraphics</code> . . . . .	116
4.8.4	Escalando la imagen . . . . .	118
4.8.5	Rotando la imagen . . . . .	121
4.8.6	Recortando la imagen . . . . .	123
4.8.7	Ahorrando tonner o tinta . . . . .	125
5	<b>Libros, trabajos fin de grado, tesis doctorales y artículos</b>	<b>127</b>
5.1	Secciones en L <sup>A</sup> T <sub>E</sub> X 2 <sub>ε</sub> . . . . .	127
5.1.1	Estructurando el texto en partes . . . . .	127
5.1.2	La numeración de las secciones . . . . .	132
5.1.3	Definiendo la profundidad de la numeración . . . . .	133
5.2	Haciendo un pie de página . . . . .	135
5.3	Revisitando la clase <code>book</code> . . . . .	135
5.3.1	Organización general de un libro . . . . .	136
5.3.2	Partes de un libro . . . . .	137
5.3.3	Inclusión de apéndices . . . . .	139
5.4	La clase <code>article</code> . . . . .	139
5.4.1	Escribiendo el título . . . . .	140
5.4.2	El resumen . . . . .	141
5.4.3	Opciones de configuración del resumen . . . . .	142
5.5	Organizando el proyecto en L <sup>A</sup> T <sub>E</sub> X 2 <sub>ε</sub> . . . . .	143
5.5.1	Organizando el documento en ficheros . . . . .	143

5.5.2	Incluyendo solo algunos ficheros . . . . .	145
5.5.3	Utilizando artículos como capítulos de un libro . . . . .	145
5.6	Índices, referencias y citas . . . . .	146
5.6.1	Índices en L <sup>A</sup> T <sub>E</sub> X 2 <sub>ε</sub> . . . . .	146
5.6.2	Creando el índice . . . . .	147
5.6.3	Las referencias cruzadas . . . . .	150
5.6.4	Bibliografías sin BIB <sub>T</sub> E <sub>X</sub> – thebibliography . . . . .	152
<b>6</b>	<b>Presentaciones en L<sup>A</sup>T<sub>E</sub>X 2<sub>ε</sub></b>	<b>157</b>
6.1	El paquete beamer . . . . .	157
6.2	Comandos de beamer . . . . .	168
6.3	Creando una tabla de contenidos . . . . .	173
6.4	overlays en beamer . . . . .	175
6.5	Cambiando el flujo de la presentación . . . . .	181
6.6	Cambiando la estética de nuestra presentación . . . . .	184
6.7	Entornos frágiles y presentaciones . . . . .	187
6.8	Las anotaciones . . . . .	188
6.9	Imprimiendo presentaciones . . . . .	201
6.10	Montando presentaciones . . . . .	203
<b>7</b>	<b>Cómo escribir una carta en L<sup>A</sup>T<sub>E</sub>X 2<sub>ε</sub></b>	<b>205</b>
7.1	La clase letter . . . . .	206
7.2	Los comandos del preámbulo . . . . .	208
7.3	Los comandos del cuerpo del texto del entorno letter . . . . .	211

<b>8</b>	<b>Bibliografía con BIB<sub>T</sub>E<sub>X</sub></b>	<b>215</b>
8.1	Filosofía de BIB <sub>T</sub> E <sub>X</sub> . . . . .	216
8.2	Ejecutando BIB <sub>T</sub> E <sub>X</sub> . . . . .	217
8.3	Citando bibliografía con BIB <sub>T</sub> E <sub>X</sub> . . . . .	219
8.4	Entradas bibliográficas bajo L <sup>A</sup> T <sub>E</sub> X 2 <sub>ε</sub> . . . . .	221
8.4.1	El archivo de entradas bibliográficas . . . . .	221
8.4.2	Extrayendo las entradas BIB <sub>T</sub> E <sub>X</sub> de la Web . . . .	223
8.4.3	Los campos de las referencias bibliográficas . . . .	226
8.4.3.1	Campos obligatorios . . . . .	226
8.4.3.2	Campos alternos . . . . .	227
8.4.3.3	Campos optativos . . . . .	227
8.4.4	Los tipos de referencia en BIB <sub>T</sub> E <sub>X</sub> . . . . .	228
8.4.4.1	Las referencias de tipo <code>article</code> . . . . .	228
8.4.4.2	Las referencias de tipo <code>Book</code> . . . . .	230
8.4.4.3	Las referencias de tipo <code>Booklet</code> . . . . .	232
8.4.4.4	Las referencias de tipo <code>TechReport</code> . . . . .	233
8.4.4.5	Las referencias de tipo <code>Manual</code> . . . . .	234
8.4.4.6	Las referencias de tipo <code>PhdThesis</code> . . . . .	235
8.4.4.7	Las referencias de tipo <code>MastersThesis</code> . . . . .	237
8.4.4.8	Las referencias de tipo <code>Proceedings</code> . . . . .	238
8.4.4.9	Las referencias de tipo <code>InProceedings</code> . . . . .	239
8.4.4.10	Las referencias de tipo <code>InCollection</code> . . . . .	242
8.4.4.11	Las referencias de tipo <code>InBook</code> . . . . .	244
8.4.4.12	Las referencias de tipo <code>Unpublished</code> . . . . .	247
8.4.4.13	Las referencias de tipo <code>Misc</code> . . . . .	248



8.4.5	Rellenando los campos . . . . .	249
8.4.5.1	Poniendo mayúsculas dónde queramos	251
8.4.5.2	Autores con dos apellidos . . . . .	253
8.4.5.3	Autores sin cambio de orden en el nombre	253
8.4.5.4	Bibliografías con caracteres generados por composición . . . . .	254
8.5	Estilos en BibT <sub>E</sub> X . . . . .	255
8.5.1	Los tipos de bibliografía . . . . .	257
8.6	Ejemplo de selección de estilo de bibliografía . . . . .	258
8.6.1	Texto base del documento de ejemplo . . . . .	259
8.6.2	Archivo de bibliografía ejemplo . . . . .	260
8.6.3	Ejemplo de la bibliografía tipo <code>abstract</code> . . . . .	264
8.6.4	Ejemplo de la bibliografía tipo <code>aer</code> . . . . .	265
8.6.5	Ejemplo de la bibliografía tipo <code>aertt</code> . . . . .	266
8.6.6	Ejemplo de la bibliografía tipo <code>alphadin</code> . . . . .	267
8.6.7	Ejemplo de la bibliografía tipo <code>annotate</code> . . . . .	268
8.6.8	Ejemplo de la bibliografía tipo <code>aomplain</code> . . . . .	269
8.6.9	Ejemplo de la bibliografía tipo <code>apsr</code> . . . . .	270
8.6.10	Ejemplo de la bibliografía tipo <code>IEEEtranM</code> . . . . .	271
8.6.11	Ejemplo de la bibliografía tipo <code>IEEEtranN</code> . . . . .	272
8.6.12	Ejemplo de la bibliografía tipo <code>IEEEtran</code> . . . . .	273
8.6.13	Ejemplo de la bibliografía tipo <code>IEEEtranSA</code> . . . . .	274
8.6.14	Ejemplo de la bibliografía tipo <code>IEEEtranSN</code> . . . . .	275
8.6.15	Ejemplo de la bibliografía tipo <code>IEEEtranS</code> . . . . .	276
8.6.16	Ejemplo de la bibliografía tipo <code>imac</code> . . . . .	277

8.6.17	Ejemplo de la bibliografía tipo <code>phrmp</code> . . . . .	278
8.6.18	Ejemplo de la bibliografía tipo <code>plain-letters</code> . . . . .	279
8.6.19	Ejemplo de la bibliografía tipo <code>savetrees</code> . . . . .	280
8.6.20	Ejemplo de la bibliografía tipo <code>siam-letters</code> . . . . .	281
8.6.21	Ejemplo de la bibliografía tipo <code>siam</code> . . . . .	282
8.6.22	Otros ejemplos de bibliografía . . . . .	283
<b>9</b>	<b>Edición de fórmulas matemáticas en <math>\text{\LaTeX} 2_{\epsilon}</math> con <math>\mathcal{A}\mathcal{M}\mathcal{S}\text{-}\text{\LaTeX}</math></b>	<b>285</b>
9.1	Antes de comenzar . . . . .	286
9.2	$\mathcal{A}\mathcal{M}\mathcal{S}\text{-}\text{\LaTeX}$ . . . . .	287
9.3	Los entornos matemáticos . . . . .	289
9.3.1	Los entornos matemáticos más simples . . . . .	289
9.3.2	Entornos matemáticos más complicados . . . . .	293
9.4	Edición de fórmulas . . . . .	298
9.4.1	Conceptos básicos de edición de fórmulas . . . . .	298
9.4.2	Símbolos matemáticos en $\text{\LaTeX} 2_{\epsilon}$ . . . . .	301
9.4.2.1	Símbolos específicos de matemáticas . . . . .	301
9.4.2.1.1	Letras griegas . . . . .	302
9.4.2.1.2	Operadores en horizontal . . . . .	303
9.4.2.1.3	Operadores relacionales en horizontal . . . . .	305
9.4.2.1.4	Símbolos de puntuación . . . . .	308
9.4.2.1.5	Funciones matemáticas frecuentes . . . . .	308
9.4.2.1.6	Flechas . . . . .	309
9.4.2.1.7	Operadores de agregación . . . . .	310

9.4.2.1.8	Delimitadores . . . . .	311
9.4.2.1.9	Otros símbolos disponibles . .	312
9.4.2.2	Los diacríticos en los entornos matemáticos . . . . .	313
9.4.2.2.1	Diacríticos en caracteres . . . .	313
9.4.2.2.2	Diacríticos de fórmulas . . . . .	315
9.4.2.3	Los puntos suspensivos . . . . .	316
9.4.2.3.1	Confiando en L <sup>A</sup> T <sub>E</sub> X 2 <sub>ε</sub> dónde van	316
9.4.2.3.2	Especificando a L <sup>A</sup> T <sub>E</sub> X 2 <sub>ε</sub> dónde van . . . . .	317
9.4.2.3.3	Puntos suspensivos de integrales	318
9.4.2.3.4	Puntos suspensivos en matrices, determinantes, y sistemas de ecuaciones . . . . .	319
9.4.2.4	Poniendo texto dentro de una fórmula .	320
9.4.2.5	Fuentes matemáticas . . . . .	322
9.4.2.5.1	Ejemplos de fuentes en entorno matemático (fuentes completas)	322
9.4.2.5.2	Ejemplos de fuentes en entorno matemático (solo mayúsculas) .	324
9.4.2.5.3	Uso de las fuentes en entorno matemático . . . . .	325
9.4.2.6	Paréntesis más grandes . . . . .	326
9.4.3	Componiendo ecuaciones . . . . .	330
9.4.3.1	Retomando la composición básica . . . .	330
9.4.3.2	Superíndices y subíndices . . . . .	332
9.4.3.3	Superíndices y subíndices en operadores matemáticos . . . . .	335

9.4.3.4	Montando geométricamente arriba o abajo	338
9.4.3.5	Montando caracteres en flechas . . . . .	341
9.4.3.6	Negando operadores . . . . .	343
9.4.3.7	Las llaves . . . . .	344
9.4.3.8	Afinando los “castillos de fracciones” . .	345
9.4.3.9	Las matrices y los determinantes . . . . .	347
9.5	Entornos matemáticos avanzados . . . . .	351
9.5.1	Ecuaciones multilínea en L <sup>A</sup> T <sub>E</sub> X 2 <sub>ε</sub> . . . . .	351
9.5.2	Entornos multiecuaciones no alineados . . . . .	359
9.5.3	Entornos matemáticos de $\mathcal{A}\mathcal{M}\mathcal{S}$ -L <sup>A</sup> T <sub>E</sub> X . . . . .	360
9.5.4	Sistemas de ecuaciones con $\mathcal{A}\mathcal{M}\mathcal{S}$ -L <sup>A</sup> T <sub>E</sub> X . . . . .	362
9.6	Poniendo las fórmulas en cajas . . . . .	364
<b>10</b>	<b>Gráficos en L<sup>A</sup>T<sub>E</sub>X 2<sub>ε</sub></b>	<b>367</b>
10.1	El entorno <code>picture</code> . . . . .	368
10.1.1	El área de dibujo . . . . .	368
10.1.2	El tamaño del dibujo . . . . .	369
10.1.3	Las directivas de posición . . . . .	373
10.1.4	Dibujando líneas . . . . .	376
10.1.5	Dibujando vectores . . . . .	379
10.1.6	Dibujando circunferencias y círculos . . . . .	380
10.1.7	Poniendo textos en gráficos . . . . .	381
10.1.8	Un ejemplo complejo de entorno <code>picture</code> . . . . .	382
10.2	Poniendo textos en cajas . . . . .	383
10.2.1	El paquete <code>shadow</code> . . . . .	383

10.2.1.1	Textos en cajas simples . . . . .	383
10.2.1.2	Textos en cajas “sombreadas” . . . . .	384
10.2.2	El paquete <code>fancybox</code> . . . . .	384
10.2.2.1	Usando <code>shadowbox</code> . . . . .	385
10.2.2.2	Usando <code>doublebox</code> . . . . .	385
10.2.2.3	Usando <code>ovalbox</code> . . . . .	386
10.2.2.4	Usando <code>Ovalbox</code> . . . . .	386
10.3	Uso avanzado del entorno <code>picture</code> . . . . .	387
10.3.1	Rectángulos con texto dentro en <code>picture</code> . . . . .	387
10.3.2	Dibujando ovals . . . . .	395
10.3.3	Poniendo texto en columnas . . . . .	400
10.3.4	El comando <code>dashbox</code> . . . . .	402
10.3.5	Las curvas Bézier . . . . .	406
10.3.6	Cambiando el ancho de las líneas . . . . .	410
10.4	Dibujando gráficas en la realidad . . . . .	413
10.5	Dibujando gráficos de barras verticales con <code>bar</code> . . . . .	414
10.5.1	Usando el paquete <code>bar</code> . . . . .	414
10.5.2	Mejorando el gráfico de barras <code>bar</code> . . . . .	416
10.5.3	Los ejes del gráfico de barras <code>bar</code> . . . . .	421
10.5.4	Poniendo leyendas al gráfico en <code>bar</code> . . . . .	430
10.5.5	Mejorando la estética del resultado en <code>bar</code> . . . . .	433
10.5.6	Cambiando el número de la barra en <code>bar</code> . . . . .	435
10.6	Dibujando gráficos de barras horizontales con <code>bchart</code> . . . . .	440
10.6.1	Usando el paquete <code>bchart</code> . . . . .	441
10.6.2	Mejorando el gráfico de barras <code>bchart</code> . . . . .	442

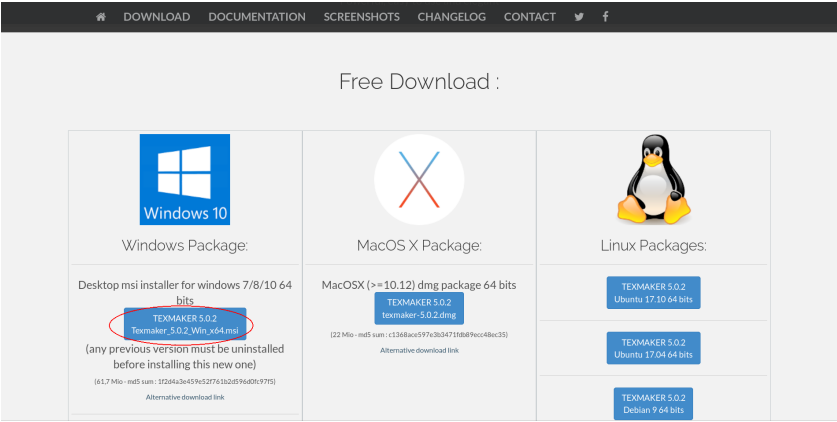
10.6.3	Los ejes del gráfico de barras <code>bchart</code> . . . . .	447
10.6.4	Colores TikZ en el gráfico de barras <code>bchart</code> . . .	453
10.7	Dibujando fórmulas con <code>pgfplots</code> . . . . .	459
10.7.1	Utilizando el paquete <code>pgfplots</code> . . . . .	459
10.7.2	Dibujando gráficas de funciones con <code>pgfplots</code> .	460
10.7.3	Dibujando gráficas a partir de ficheros con volcados de números . . . . .	465
10.7.4	Cambiando el tamaño del gráfico . . . . .	467
10.7.5	Opciones adicionales de <code>addplot</code> . . . . .	471
10.7.6	Modificando los ejes . . . . .	473
10.7.7	Dibujando leyendas de un gráfico . . . . .	491
10.7.8	Dibujando gráficas de dispersión . . . . .	508
10.7.9	Dibujando gráficas en 3D . . . . .	511
10.8	L <sup>A</sup> T <sub>E</sub> X 2 <sub>ε</sub> en química y bioquímica . . . . .	518
10.9	Si no quiero diseñar los gráficos desde L <sup>A</sup> T <sub>E</sub> X 2 <sub>ε</sub> . . . . .	521
<b>11</b>	<b>Control de formato y maquetación manual en L<sup>A</sup>T<sub>E</sub>X 2<sub>ε</sub></b>	<b>523</b>
11.1	Tomando el control del formato de nuestro documento .	525
11.1.1	Definiendo cabeceras . . . . .	525
11.1.2	El formato de página, a bajo nivel . . . . .	531
11.1.3	Control adicional de espaciado . . . . .	537
11.1.4	Control fino de espacios y saltos en una hoja . . .	547
11.1.4.1	Espaciados en modo horizontal . . . . .	547
11.1.4.2	Espaciados en modo vertical . . . . .	551
11.1.4.3	Control aún más fino del salto vertical .	555
11.1.5	Modificando títulos y nombres de secciones . . .	556

11.1.6	Definiendo nuestros propios identificadores de secciones . . . . .	564
11.1.7	Fuentes en L <sup>A</sup> T <sub>E</sub> X 2 <sub>ε</sub> . . . . .	565
11.1.7.1	La fuente del documento entero . . . . .	565
11.1.7.2	Seleccionando una fuente en concreto . . . . .	569
11.1.7.3	Fuentes con efectos artísticos . . . . .	571
11.1.8	Colores en L <sup>A</sup> T <sub>E</sub> X 2 <sub>ε</sub> . . . . .	572
11.1.8.1	Uso de color y modelos de color . . . . .	572
11.1.8.2	Poniendo texto de color . . . . .	575
11.1.8.3	Poniendo el fondo de color . . . . .	576
11.2	Revisitando temas anteriores . . . . .	578
11.2.1	Revisitando los flotantes . . . . .	578
11.2.1.1	Recomendando una ubicación al maqueta- dador inteligente . . . . .	579
11.2.1.2	Forzando la ubicación del flotante, el mo- delo de “maquetador obediente” . . . . .	580
11.2.1.3	Haciendo que el texto rodee al flotante . . . . .	581
11.2.2	Revisitando las listas . . . . .	584
11.2.2.1	Listas compactas . . . . .	585
11.2.2.2	Redefiniendo los símbolos . . . . .	586
11.2.2.3	Contadores de enumeraciones . . . . .	590
11.2.3	Revisitando las tablas . . . . .	592
11.2.3.1	Definiendo la geometría de las tablas . . . . .	592
11.2.3.2	Tablas más complejas . . . . .	594
11.2.4	Tablas más estéticas . . . . .	597
11.2.4.1	Tablas apaisadas . . . . .	600

11.2.4.2	Definiendo colores en las tablas . . . . .	602
11.2.4.3	Tablas inmensas . . . . .	607
11.3	Nuevas cosas interesantes que hacer . . . . .	611
11.3.1	Formularios . . . . .	611
11.3.2	Incluyendo listados y salidas de programas . . . . .	613
11.3.3	Creando tus propios comandos, macros y entornos	622
11.3.4	Opciones de ayuda a maquetación . . . . .	628
11.3.5	Incluyendo URLs y enlaces en el PDF . . . . .	628
11.3.6	Escribiendo textos con alfabeto Hangul con L <sup>A</sup> T <sub>E</sub> X 2 <sub>ε</sub>	636
11.3.7	Cajas y el entorno <code>minipages</code> . . . . .	638
11.4	Depurando en L <sup>A</sup> T <sub>E</sub> X 2 <sub>ε</sub> . . . . .	642
11.4.1	Operadores y comandos frágiles . . . . .	642
11.4.2	Error: las páginas pares aparecen descuadradas .	644
11.4.3	Error: no me aparece la cabecera en la primera pá- gina del capítulo . . . . .	644
11.4.4	Error: “Cite undefined” . . . . .	645
11.4.5	Error: aparece ?? en lugar de la cita o la referencia	645
11.4.6	Error: no aparecen <code>location</code> ni <code>telephone</code> en las cartas si no se incluye el comando <code>address</code> .	645
11.4.7	Error: “No me aparece la sangría en el primer pá- rrafo de cada capítulo” . . . . .	645
11.4.8	Error: “! Package inputenc Error: Unicode char <200b>(U+200B)” . . . . .	646
11.4.9	Error: “No room for a new loquesea” . . . . .	646
11.5	L <sup>A</sup> T <sub>E</sub> X 2 <sub>ε</sub> como lenguaje de programación . . . . .	647
<b>12</b>	<b>Y ahora, ¿Qué?</b>	<b>653</b>



Luego elegimos la plataforma, descargamos el instalador y lo instalamos:



Seguimos las instrucciones, y listo.

Cuando lo abramos, podremos editar texto en  $\text{\LaTeX} 2_{\epsilon}$ . La duda que tendremos es cómo generar a partir de nuestro texto en  $\text{\LaTeX} 2_{\epsilon}$  el fichero en calidad profesional. Esto lo haremos “compilando” el código en  $\text{\LaTeX} 2_{\epsilon}$ . El resultado de esta operación será un PDF maquetado.

La opción para “compilar” un fichero  $\text{\LaTeX} 2_{\epsilon}$  está en “Herramientas ->Compilación rápida”. Recuerda antes de “compilar” un archivo guardarlo con la extensión *.tex*.

A partir de ahora, podemos utilizar texmaker como el “editor de textos”, y  $\text{\LaTeX} 2_{\epsilon}$  será nuestro “maquetador inteligente” que, a partir de las órdenes que le demos, generará un fichero PDF maquetado y de calidad profesional.

### 2.4.3. Qué puede salir mal

A diferencia de en MacOS X y en Linux, en Windows pueden salir mal algunas cosas, y es interesante que sepamos qué error da para poder arreglar lo que ha fallado.

# Capítulo 3

## Conceptos básicos de L<sup>A</sup>T<sub>E</sub>X 2<sub>ε</sub>

---

*Todo texto tiene su estructura propia. La diferencia entre usar L<sup>A</sup>T<sub>E</sub>X 2<sub>ε</sub> y otros sistemas es que en L<sup>A</sup>T<sub>E</sub>X 2<sub>ε</sub> debemos informar de forma explícita al sistema de esta estructura; ya que en L<sup>A</sup>T<sub>E</sub>X 2<sub>ε</sub> delegamos la presentación final<sup>†</sup> en el propio L<sup>A</sup>T<sub>E</sub>X 2<sub>ε</sub>; y esta depende de la estructura del texto. Estructuramos en texto mediante entornos, y damos órdenes a nuestro “maquetador inteligente” mediante comandos para especificar el formato.*

---

Para editar un documento en L<sup>A</sup>T<sub>E</sub>X 2<sub>ε</sub> lo más importante es tener muy claro qué es lo que vamos a escribir. La presentación estética es lo de menos en un principio, ya que esa será la tarea de la que se encargará en principio el propio L<sup>A</sup>T<sub>E</sub>X 2<sub>ε</sub>. No es que no podamos variar la presentación; de hecho, más adelante veremos como variarla. Sin embargo, lo normal es dejar primero que L<sup>A</sup>T<sub>E</sub>X 2<sub>ε</sub> haga su trabajo según los estándares y las convenciones de maquetación; y luego nosotros ajustemos lo que no nos guste.

---

<sup>†</sup> Tranquilo, de momento nos limitaremos a delegar en L<sup>A</sup>T<sub>E</sub>X 2<sub>ε</sub> la presentación final. Cuando conozcamos el lenguaje para “delegar tareas” en nuestro maquettador inteligente pero independiente, entonces podremos aprender cómo hacer que el documento tenga la apariencia que queremos. Para que te hagas una idea de dónde llegaremos, este libro está maquetado íntegramente con L<sup>A</sup>T<sub>E</sub>X 2<sub>ε</sub>; y en este libro aprenderás todo lo necesitas para conseguir una terminación como la de que ves en estas páginas.

Ahora vamos a comenzar centrándonos en algunos aspectos cruciales para entender la composición de un texto en L<sup>A</sup>T<sub>E</sub>X 2<sub>ε</sub>. Vamos a comenzar estudiando, pues, un conjunto de “retales” de L<sup>A</sup>T<sub>E</sub>X 2<sub>ε</sub> que constituye lo que tiene en común todo texto en L<sup>A</sup>T<sub>E</sub>X 2<sub>ε</sub>.

De entrada: es **muy** importante que nos acostumbremos a trabajar en L<sup>A</sup>T<sub>E</sub>X 2<sub>ε</sub> como se trabaja en el mundo profesional de redacción de artículos y escritura de libros: primero nos centramos en el contenido; y solo nos preocupamos de la estética al final, terminado el proceso creativo de redacción del libro. Primero vamos a escribir lo que queramos escribir, despreocupándonos del formato en sí. **En esta primera fase, sí indicaremos en el texto qué es cada cosa –qué es un nombre de capítulo, qué va en un pie de página, qué va en un cuadro–, pero nunca dónde va –cómo es el formato del nombre de capítulo y si aparece o no en la cabecera, en qué página va el pie de página, donde se va a maquetar finalmente el cuadro–.** En esta primera fase también podemos incluir otros aspectos semánticos distintos de la distribución del texto en capítulos y secciones: qué hay que enfatizar, qué es importante, qué hay que poner en secuencia. Si hay imágenes, en qué zona están. **Repito: en qué zona van.** Esto se lo vamos a pasar a nuestro “maquetador inteligente”, que es el “compilador” de L<sup>A</sup>T<sub>E</sub>X 2<sub>ε</sub>. El maquetador lo monta, y nos lo devuelve para revisión un PDF. Ya con el PDF en la mano, seremos nosotros los que decidimos que queremos más margen, cambiar la longitud de la sangría francesa, una fuente sin serifa... políticas genéricas para todo el documento. Algunas fotos decimos que queremos que vayan en un sitio específico, por fuerza. El maquetador intenta hacer lo que sea posible con las restricciones adicionales que le hemos dado. El proceso se repite hasta que nos place el resultado. Entonces, y solo entonces, se hace algún microajuste suelto de formato. Pero un libro entero puede tener menos de media docena de microajustes si hemos hecho bien el proceso.

**Este es el proceso de trabajo con L<sup>A</sup>T<sub>E</sub>X 2<sub>ε</sub>; como vemos, filosóficamente idéntico a cómo trabaja una editorial, y en las antípodas de cómo trabaja Office.** La diferencia de este proceso de L<sup>A</sup>T<sub>E</sub>X 2<sub>ε</sub> con el proceso de una editorial, es que utilizando L<sup>A</sup>T<sub>E</sub>X 2<sub>ε</sub> podemos llamar a nuestro maquetador todas las veces que queramos: es gratis.

L<sup>A</sup>T<sub>E</sub>X 2<sub>ε</sub> tiene comandos para hacer las tres cosas: especificación general del contenido del documento, cambiar el diseño y el criterio de

Pero aquí no acaba la cosa, ya que Babel no hace apenas esto. Con Babel podemos crear un documento que contenga varias lenguas, y elegir entre una y otra en distintas partes del documento. La lengua nos determinará, entre otras cosas, los patrones de separación silábica, así como algunos patrones de espaciado y formateo y muchos detalles estéticos.

### 3.4.2. Manos a la obra con Babel

El primer paso para trabajar en Babel es incluir la línea:

```
\usepackage[idiomas]{babel}
```

Antes del:

```
\begin{document}
```

donde `idiomas` son un conjunto de idiomas, separados por comas. Algunos de los idiomas soportados en la opción `idiomas` con el idioma al que corresponden son:

Nombre en Babel	⇒	Idioma
american	⇒	Ingles americano
austrian	⇒	Alemán austriaco
brazil	⇒	Portugués brasileño
catalan	⇒	Catalán
croatian	⇒	Croata
czech	⇒	Checo
danish	⇒	Danés
dutch	⇒	Alemán
english	⇒	Inglés
esperanto	⇒	Esperanto
finnish	⇒	Suomi
francais	⇒	Francés
french	⇒	Francés
galician	⇒	Gallego
german	⇒	Alemán
germanb	⇒	Variante Alemán

italian	⇒	Italiano
magyar	⇒	Húngaro
norsk	⇒	Noruego
nynorsk	⇒	Variante Noruego
polish	⇒	Polaco
portugues	⇒	Portugués
romanian	⇒	Rumano
russian	⇒	Ruso
slovak	⇒	Eslovaco
slovene	⇒	Esloveno
spanish	⇒	Español
swedish	⇒	Sueco
turkish	⇒	Turco

El idioma por defecto es el último especificado, por lo que si hacemos:

```
\usepackage[english,spanish]{babel}
```

estamos diciendo a Babel que cargue los módulos de inglés y español, de forma que podamos trabajar con los dos. Sin embargo, también estamos diciendo que comenzaremos trabajando en español.

En cualquier momento podremos pasar de un idioma a otro con:

```
\selectlanguage{idioma}
```

seleccionando solo uno para su uso. Por ejemplo, un texto válido en L<sup>A</sup>T<sub>E</sub>X 2<sub>ε</sub> es:

```
\selectlanguage{spanish}
Machado de Assis decía en su excelente libro
\selectlanguage{brazil}
‘‘Memórias Póstumas de Brás Cubas’’ :
{\em Esta, ao reconhecer-me, ficou pálida, e baixou os
olhos; mas foi obra de um instante. Ergueu logo a cabe
ça, e fitou-me com muita dignidade.}
```

los enfatizados del documento por otra forma de escribirlos –por ejemplo, ponerlos en negrita– si el resultado no nos gusta. Es decir, podemos decir que algo va en itálica con `{\it }` –modelo de “maquetador obediente”–, o podemos decir que algo hay que enfatizarlo con `{\em }`; y dejar que sea L<sup>A</sup>T<sub>E</sub>X 2<sub>ε</sub> el que según las convenciones de maquetación, el tipo de documento, o incluso una orden explícita en la cabecera de cómo debe interpretar el enfatizado en todo el documento sea el que determine si el enfatizado va a ser en itálica, en negrita, cambiando el color, o de cualquier otra forma alternativa. Entender esto es entender el 90 % de la filosofía de L<sup>A</sup>T<sub>E</sub>X 2<sub>ε</sub>: preocúpate por el contenido, deja que el maquetaador inteligente de formato según los convenios, convenciones y estándares de maquetación, si quieres cambiar algo que afecte a todo el documento das la orden en el preludio, y ya darás tú la última revisión vía comandos.

Algunos de los entornos que he comentado tienen otra forma de utilizarse: dentro de cláusulas `begin` y `end`. Vamos a ver algunos ejemplos.

Los entornos de centrado y alineado son:

```
\begin{center}
Texto centrado.
\end{center}

\begin{flushright}
Texto alineado a la derecha
\end{flushright}

\begin{flushleft}
Texto alineado a la izquierda
\end{flushleft}
```

Y el resultado obtenido de estos ejemplos es:

---

Texto centrado.

Texto alineado a la derecha

Texto alineado a la izquierda

---

El resultado será:

---



---

Estamos incluyendo la imagen `davidsantoorcero.jpg`; que es un jpeg. Ojo: la imagen se verá tal cual está, por lo que puedes tener una imagen que sea más grande que la página, y que rebase los márgenes. Para ajustarla al tamaño, tendremos que escalarla.

#### 4.8.4. Escalando la imagen

También podemos tener interés en modificar el tamaño de la imagen importada escalándola al hueco que queramos que ocupe. Para ello tenemos dos parámetros: `height` y `width`.

`width=anchura` indica la anchura a la que vamos a escalar la imagen que estamos importando, en cualquiera de las unidades soportadas por L<sup>A</sup>T<sub>E</sub>X 2<sub>ε</sub>. Por ejemplo, podemos hacer:

```
\begin{center}  
\includegraphics[width=3.5cm]{davidsantoorcero.jpg}  
\end{center}
```

Que genera como salida:

---



# Capítulo 6

## Presentaciones en $\text{\LaTeX 2}_{\epsilon}$

---

*Hay un aspecto muy importante y de uso muy frecuente en  $\text{\LaTeX 2}_{\epsilon}$ : las presentaciones. Querer hacer una presentación con una terminación perfecta, y pudiendo aprovechar las gráficas, las ecuaciones y parte del texto de nuestros artículos o de la tesis. Vamos a aprender ahora a hacer presentaciones en  $\text{\LaTeX 2}_{\epsilon}$ .*

---

### 6.1. El paquete `beamer`

Existen muchas clases que podemos utilizar para incluir presentaciones en  $\text{\LaTeX 2}_{\epsilon}$ . Personalmente he probado varias de ellas; y te tengo que comentar que hay grandes diferencias entre ellas.

Después de utilizar media docena de estas clases, la que me ha sido más cómoda de utilizar ha sido, con diferencia, `beamer`. Tiene una sintaxis bastante fácil de aprender, y además permite utilizar comandos de las clases `prospect` y `foils` para presentaciones. Actualmente la podemos considerar, de hecho, la más popular de las distintas clases existentes para hacer presentaciones.

Lo primero, entender qué hace y qué no hace una clase de presentaciones para  $\text{\LaTeX 2}_{\epsilon}$ . Todas funcionan de forma análoga: partimos de



Significa “ponme en todas las capas que deriven de este `frame`, a partir de la segunda, el punto `Punto B`”.

Ojo: el número de capa corresponde con el número de transparencia **dentro del** `frame`. Por lo que habrá una capa 1 para cada `frame`. Las capas se comienzan a contar por el 1.

La sintaxis general del rango será:

```
<min-max> Texto
```

Donde `min` será la primera capa del `frame` en la que se verá el texto, y `max` será la última capa del `frame` en la que se verá el texto. Si no incluimos `min`, comenzará desde la primera; y si no incluimos `max`, terminará en la última.

Podemos especificar rangos, números de capas separados por comas, y rangos y números de capas separados por comas. Por ejemplo, si hacemos:

```
\item<1-3,5,7-9,11-> Punto A
```

Entonces el `item` del `itemize` con texto `Punto A` aparecerá en las capas 1, 2, 3, 5, 7, 8, 9, 11 y posteriores a la capa 11. Por otro lado, no aparecerá en las capas 4, 6 y 10.

Podemos marcar el punto actual de un `itemize` que estemos en cada momento abriendo el `itemize` con:

```
\begin{itemize}[<+- alert@+>]
```

Haciendo:

```
\begin{frame}
\begin{block}{Nombre bloque}
Ejemplo {\tt alert} genérico
\begin{itemize}[<+-| alert@+>]
\item Loquesea
\item Otro loquesea
\item Loquesea final
\end{itemize}
\end{block}
\end{frame}
```

Dónde insertamos el logo dependerá del tema; aunque podemos forzar insertar el logo en un punto concreto haciendo:

```
\insertlogo
```

## 6.7. Entornos frágiles y presentaciones

A veces una presentación fallará porque estamos utilizando comandos frágiles dentro de un `frame`; generando un error arcano –normalmente, que ha llegado al final del texto sin encontrar el cierre del entorno frágil aunque sí hemos puesto el cierre requerido–. Nos podemos volver bastante locos hasta descubrir qué es este error si no conocemos este problema; si lo conocemos, cuando veamos cierres de entorno que son ignorados, podemos sospechar en primer lugar que hemos invertido el orden de cierre, algo que `beamer` gestiona muy mal. Si no es así, probablemente es un entorno frágil dentro de un `frame`.

Un escenario en el que ocurre y que es habitual es cuando utilizamos un `verbatim` dentro de un entorno `frame`. Por ejemplo, esto generará un error:

```
\begin{frame}
\begin{block}
    {La forma simple de definir la función principal}
\begin{verbatim}
int main (void){

}
\end{verbatim}
\end{block}
\end{frame}
```

Respecto a lo de los comandos frágiles, explicaremos qué son y por qué existen en la página 642 de este mismo libro. No es problema, porque tenemos cómo solucionarlo. Basta con añadir al entorno `frame` el modificador `agile`; haciendo:

```
\begin{frame}[fragile]
```

Si configuramos el proyector y el monitor que ve el ponente como una única pantalla cuyo tamaño es la suma de ambas, significa que una aplicación a pantalla completa nos mostrará parte de su contenido en una pantalla, y parte del contenido en la otra. Y este es precisamente el truco que usa `beamer`.

A efectos de nomenclatura de `beamer`, se considera monitor primario el que va a representar la imagen en el proyector, y monitor secundario el que solo ve el ponente. Es intrascendente cual es el monitor configurado como primario de sistema, y el configurado como secundario de sistema, ya que ni `beamer` ni el visor PDF van a usar eso.

La idea es que cuando `beamer` genera la presentación, genera una presentación que tiene el doble de ancho –o de alto– que una presentación normal; y pone en la mitad de cada página del fichero de presentación la transparencia que corresponda, y en la otra mitad las anotaciones de la susodicha transparencia que corresponde. Si presentas ese PDF a pantalla completa, saldrá la transparencia por el proyector, y las anotaciones por el monitor.

Si ves esto confuso, no te preocupes. Vamos a ver más adelante un ejemplo que lo aclarará.

Para hacer presentaciones a dos monitores, lo primero que debemos hacer es incluir el paquete `pgfpages`:

```
\usepackage{pgfpages}
```

En el preámbulo del documento. Inmediatamente después, dentro del preámbulo, incluimos la instrucción:

```
\setbeameroption{show notes on second screen=ubicación}
```

Donde `ubicación` puede ser:

- `right`: Las anotaciones estarán a la derecha de la transparencia.
- `left`: Las anotaciones estarán a la izquierda de la transparencia.
- `bottom`: Las anotaciones estarán debajo de la transparencia.
- `top`: Las anotaciones están encima de la transparencia.

# Capítulo 8

## Bibliografía con BIBTEX

---

*Las bibliografías son una de las tareas más pesadas a las que te enfrentarás en tu tesis doctoral, y no es precisamente una tarea cómoda en artículos científicos o en tu trabajo fin de grado o de master. Un montón de entradas que tienen que ser escritas exactamente en el mismo formato, con referencias cruzadas dispersas por todo el texto y en las que además tenemos que rehacer el trabajo más pesado –verificar las referencias y los formatos– cada vez que añadimos algo. Trabajo, en principio, no reutilizable cada cambio de revista, o para escribir el cuerpo de tu tesis a partir de artículos ya publicados. ¿Tendrá  $\text{\LaTeX}$  algo para ayudarnos a hacer este trabajo?*

---

Si hemos hecho la pregunta de esta forma, es porque ya sabemos la respuesta: sí.  $\text{\LaTeX}$  tiene un sistema de bibliografía muy potente, el BIBTEX, que es muy cómodo tanto para aquellos que necesitan ocasionalmente mandar algún artículo a una revista o hacer algún trabajo escolar como para los que están escribiendo un libro o una tesis. De hecho, los escritores de tesis y tesinas son los principales beneficiados de BIBTEX, ya que pueden reaprovechar las bibliografías de los artículos que van publicando; y según trabajas en tu tesis, te creas una base de datos bibliográfica que te servirá de inestimable ayuda tanto para publicar artículos, como para la redacción final de la tesis.

## 8.1. Filosofía de BIB<sub>T</sub>E<sub>X</sub>

La idea básica de BIB<sub>T</sub>E<sub>X</sub> es tener un archivo que contenga la base de datos bibliográfica completa, con todas las entradas bibliográficas de todos los artículos y libros con los que vayas trabajando, y que vayas generando de forma incremental. Ese archivo estará en un formato neutro, que ya te aviso que es un estándar y que incluso te puedes descargar ya las referencias bibliográficas en ese formato. Según trabajas en tu tesis doctoral, acumulas una gran cantidad de artículos que has utilizado o consultado.

Cuando quieras hacer un artículo, la memoria de la tesis o del trabajo de fin de grado o de master, en el texto citarás lo que creas que debes citar; y será BIB<sub>T</sub>E<sub>X</sub> el que, de forma automática, extraerá de la base de datos bibliográfica los artículos citados, los incluirá en la bibliografía, y le dará a la bibliografía el formato exacto que se te ha requerido.

Con esto, se acabó el andar manteniendo y revisando la bibliografía, comprobando que no te has dejado referencias en la bibliografía de documentos que luego no citas; y puedes reutilizar la totalidad de la base de datos entre artículos, o entre artículos y la tesis. El trabajo ahorrado es realmente ingente.

Debemos tener en cuenta que normalmente trabajaremos en nuestro documento en el mismo tema; y en documentos futuros, trabajaremos con temas análogos. Esto hará que, con el tiempo, nuestra base de datos bibliográfica se irá haciendo cada vez más grande, por lo que hacer bibliografías es cada vez más sencillo –como sabrás, es normal dentro de un área de conocimiento que se cite siempre un conjunto determinado de artículos como de referencia, por lo que con frecuencia la bibliografía entre dos artículos de la misma temática puede actualizarse, pero no cambia en la mayor parte de las referencias citadas–. Siempre que redactemos un nuevo artículo en L<sup>A</sup>T<sub>E</sub>X 2<sub>ε</sub> incorporaremos las citas que todavía no tuviéramos a nuestra base de datos bibliográfica e indicaremos en el cuerpo del texto qué citamos esta vez.

Esta base de datos estará en un archivo de extensión “.bib”, de cuyo formato hablaremos más adelante. Tenemos varias alternativas para generar esta base de datos; que no son mutuamente excluyentes. La primera, bajar de Internet una base de datos bibliográfica en formato

Podemos citarla con:

```
\cite{DSOpfc}
```

Recordemos que el campo de referencia debe ser único; es decir, no debe haber dos entradas bibliográficas que tengan la misma referencia. La referencia es un campo imprescindible, ya que necesitamos de él para referenciar la cita. El campo referencia no puede tener espacios.

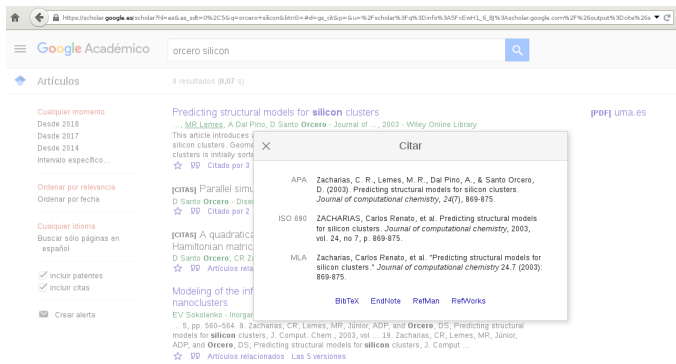
Un detalle importante es el relativo al orden de las entradas en el fichero `.bib`: el orden de las entradas bibliográficas es indiferente, es decir, no tenemos que poner ningún orden específico en las entradas bibliográficas a priori para que  $\text{\LaTeX}$  funcione. El orden de los campos de cada entrada bibliográfica también es indiferente, es decir, podemos poner los campos dentro de una referencia bibliográfica como nos interese. La única excepción es el campo referencia, que identifica unívocamente la referencia bibliográfica y que va siempre el primero.

## 8.4.2. Extrayendo las entradas $\text{\LaTeX}$ de la Web

Para rellenar los campos, una primera aproximación perezosa es no saber de  $\text{\LaTeX}$ , y no complicarse la existencia. La mayor parte de los sitios de referencias bibliográficas permiten exportar la referencia en formato  $\text{\LaTeX}$ . Por ejemplo, si es a través de Google Scholar, realizamos la búsqueda del artículo que queremos referenciar:

The screenshot shows a Google Scholar search interface. The search bar contains 'orcero silicon' and the results are displayed as a list of articles. The first article is 'Predicting structural models for silicon clusters' by MR Lemus, A Dal Pino, D Santo Orcero, published in the Journal of ... in 2003. The second article is '[CITAS] Parallel simulation and optimization of silicon clusters' by D Santo Orcero, published in a dissertation in 2002. The third article is '[CITAS] A quadratical scaling variation of the Lanczos method for tight-binding Hamiltonian matrices' by D Santo Orcero, CR Zacharias, and A Falcó. The fourth article is 'Modeling of the influence of defects on the electronic structure of silicon nanoclusters' by EV Sokolenko, published in Inorganic Materials in 2015. The interface includes filters for 'Cualquier momento' (Any time), 'Ordenar por relevancia' (Sort by relevance), 'Cualquier idioma' (Any language), and checkboxes for 'incluir patentes' (include patents) and 'incluir citas' (include citations). There is also a 'Crear alerta' (Create alert) button.

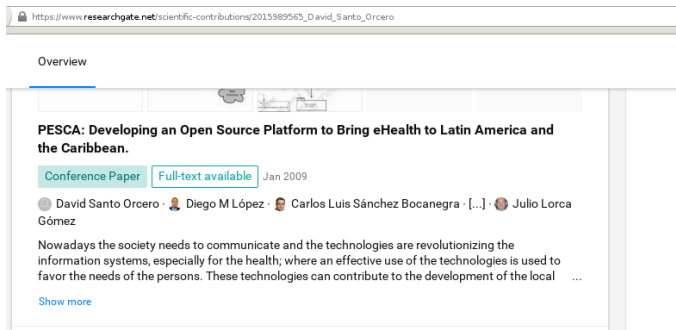
Seleccionamos las comillas que aparecen debajo del artículo a la derecha de la estrella, y nos aparece un desplegable:



Seleccionamos BibTeX, y podemos descargar la entrada bibliográfica ya lista:

```
@article{zacharias2003predicting,
title={Predicting structural models for silicon clusters},
author={Zacharias, Carlos Renato and
Lemes, Maurício Ruvo and Dal Pino, Arnaldo
and Santo Orcero, David},
journal={Journal of computational chemistry},
volume={24},
number={7},
pages={869--875},
year={2003},
publisher={Wiley Online Library}
}
```

Además de en Google Scholar, lo encontramos en otros sitios. Por ejemplo, en Research Gate, buscamos el artículo:



Pulsando en el título, entramos en la página del artículo:

The screenshot shows the ResearchGate interface. At the top, there's a search bar and navigation links. Below, the article title is prominently displayed. Underneath the title, it says 'Conference Paper (PDF Available)' and 'January 2009 with 48 Reads'. There are links for 'See all > 6 References' and 'See all > 3 Figures'. At the bottom of the visible section, there's a link that says 'Cite this publication' with a download icon.

Dentro de la página del artículo, seleccionamos “Cite this publication” y nos aparece este menú:

The modal window titled 'Download citation' is centered over the article page. It contains two sections: 'What type of file do you want?' with radio buttons for 'RIS' (selected), 'BibTeX', and 'Plain Text'; and 'What do you want to download?' with radio buttons for 'Citation only' (selected) and 'Citation and abstract'. A blue 'Download' button is at the bottom right.

Seleccionamos BibTeX, y listo. También podemos exportar a BibTeX en DBLP:

The screenshot shows the DBLP profile page for David Santo Orcero. The URL is 'https://dblp.uni-trier.de/pers/hd/or/Orcero:David\_Santo'. The page header includes the DBLP logo and 'computer science bibliography'. Below the header, there's a navigation bar with 'Home > Persons'. The main content area shows a list of publications, with the first one from 2003 selected. A dropdown menu is open for the selected publication, showing options to 'export record' in various formats: BibTeX, RIS, RDF N-Triples, RDF/XML, and XML. The 'dblp key' is also displayed: 'journalst/jcc/ZachariasLJO03'.

Y en otras muchas páginas de búsqueda de referencias bibliográficas.



### 8.4.3. Los campos de las referencias bibliográficas

Una referencia bibliográfica tiene gran cantidad de información. Esta información está organizada en campos, cada uno de los cuales debe tener un tipo de información predeterminada y fija para poder referenciarse de forma correcta en una bibliografía; así como campos que son opcionales. Esto tiene su lógica, similar a la de L<sup>A</sup>T<sub>E</sub>X 2<sub>ε</sub>; nosotros no decimos qué va en cursiva o qué va en negrita; nosotros lo que diremos a L<sup>A</sup>T<sub>E</sub>X 2<sub>ε</sub> es quién es el autor, cual es el título de la referencia, unos datos acerca de como localizarla, y le dejamos el formato estilo de bibliografía a él.

#### 8.4.3.1. Campos obligatorios

Cada tipo distinto de referencia bibliográfica tiene un conjunto de campos que son obligatorios. Estos campos dependen del tipo de referencia bibliográfica; y si no los incluimos, aunque L<sup>A</sup>T<sub>E</sub>X 2<sub>ε</sub> y BibT<sub>E</sub>X sean capaces de generar la bibliografía, esta no va a cumplir las normas bibliográficas de ese estilo –además, quedando bastante antiestética–. Los campos obligatorios coinciden con aquellos que la mayor parte de las revistas consideran como imprescindibles, y con aquellos que son imprescindibles para localizar un documento y son comunes en una bibliografía.

Algo que he escuchado de algún alumno es la pregunta de qué hacemos si no tenemos suficiente información para un campo obligatorio de una referencia bibliográfica. Si disponemos del documento citado, o podemos localizarlo de forma inconfundible con la información que tenemos, probablemente es que estamos utilizando un tipo de referencia bibliográfica L<sup>A</sup>T<sub>E</sub>X 2<sub>ε</sub> equivocada, citando un trabajo de procedimiento como un libro, o un libro como artículo. Otro escenario muy distinto es que no dispongamos del artículo, no lo hemos leído, y queremos incorporarlo en la bibliografía. Aquí es cierto que pueden faltarnos datos para realizar la entrada bibliográfica correcta en L<sup>A</sup>T<sub>E</sub>X 2<sub>ε</sub>; pero debemos recordar que si lo que vamos a citar es lo suficientemente importante para ir en la bibliografía, significa que merece el esfuerzo de buscarlo y al menos ojearlo. Si no lo hemos leído, y no merece el esfuerzo ni de buscarlo, no debe estar en la bibliografía, y no debemos citarlo. No tiene

## Capítulo 9

# Edición de fórmulas matemáticas en $\text{\LaTeX} 2_{\epsilon}$ con $\mathcal{A}\mathcal{M}\mathcal{S}\text{-}\text{\LaTeX}$

---

*Uno de los puntos fuertes de  $\text{\LaTeX} 2_{\epsilon}$  es la facilidad para escribir funciones matemáticas. Mediante un lenguaje especial que vamos a aprender especificamos como es la fórmula, y  $\text{\LaTeX} 2_{\epsilon}$  se encarga de maquetarla. Quien ha visto los resultados de las fórmulas en  $\text{\LaTeX} 2_{\epsilon}$  conoce la increíble terminación de estas.*

---

Cuando estudiaba la carrera de Ingeniería Informática –en la época, aún era “licenciatura en informática”–, tuve que hacer un trabajo bastante grande para una asignatura que involucraba muchas fórmulas. Comencé haciendo el trabajo no recuerdo bien si en WordStar o en WordPerfect. Aquello era un trabajo largo, tedioso y extremadamente complicado. Cada fórmula salía a sangre y hierro. Con los procesadores de textos modernos, no habría cambiado mucho el esfuerzo. En aquella época, llevaba ya medio año trabajando en Linux; y todavía lo usaba apenas para programar. Tomé la decisión de intentar reescribir el trabajo en  $\text{\LaTeX} 2_{\epsilon}$ —en la época,  $\text{\LaTeX}$ —, porque había escuchado que era muy bueno para eso de las fórmulas. Con un chuleterio que encontré con algunos comandos –Internet aún estaba fuera del alcance de los alumnos de la Universidad, salvo que te colaras en la sala de terminales del centro de cálculo y supieras como entrar en el VAX y defenderte con el Gopher

La capacidad de agrupamiento de estos operadores la comparten muchos otros: la tienen también `\widehat`, `\widetilde`, y todos los acentos anchos disponibles en L<sup>A</sup>T<sub>E</sub>X 2<sub>ε</sub>, y de los que ya hemos hablado en las tablas de símbolos incluidas con anterioridad.

#### 9.4.3.6. Negando operadores

Muchos símbolos se pueden negar. En la práctica esto supone que L<sup>A</sup>T<sub>E</sub>X 2<sub>ε</sub> incluya una barra adicional que “tache” el operador. En L<sup>A</sup>T<sub>E</sub>X 2<sub>ε</sub> conseguimos esto con el operador `\not`.

Por ejemplo, un operador afirmado sería:

```
\begin{equation}
a < b \Rightarrow a \in A
\end{equation}
```

Lo que genera como salida:

---

$$a < b \Rightarrow a \in A \qquad (9.72)$$

---

Por otro lado, podemos negar el operador:

```
\begin{equation}
i \not< j \not\Rightarrow a \not\in A
\end{equation}
```

Lo que a su vez genera como salida:

---

$$i \not< j \not\Rightarrow a \not\in A \qquad (9.73)$$

---

Ojo: no debe haber espacio entre `\not` y el operador que se va a negar para que funcione `\not` si el operador comienza con `\` o por un símbolo. Si el operador comienza por una letra, dejaremos un espacio simple.

Es importante que recordemos que  $\mathcal{A}\mathcal{M}\mathcal{S}$ -L<sup>A</sup>T<sub>E</sub>X tiene caracteres propios para representar algunos símbolos de negación definidos en el paquete `amssymb`; y que hemos visto en secciones anteriores de este capítulo. A veces nos interesa más utilizarlos, o a veces nos interesa negar con `\not`. Veámoslo con un ejemplo. Si incluimos dicho paquete, podemos hacer lo anterior con:

```
\begin{eqnarray}
i \not< j \not\rightarrow a \not\in A \\
i \nless j \nrightarrow a \notin A
\end{eqnarray}
```

Que genera como salida:

---

$$i \not< j \not\rightarrow a \not\in A \tag{9.74}$$

$$i \nless j \nrightarrow a \notin A \tag{9.75}$$

---

Difícil de diferenciar los resultados. Como curiosidad, el símbolo  $\ni$  no tiene un comando fácil para representar el glifo del susodicho símbolo tachado; se puede tachar mediante paquetes que incluyen  $\nexists$  como glifo independiente, pero son paquetes que “tocan” todo el modo matemático, y traen más complicaciones de las que solucionan. Tachar el glifo  $\ni$  es, por otro lado, tan sencillo como hacer `\not\ni` en entorno matemático. El concepto clave aquí es que, independientemente de que no tengamos el comando para dibujar un glifo tachado en concreto, en la práctica podemos tachar casi cualquier cosa.

### 9.4.3.7. Las llaves

Podemos poner llaves en una fórmula, utilizando el entorno `cases`, que es subentorno del entorno matemático. Dentro de este entorno podemos usar `&` para centrar y tabular.

Un ejemplo del uso de `cases` es:

```
\begin{equation}
S_n=
\begin{cases}
n & \text{si } n \text{ es par} \\
\frac{1}{n} & \text{si } n \text{ es impar}
\end{cases}
\end{equation}
```

Que genera como salida:

$$S_n = \begin{cases} n & \text{si } n \text{ es par} \\ \frac{1}{n} & \text{si } n \text{ es impar} \end{cases} \quad (9.76)$$

#### 9.4.3.8. Afinando los “castillos de fracciones”

Podemos conseguir fracciones anidadas de forma estética utilizando el comando `\cfrac`. Antiguamente teníamos `\lfrac` y `\rfrac`; `\lfrac` centraba las fracciones a la izquierda, y `\rfrac` centraba las fracciones a la derecha. `\cfrac` deja escoger el centrado a L<sup>A</sup>T<sub>E</sub>X 2<sub>ε</sub>, y funciona actualmente tanto para fracciones a la izquierda como para fracciones a la derecha, y `\lfrac` y `\rfrac` no funcionan ya.

Un ejemplo de centrado de fracciones por la derecha es:

```
\begin{equation}
\cfrac{1}{\sqrt{1}}+
\cfrac{2}{\sqrt{2}}+
\cfrac{3}{\sqrt{3}}+
\cfrac{4}{\sqrt{4}}+
\cfrac{5}{\sqrt{5}}+
\cfrac{6}{\sqrt{6}}+
\cfrac{7}{\sqrt{7}}+
\cfrac{8}{\sqrt{8}}+\dots
\end{equation}
```

# Capítulo 10

## Gráficos en L<sup>A</sup>T<sub>E</sub>X 2<sub>ε</sub>

---

*Los gráficos son una de las “bestias pardas” de los procesadores WYSIWYG. Por cuando saltan, y por cuando se “deshacen”. Pero son imprescindibles. Vamos a aprender en este capítulo cómo dibujar gráficos en L<sup>A</sup>T<sub>E</sub>X 2<sub>ε</sub>.*

---

L<sup>A</sup>T<sub>E</sub>X 2<sub>ε</sub> tiene un planteamiento completamente distinto para los gráficos que los procesadores WYSIWYG. O mejor dicho, tiene varios planteamientos completamente distintos. Cada planteamiento de los varios que tiene L<sup>A</sup>T<sub>E</sub>X 2<sub>ε</sub> tiene sus ventajas y sus inconvenientes, solo que todos tienen una faceta en común: funcionan. Puede que según el problema sean unos métodos más cómodos que otros, pero siempre conseguiremos aquello que nos proponemos, y no aquello que quiere el programa que hagamos. En este capítulo veremos algunos de los planteamientos que nos proporciona L<sup>A</sup>T<sub>E</sub>X 2<sub>ε</sub>, y que son más genéricos: dibujar elementos simples para componer dibujos; dibujar gráficos, y gráficos de barras, e incorporar imágenes dibujadas con un editor externo. Sin embargo, en L<sup>A</sup>T<sub>E</sub>X 2<sub>ε</sub> tenemos paquetes capaces de dibujar muchas cosas específicas: árboles, partituras musicales y hasta posiciones de piedras en tableros de weiqi<sup>1</sup> o fórmulas químicas. Siempre será más cómodo

---

<sup>1</sup>Juego de origen chino, con cuatro mil años de antigüedad, que se consideraba una de las artes clásicas en China.

diámetro cuyo centro estará a 11'25 centímetros a la derecha del margen izquierdo del flotante, y a siete centímetros y medio encima del margen inferior del flotante.

Podemos poner varios comandos a partir de un `put`. Por ejemplo:

```
\setlength{\unitlength}{1cm}
\begin{picture}(4,4)
\put(3,2){\circle{1.0}\circle{0.8}\circle{0.6}}
\end{picture}
```

Genera como salida:

---



---

Pone tres círculos concéntricos, de 1.0cm, 0.8cm y 0.6cm, centrados a tres centímetros sobre el eje X del centro de coordenadas, y a 2 cm sobre el eje Y del centro de coordenadas.

Muchas veces tenemos elementos repetidos. Afortunadamente, el entorno `picture` nos dota de un comando que nos permite dibujar varias veces la misma figura a partir de una definición única de un elemento. Para ello, tenemos el comando:

```
\multiput (coordenadaX, coordenadaY)
          (incrementoX, incrementoY)
          {número} {elemento}
```

Donde se repetirá el dibujo del elemento `elemento` el número de veces indicado en `número`, comenzando por la coordenada X especificada en `coordenadaX`, y la coordenada Y `coordenadaY`; e incrementando en cada repetición `incrementoX` a la coordenada X e `incrementoY` a la coordenada Y.

```
\setlength{\unitlength}{1cm}
\begin{picture}(18,2)
\put(1,1){\makebox(6,1)[t]{Dentro t}}
\put(1,1){\makebox(6,1)[b]{Dentro b}}
\put(1,1){\makebox(6,1)[tl]{Dentro tl}}
\put(1,1){\makebox(6,1)[br]{Dentro br}}
\put(1,1){\makebox(6,1)[s]{Dentro s}}
\end{picture}
```

Que genera como salida:

---

Dentro tl	Dentro t	Dentro tr
Dentro l	Dentro vacio	Dentro r
Dentro bl	Dentro b	Dentro br

---

### 10.3.2. Dibujando ovals

Aclaremos primero que en el entorno `picture` una oval es un rectángulo de puntas romas. Ahora que sabemos esto, vamos a ver cómo dibujar ovals. La sintaxis de una oval es:

```
\begin{picture}
\oval(anchoOval,altoOval)[parteOval]
\end{picture}
```

Donde `anchoOval` es el ancho de la oval, y `altoOval` es el alto de la oval.

`parteOval` determina qué parte de la oval se va a dibujar. Si no se indica, se dibuja la oval completa.



Donde cada barra del gráfico está determinada por una instrucción `bar`, y podemos poner tantas barras como, en principio, queramos -o quepan-. La sintaxis de una única línea `bar` es:

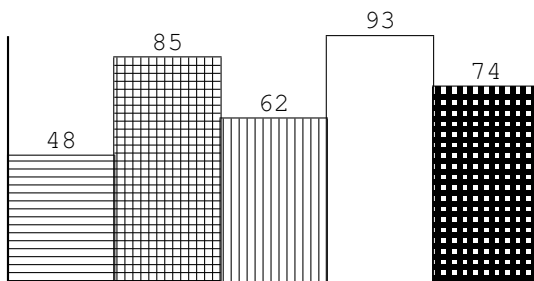
```
\bar{cantidad}{relleno}
```

Donde `cantidad` es la cantidad que modela la barra, y que determina su tamaño; y `relleno` es el patrón de relleno de la barra. Tenemos ocho rellenos distintos, que son los números entre 1 y 8; y el 0, que determina que se escriba la cantidad determinada por `cantidad`; pero no se dibuje la barra; lo que es distinto de 1, que significa *dibuja la barra y no pongas ningún patrón dentro*.

Un primer ejemplo de gráfico de barras es:

```
\begin{barenv}
\bar{48}{3}
\bar{85}{4}
\bar{62}{2}
\bar{93}{1}
\bar{74}{5}
\end{barenv}
```

Que genera como salida:

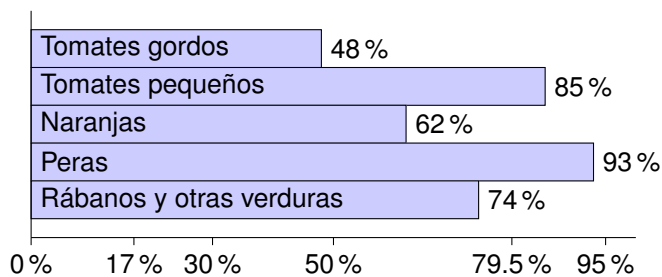


En este ejemplo tenemos un gráfico de cinco barras; donde la primera tendrá un tamaño de 48 y un patrón de relleno de 3 –es decir, estará rellena de líneas horizontales–, la segunda un tamaño de 85 y estará rellena de líneas horizontales y verticales formando cuadrados –patrón de relleno 4–, la tercera un tamaño de 62 y un patrón de relleno 2 –líneas

Por ejemplo, podemos hacer:

```
\begin{bchart}[steps={17,30,50,79.5,95},unit=\%]
\bcbar[text={Tomates gordos}]{48}
\bcbar[text={Tomates peque~nos}]{85}
\bcbar[text={Naranjas}]{62}
\bcbar[text={Peras}]{93}
\bcbar[text={R\'abanos y otras verduras}]{74}
\end{bchart}
```

Que genera como salida:



Ojo: si indicamos la lista de marcas con `steps`, el paquete `bchart` ignorará el parámetro `step` y no hará las marcas a intervalos regulares.

Finalmente, podemos esconder las marcas con:

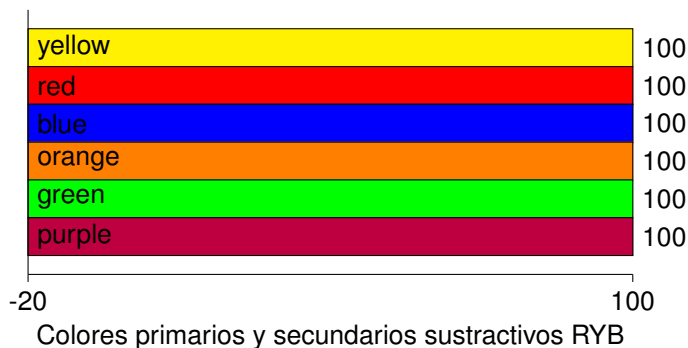
```
\begin{bchart}[plain]
```

Por ejemplo, podemos hacer:

```
\begin{bchart}[plain,unit=\%]
\bcbar[text={Tomates gordos}]{48}
\bcbar[text={Tomates peque~nos}]{85}
\bcbar[text={Naranjas}]{62}
\bcbar[text={Peras}]{93}
\bcbar[text={R\'abanos y otras verduras}]{74}
\end{bchart}
```

```
\bcbars[text={purple},color=purple]{100}
\bcxlabel
      {Colores primarios y secundarios sustractivos RYB}
\end{bchart}
```

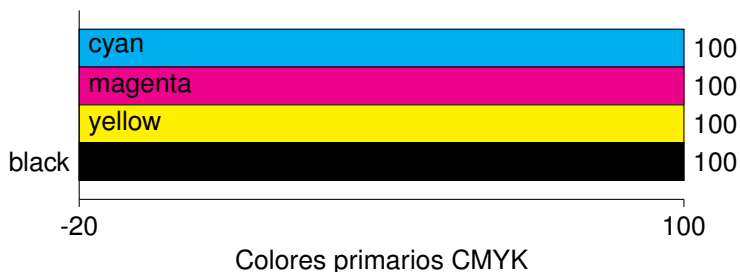
Que genera como salida (en el libro impreso se verá como tonos de grises):



Por ejemplo, podemos hacer con los colores CMYK:

```
\begin{bchart}
\bcbars[text={cyan},color=cyan]{100}
\bcbars[text={magenta},color=magenta]{100}
\bcbars[text={yellow},color=yellow]{100}
\bcbars[label={black},color=black]{100}
\bcxlabel{Colores primarios CMYK}
\end{bchart}
```

Que genera como salida (en el libro impreso se verá como tonos de grises):



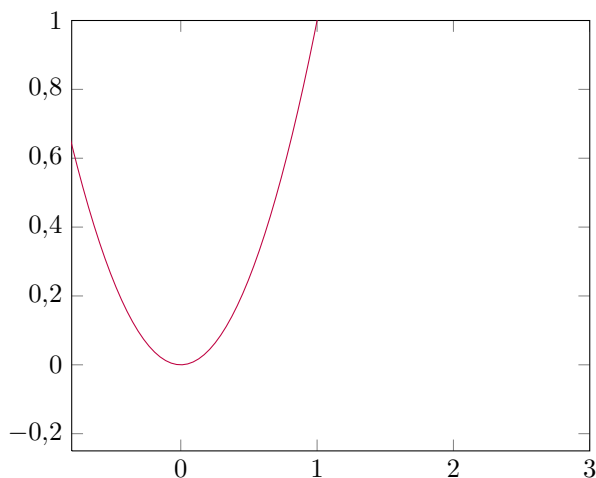
la misma barra invertida, y que la declaración de la función se hace entre llaves.

Por ejemplo, si hacemos:

```
\begin{tikzpicture}
  \begin{axis}[xmin=-0.8, xmax=3, ymin=-0.25, ymax=1]
    \addplot[purple, domain=-1:3, samples=100]
      plot (\x, {\x^2});
  \end{axis}
\end{tikzpicture}
```

Obtendremos:

---



---

Es importante observar que la gráfica no va a rebasar nunca los ejes; es decir, que los valores de la función cuyo  $X$  sea menor que  $X_{\min}$ , o mayor que  $X_{\max}$ ; o aquellos cuyo  $Y$  sea inferior a  $Y_{\min}$  o superior a  $Y_{\max}$  no se dibujan.

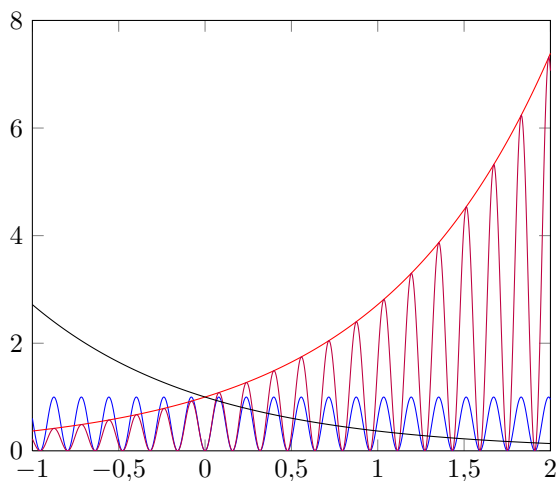
La fórmula a dibujar puede ser arbitrariamente compleja, e incluir funciones trigonométricas y exponenciales, entre otras muchas, así como constantes de uso común, como  $e$  o  $\pi$ ; aunque siempre por su denominación en inglés.

Podemos dibujar más de una gráfica en los mismos ejes, solamente añadiendo varios `addplot` dentro del mismo entorno `axis`. Por ejemplo, haciendo:

```
\begin{tikzpicture}
  \begin{axis}[xmin=-1, xmax=2, ymin=0, ymax=8]
    \addplot[blue,domain=-1:2,samples=1000]
      plot (\x,{sin(360*pi *\x)^2});
    \addplot[red,domain=-1:2 ,samples=1000]
      plot (\x,{e^{\x}});
    \addplot[purple,domain=-1:2,samples=1000]
      plot (\x,{e^{\x}*sin(360*pi *\x)^2});
    \addplot[domain=-1:2 ,samples=1000]
      plot (\x,{e^{-\x}});
  \end{axis}
\end{tikzpicture}
```

Obtendremos:

---

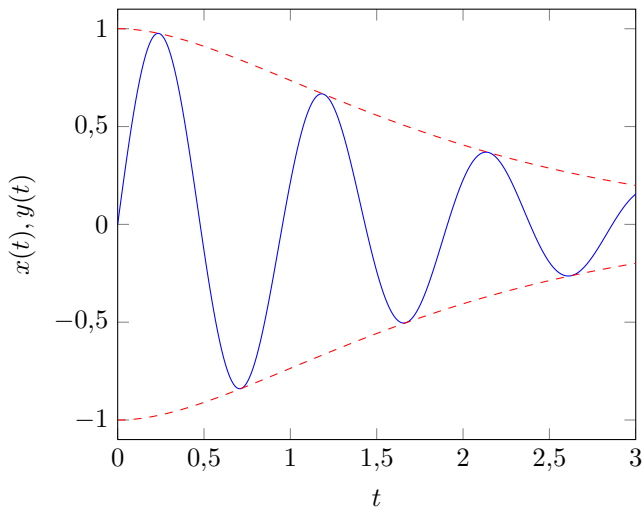


Si incluimos varias gráficas, a veces queda confuso si debemos utilizar el mismo color para todas las funciones –por ejemplo, si vamos a imprimir en impresora en blanco y negro, o editar en un libro, ya que la impresión en color dispara los costes en un orden de magnitud; y en muchas editoriales, ni siquiera es planteable el color fuera de la porta-

da y la contraportada-. Aquí nos puede interesar hacer uso de líneas punteadas. Esto se consigue con `dashed`. Por ejemplo, podemos hacer:

```
\begin{tikzpicture}
  \begin{axis}[xmin=0, xmax=3, ymin=-1.1, ymax=1.1,
    xlabel={t}, ylabel={$x(t)$, $y(t)$}]
    \addplot[blue, domain=0:3, samples=1000]
      plot (\x, {(1 + \x)*exp(-\x) * sin(120* pi * \x)});
    \addplot[red, domain=0:3, , samples=1000, dashed]
      plot (\x, { (1+\x) * exp(-\x) } );
    \addplot[red, domain=0:3, samples=1000, dashed]
      plot (\x, { -(1+\x) * exp(-\x) } );
  \end{axis}
\end{tikzpicture}
```

Obteniendo como resultado:



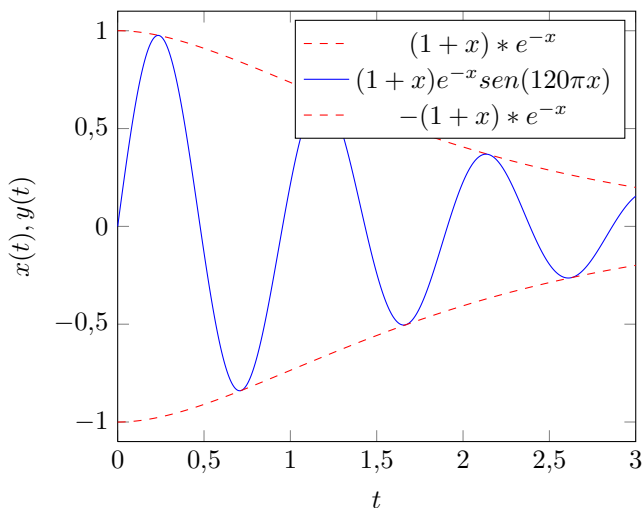
### 10.7.3. Dibujando gráficas a partir de ficheros con volcados de números

Podemos perfectamente incluir los datos a partir de un fichero, en lugar de hacerlo de una función.

Y si alineación es `center` –el valor por defecto–, haciendo esto:

```
\begin{tikzpicture}
  \begin{axis}[xmin=0, xmax=3, ymin=-1.1, ymax=1.1,
    xlabel={t}, ylabel={x(t), y(t)},
    legend cell align=center
  ]
    \addplot[red, domain=0:3, , samples=1000, dashed]
      plot (\x, { (1+\x) * exp(-\x) });
    \addplot[blue, domain=0:3, samples=1000]
      plot (\x, {(1 + \x)*exp(-\x) * sin(120* pi * \x)});
    \addplot[red, domain=0:3, samples=1000, dashed]
      plot (\x, { -(1+\x) * exp(-\x) });
    \legend{$(1+x)*e^{-x}$,
      $(1+x)e^{-x}\text{sen}(120\pi x)$,
      $-(1+x)*e^{-x}$}
  }
\end{axis}
\end{tikzpicture}
```

Obtenemos esto:



Del mismo modo que podemos cambiar la alineación del texto en la leyenda, también podemos cambiar la alineación del cuadro de leyenda respecto del gráfico. Esto lo haremos especificando dentro los parámetros del entorno `axis` el comando:

Cada una de estas líneas corresponde con un único punto, de coordenadas `posX` y `posY`, etiquetado con la etiqueta `etiqueta`. No es necesario que se pongan todos los puntos ordenados por el mismo tipo de etiquetas.

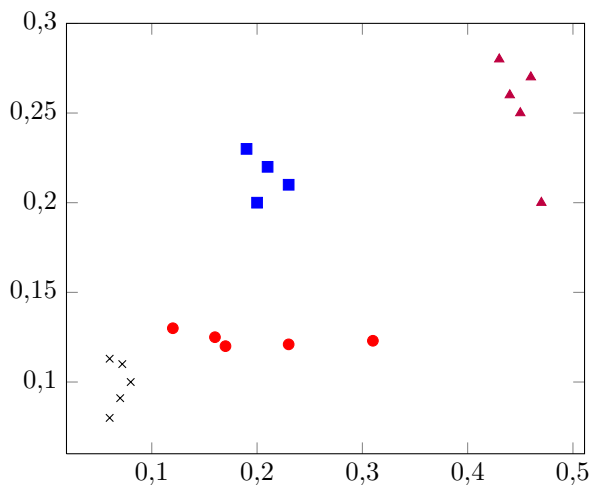
Por ejemplo, si hacemos:

```
\begin{tikzpicture}
\begin{axis}
\addplot [
  scatter,
  only marks,
  point meta=explicit symbolic,
  scatter/classes={
    patos={mark=square*,blue},
    peces={mark=triangle*,purple},
    elefantes={mark=*,red},
    humanos={mark=x,black}
  },
] table [meta=label]
{
  x y label
  0.2 0.2 patos
  0.43 0.28 peces
  0.47 0.20 peces
  0.23 0.21 patos
  0.44 0.26 peces
  0.46 0.27 peces
  0.45 0.25 peces
  0.17 0.12 elefantes
  0.16 0.125 elefantes
  0.12 0.13 elefantes
  0.21 0.22 patos
  0.23 0.121 elefantes
  0.31 0.123 elefantes
  0.06 0.08 humanos
  0.072 0.11 humanos
  0.07 0.091 humanos
  0.19 0.23 patos
  0.06 0.113 humanos
  0.08 0.1 humanos
};
```



```
\end{axis}
\end{tikzpicture}
```

Obtenemos:



Todo lo que se ha dicho en este capítulo respecto a este entorno de ejes, de escalado, y de cambiar propiedades funciona en los gráficos de dispersión, por lo que podemos usar los comandos anteriores para tareas tales como crear leyendas, o ajustar los ejes a nuestro interés estético.

### 10.7.9. Dibujando gráficas en 3D

El dibujo de gráficas 3D es un mundo aún mayor que el de gráficas 2D; se podría escribir un libro sobre lo que se puede llegar a hacer. Y no son de uso mayoritario. Sin embargo, lo interesante es que aprendido lo que sabemos una vez llegamos hasta aquí, no es complicado entender cómo funcionan: aquellas familias de comandos para los que tenemos una versión para la coordenada X, y otra para la coordenada Y, tendrán una función adicional para coordenada Z. Por ello, en lugar de repetir el todo lo que llevamos de `axis` otra vez para la coordenada Z, vamos a poner algunos de los usos ilustrativos más comunes, y vamos a explicar lo que aparece nuevo. Un ejemplo simple de gráfica 3D es:

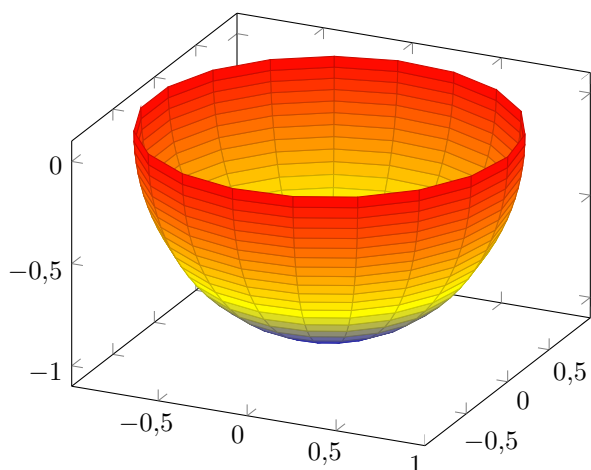
```
\begin{tikzpicture}
\begin{axis}[]
\addplot3 [
    surf,shader=flat,
    samples=20,
    domain=-1:0,
    y domain=0:2*pi,
    z buffer=sort,
] (
    {sqrt(1-x^2) * cos(deg(y))},
    {sqrt(1-x^2) * sin(deg(y))},
    x
);
\end{axis}
\end{tikzpicture}
```

Vemos que utilizamos el comando `\addplot3`, que tiene una sintaxis muy parecida. Que especificamos dos rangos, en lugar de uno –el segundo, lo especificamos con `y domain=`, y añadimos:

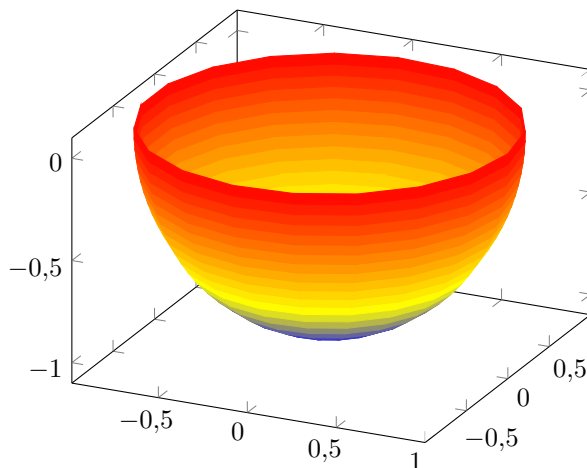
```
surf,
shader=flat,
```

`surf` lo pondremos cuando queramos que nos dibuje la superficie, y `shader=flat` cuando no queramos que aparezcan las líneas entre superficies calculadas. En este caso, obtenemos sin `shader=flat`:

---



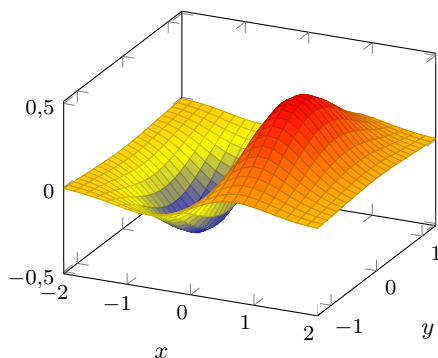
Y con `shader=flat`:



Es interesante ver cómo se rota una gráfica. En principio, una gráfica sin rotar sería:

```
\begin{tikzpicture}
\begin{axis}[xlabel=$x$, ylabel=$y$, small]
\addplot3 [
    surf,
    domain=-2:2,
    domain y=-1.3:1.3,
] {exp(-x^2-y^2)*x};
\end{axis}
\end{tikzpicture}
```

Obtenemos:



# Capítulo 11

## Control de formato y maquetación manual en $\text{\LaTeX 2}_{\epsilon}$

---

*Podemos pensar que “falta poco” por saber de  $\text{\LaTeX 2}_{\epsilon}$ . Y no es así. Hasta ahora hemos visto una introducción a  $\text{\LaTeX 2}_{\epsilon}$ ; pero aún falta muchísimo por ver. Hay mucho material sobre cómo hacer auténticas virguerías con  $\text{\LaTeX 2}_{\epsilon}$ . Hemos estado confiando hasta ahora en este libro en el “maquetador inteligente”. Ahora que sabemos pilotar, igual ya no necesitamos asistencia. Así que el “maquetador inteligente” nos pasa los mandos, y nosotros daremos las órdenes al “maquetador obediente”.*

---

Hasta ahora hemos seguido el modelo que he denominado repetidas veces “de maquettador inteligente”. En este modelo de trabajo, suponemos que  $\text{\LaTeX 2}_{\epsilon}$  es un “maquetador inteligente”; es decir, un buen profesional, que sabe lo que hace, que va a dar un formato coherente y correcto a todo lo que hace, según unos estándares preestablecidos y aceptados por la industria.

Y si bien esto suele ser así, a veces el modelo de “maquetador inteligente” no nos es adecuado: o porque la forma de generar el documento no concuerda con las normas de maquetación de nuestra universidad o empresa, porque el formato no coincide con el que queremos entregar, o simplemente porque no nos gusta la estética.

En este capítulo, vamos a cambiar el modelo. Vamos a aprender a dar órdenes a L<sup>A</sup>T<sub>E</sub>X 2<sub>ε</sub> sobre cómo maquetar –hasta ahora, le hemos dicho qué maquetar, no cómo hacerlo–. Ahora nuestro maquetador no es tan listo, no nos satisface su trabajo, y le vamos a dar órdenes puntuales de cómo queremos las cosas.

Dentro de estas órdenes de formato, tenemos dos opciones: las de preámbulo, o las que “empotramos” en el texto. Las de preámbulo nos permiten modificar prácticamente sin trabajo documentos enteros de cientos de páginas: por ejemplo, yo tengo las órdenes de preámbulo para el formato de trabajos fin de grado de la Escuela Técnica Superior de Ingenieros en Informática; por lo que aquellos alumnos que les oriento<sup>1</sup> el trabajo fin de grado ignoran el formato de lo que hacen hasta el último día: le pegamos la cabecera, recompilamos un par de veces, y a imprimir.

Para los puntos donde excepcionalmente queremos conseguir algún efecto, tenemos órdenes puntuales de formato. Podemos hacerlo –vamos a explicarlo–; pero también podemos crear nuestros propios entornos, para conseguir efectos impresionantes con poco trabajo.

Hay algo que debes saber: tanto todos mis libros autopublicados, como el material de todos los cursos que imparto, están hechos íntegramente en L<sup>A</sup>T<sub>E</sub>X 2<sub>ε</sub>. También redacto mis informes periciales, mis informes de auditoría, mis presentaciones, y mis proyectos en L<sup>A</sup>T<sub>E</sub>X 2<sub>ε</sub>. Absolutamente todo lo que he utilizado para hacer cualquiera de los libros, cualquiera de las audiorías, cualquiera de las consultorías o cualquiera los materiales que yo he creado en los últimos 20 años lo he hecho utilizando L<sup>A</sup>T<sub>E</sub>X 2<sub>ε</sub>; y absolutamente todos los “trucos” que he utilizado los últimos 20 años, están explicados en este libro. Muchos de los efectos que ves los he conseguido creando mis propios entornos y comandos, de forma que he podido después reajustar todo desde cabecera para que quede bien. Lo que te falta por aprender, lo vas a aprender ahora.

En este capítulo también vas a ver algunos entornos interesantes para hacer cosas específicas –incluir código fuente de programas con una buena estética, dependiente del lenguaje de programación; o incluir fórmulas químicas, por poner algunos ejemplos–. Espero que con todo esto obtengas los conocimientos que necesitas para editar correctamente el

---

<sup>1</sup>Nunca me ha gustado la palabra “dirigir” relativa a lo que hago, yo no dirijo nada. En EEUU se habla de “advisor” –consejero– y en Brasil de “orientador”, lo que me parece una definición más correcta de mi papel.

material que tenías en mente cuando adquiriste este libro: sea un trabajo de clase, un trabajo fin de grado, una tesis doctoral, o la documentación que le quieres mandar a un cliente.

De cualquier forma, es importante que tengas claras dos ideas: la primera, que el hecho de que **puedas** hacer algo no significa que **ten-gas** que hacerlo. Mantén limpio tu documento de “efectos especiales”; procura tener una maquetación simple y clara, y no “enloquecer” añadiendo fuentes y colorines –aprenderemos también ahora a hacer estas dos cosas: añadir fuentes distintas y colores distintos a nuestro material en L<sup>A</sup>T<sub>E</sub>X 2<sub>ε</sub>–.

La segunda idea es que siempre es mejor ajustar el formato en cabecera que hacerlo con microajustes donde algo no te gusta. Y siempre es mejor hacer los efectos –como el párrafo introductorio de cada capítulo de este libro– mediante entornos y comandos que tú mismo crees, que copiar y pegar un mismo código de maquetación decenas de veces por el documento.

Comentado todo esto, entramos en faena.

## 11.1. Tomando el control del formato de nuestro documento

### 11.1.1. Definiendo cabeceras

Lo primero que podemos querer cambiar de una página son las cabeceras y los pies de páginas.

Hay muchas formas de hacer esto; pero la más fácil es haciendo uso del paquete `fancyheadings`, que nos permitirá hacer uso de varios estilos de cabecera y de pie de página. Incluimos el paquete de `fancyheadings` haciendo:

```
\usepackage{fancyheadings}
```

Existe una versión más reciente de este paquete, denominada `fancyhdr`, que podemos incluir con:

Por ejemplo, si incluimos esto en la cabecera de un texto *Lorem Ipsum* con la clase `book`:

```
\usepackage{fancyhdr}
\pagestyle{fancyplain}
\lhead[Par (\thepage), arriba, izquierda]
      {Impar (\thepage), arriba, izquierda}
\chead[Par, arriba, centrado] {Impar, arriba, centrado}
\rhead[Par, arriba, derecha] {Impar, arriba, derecha}
\lfoot[Par (\thepage), abajo, izquierda]
      {Impar (\thepage), abajo, izquierda}
\cfoot[Par, abajo, centrado] {Impar, abajo, centrado}
\rfoot[Par, abajo, derecha] {Impar, abajo, derecha}
```

Obtendremos esta cabecera de página impar:

---

Impar(1),arriba,izquierda	Impar,arriba,centrado	Impar,arriba,derecha
---------------------------	-----------------------	----------------------

---

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Mauris eu euismod metus. Aliquam imperdiet massa leo, eu elementum leo dictum sit amet. Aenean laoreet ac purus eget sagittis. Curabitur vel elit augue. Ut quis placerat purus. Morbi lacinia, nibh ac imperdiet dignissim, ligula dui faucibus nibh, vel ullamcorper tortor lorem at massa. Maecenas aliquet suscipit nunc quis sagittis. Fusce ipsum magna, convallis non facilisis ac, venenatis vitae magna. Nunc gravida placerat finibus. Nulla facilisi. Phasellus consequat lobortis blandit. Donec quis malesuada nisi, eget laoreet ex. Phasellus placerat dolor in imperdiet ornare.

---

Donec eget ante ut ex mollis eleifend at vel felis. Aliquam mollis tortor

Este pie de página impar:

---

cassiodorus ut, sagittis elit.
--------------------------------

---

Suspendisse potenti. Suspendisse iaculis, erat elementum accumsan efficitur, elit nisl imperdiet ante, in fermentum orci neque vel turpis. Morbi nisi elit, sollicitudin porttitor mi quis, volutpat fermentum nibh. Aliquam erat volutpat. Etiam lacinia non felis nec tristique. Sed pellentesque quam eget arcu commodo laoreet. Morbi dui ex, varius et metus non, fringilla auctor purus. Suspendisse potenti. Morbi velit sem, egestas convallis ultricies id, semper at quam.

---

Impar(1),abajo,izquierda	Impar,abajo,centrado	Impar,abajo,derecha
--------------------------	----------------------	---------------------

---

También podemos cambiar estos parámetros mediante `renewcommand`; por ejemplo,

```
\renewcommand{\headrulewidth}{tamaño}
```

Y del pie de página con:

```
\renewcommand{\headrulewidth}{tamaño}
```

Las medidas las definimos con unidades, tal y como hemos estudiado en la página 113 de este mismo libro.

Por ejemplo, en este documento podría hacer lo mismo haciendo uso del comando `renewcommand`:

```
\renewcommand{\headrulewidth}{.8pt}
```

Que es equivalente a hacerlo con `setlength`.

### 11.1.2. El formato de página, a bajo nivel

Los márgenes es una de las cosas más importantes en la edición de textos, ya que muchas veces nos obligan a tener determinado formato de página en nuestros documentos. En L<sup>A</sup>T<sub>E</sub>X 2<sub>ε</sub> los márgenes vienen predefinidos por el estilo y por el tamaño de la página, pero este valor puede ser sobreescrito. Los comandos para modificar los márgenes son:

- `\voffset`: Desplazamiento de la página desde el extremo superior de esta antes de comenzar la impresión. Es mejor no tocar este parámetro; y para mover el margen superior modificar `\topmargin`; que hace lo mismo. Suele valer 0.
- `\topmargin`: El tamaño del margen superior será el valor de este parámetro, añadiéndole una pulgada y el valor de `\voffset`. Puede ser negativo, para ganar espacio a la pulgada y el desplazamiento. Este es el parámetro que debemos modificar para ajustar el margen superior.



medio centímetro de sangría francesa se lo debemos añadir a los márgenes. Esto lo haremos añadiendo al preámbulo:

```
\topskip=40pt
\parskip=1cm
\parindent=-0.5cm
\baselineskip=16pt
\addtolength{\oddsidemargin}{0.5cm}
\addtolength{\evensidemargin}{0.5cm}
```

Quedando:

---

2

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Mauris eu euismetus. Aliquam imperdiet massa leo, eu elementum leo dictum sit ar Aenean laoreet ac purus eget sagittis. Curabitur vel elit augue. Ut q placerat purus. Morbi lacinia, nibh ac imperdiet dignissim, ligula dui f cibus nibh, vel ullamcorper tortor lorem at massa. Maecenas aliquet s cipit nunc quis sagittis. Fusce ipsum magna, convallis non facilisis ac, nenatis vitae magna. Nunc gravida placerat finibus. Nulla facilisi. Phas lus consequat lobortis blandit. Donec quis malesuada nisi, eget laoreet Phasellus placerat dolor in imperdiet ornare.

Donec eget ante ut ex mollis eleifend at vel felis. Aliquam mollis tortor ante laoreet, in congue erat congue. Mauris consectetur, neque non venatis euismod, est elit vulputate nibh, sed luctus orci nibh non nunc. 5 vitae dolor vel ligula tempus feugiat molestie in purus. Quisque orci n sodales sed aliquet eu, euismod nec felis. In hac habitasse platea dictum Sed convallis lacus in turpis cursus, eget placerat dolor vulputate. Duis gue eros, consequat quis elementum nec, semper eget tellus. Nullam ante accumsan, varius massa ut, laoreet enim. Pellentesque eget finit leo, eu semper nisi. Orci varius natoque penatibus et magnis dis parturimontes, nascetur ridiculus mus. Sed at vulputat ipsum. Praesent sit ar metus sit amet purus dignissim eleifend ac eget arcu. Vivamus phase risus eget vestibulum iaculis.

Nulla maximus rutrum dolor, convallis faucibus lectus fringilla sed. Integ dapibus lacus ac massa gravida elementum sed id tellus. Nam vel frin lla neque, eu eleifend tortor. Integer tristique dignissim ante a tincidunt Phasellus interdum, ligula sodales blandit dapibus, nisl ligula congue c lor, quis dictum mauris dui eget sapien. Morbi dignissim augue vel lec consequat, at maximus est condimentum. Nullam congue maximus non facilisis. Nam accumsan aliquet est sit amet vehicula. Proin orci nu sagittis ut turpis quis, hendrerit egestas quam. Donec ac ultrices dolor, vulputate urna. Sed bibendum bibendum arcu tincidunt tempor. Mau rutrum nisi vitae elit ornare, at dignissim leo aliquam. Proin ut lacu lorem pellentesque malesuada. Vestibulum vehicula tellus vulputat, et mod lectus ut, sagittis elit.

Suspendisse potenti. Suspendisse iaculis, erat elementum accurelit nisl imperdiet ante, in fermentum orci neque vel turpis elit, sollicitudin porttitor mi quis, vulputat fermentum nibh. vulputat. Etiam lacinia non felis nec tristique. Sed pellentesq arcu commodo laoreet. Morbi dui ex, varius et metus non, fr purus. Suspendisse potenti. Morbi velit sem, egestas convall semper at quam.

Duis lacinia pellentesque tortor, eu varius leo vulputate eget. D gilla urna, sit amet rhoncus leo. Nunc at elit nec tellus lobor eget nisi. Pellentesque id elit vitae metus congue luctus in sex tortor mauris, elementum quis suscipit eu, eleifend a mi. Su ficitur tortor id lacus commodo, semper elementum neque p ris aliquam, est ac vulputat ornare, urna sem scelerisque diam sem sapien id arcu. Sed iaculis libero at felis vulputate, id p mollis.

Phasellus ullamcorper ligula sed interdum condimentum. Ves ipsum primis in faucibus orci luctus et ultrices posuere cubil in nunc quis justo iaculis fringilla. Praesent mollis purus eu v tincidunt. Vestibulum ante ipsum primis in faucibus orci luc posuere cubilia Curae; Morbi hendrerit nibh sed vulputat effi trices velit sit amet libero consectetur maximus non ut velit. 3 ullamcorper, euismod nibh et, dictum nisl. Quisque interdum auctor sodales, erat justo elementum ante, a rutrum augue Pellentesque hendrerit augue sed nibh sagittis, vitae auctor lo Maecenas et est non leo viverra pulvinar nec ut risus.

Aliquam bibendum lectus in sodales ullamcorper. Nullam finibudales. Ut porta lectus vitae ante semper, sit amet rhoncus eli Etiam nisl ipsum, luctus quis dapibus nec, elementum a ur arcu est. Curabitur sit amet vulputat massa, ut luctus urna. 1 dum tortor vel mi varius euismod. Praesent iaculis dolor neq mentum sapien facilisis rhoncus. Curabitur et magna vel ip

Todos estos espaciados afectan también a las listas, aunque de forma un poco más compleja.

Veamos primero los espaciados verticales de las listas.

El salto del texto de anterior de la lista al primer elemento de la lista es la suma de `\topsep`, `\parskip` y `\partopsep`.

De igual forma, el salto del final del último elemento de la lista al principio del texto inmediatamente posterior a la lista también es la suma de `\topsep`, `\parskip` y `\partopsep`.

Obtendremos:

---

Primer párrafo, salto pequeño

Segundo párrafo, salto mediano

Tercer párrafo, salto grande

Texto final

---

El tamaño de estos saltos vendrá determinado por la clase, el espacio entre líneas, y otros factores. Pero a veces no queremos esto, sino queremos un control fino de salto de párrafo. Aquí tendremos también un comando L<sup>A</sup>T<sub>E</sub>X 2<sub>ε</sub> y un comando T<sub>E</sub>X; pero, a diferencia del salto en horizontal, aquí sí tendrán un comportamiento distinto.

Podemos hacer un salto en vertical utilizando el comando:

```
\vspace{tamaño}
```

Salta exactamente el tamaño especificado en vertical; dónde `tamaño` viene especificado con sus unidades, tal y como estudiamos en la página 113 de este libro. Por ejemplo, si hacemos:

```
Sin salto.
```

```
Ahora vamos a saltar 1 centímetro.
```

```
\vspace{1cm}
```

```
Y lo hemos saltado.
```

Obtendremos:

---

Sin salto.

Ahora vamos a saltar 1 centímetro.

Y lo hemos saltado.

---

Podemos dar saltos hacia atrás, utilizando valores negativos. Esto podemos emplearlo para acortar espacios indeseados, pero es fácil que terminemos “pisando texto”; por ejemplo, si hacemos:

Obtendremos:

---

Ahacer una pruebaBde cómo recolocar textoC

---

Tenemos otro comando interesante, que nos permite hacer escalados locales de trozos de texto. Este será:

```
\scalebox{factor}{texto}
```

Donde `factor` es un factor de escalado de `texto`, sin unidades. Por ejemplo, si hacemos:

```
A\scalebox{1.2}{hacer una prueba}B  
\scalebox{.8}{de cómo redimensionar localmente texto}C
```

Obtendremos:

---

Ahacer una pruebaBde cómo redimensionar localmente textoC

---

Podemos utilizar estos dos comandos a la vez para hacer cosas como:

```
X\hskip -1mm  
$\scalebox{1.1}{\raisebox{2pt}{$\Upsilon$}}$  
\hskip -1mm M\TeX
```

Que generará:

---

X<sup>Υ</sup>M<sub>TE</sub>X

---

### 11.1.5. Modificando títulos y nombres de secciones

El espaciado no es la única cosa que podemos modificar. También podemos modificar el formato de los títulos y los nombres de las secciones.

```
\titleformat{\chapter}[display]{\bfseries\Large\itshape}
    {\thechapter}{0.5ex}
    {\rule{\textwidth}{2pt}}
    \vspace{1ex}
    \centering
}
[\vspace{-0.5ex}
\rule{\textwidth}{0.5pt}
]
```

Tendremos esto:

---

**1**

---

*Lorem ipsum dolor sit amet*

---

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Mauris eu euismod metus. Aliquam imperdiet massa leo, eu elementum leo dictum sit amet. Aenean laoreet ac purus eget sagittis. Curabitur vel elit augue. Ut

---

Hemos dejado a un lado de momento ejemplos del tipo de encabezamiento; vamos a pasar a ver primero aquellos que no requieren forzosamente especificar el espaciado del título.

Del tipo `display`, ya hemos visto un ejemplo.

Respecto al tipo `hang`, si añadimos esto:

```
\titleformat{\chapter}[hang]{\bfseries\Large\itshape}
    {Capítulo \thechapter:}{0.5ex}
    {}[]
```

Obtendremos esto:

---

## *Capítulo 1: Lorem ipsum dolor sit amet*

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Mauris eu euismod metus. Aliquam imperdiet massa leo, eu elementum leo dictum sit amet. Aenean laoreet ac purus eget sagittis. Curabitur vel elit augue. Ut quis placerat purus. Morbi lacinia, nibh ac imperdiet dignissim. ligula dui

---

Respecto al tipo `frame`, haciendo esto:

```
\titleformat{\chapter}[frame]{\bfseries\Large\itshape}
           {Capítulo \thechapter}{0.5ex}
           {\vskip -3mm\centering}[]
```

Obtenemos esto:

---

## *Capítulo 1*

### *Lorem ipsum dolor sit amet*

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Mauris eu euismod metus. Aliquam imperdiet massa leo, eu elementum leo dictum sit amet. Aenean laoreet ac purus eget sagittis. Curabitur vel elit augue. Ut quis placerat purus. Morbi lacinia, nibh ac imperdiet dignissim. ligula dui

---

Los tipos restantes requieren indicar el espaciado del título. Esto se hace con el comando:

```
\titlespacing{comandotiposección}{margenzquierdo}
           {margensuperior}{margenposterior}
           {margenderecho}
```

Donde:

- `comandotiposección` es comando del tipo de la sección que queremos alterar; puede ser `chapter`, `section`, `subsection`, por poner ejemplos.

- `margenizquierdo` es el parámetro del margen a la izquierda del título de sección; y su significado exacto varía según el tipo de encabezamiento. En `runin` es la sangría antes del título, en `drop`, `leftmargin` y `leftmargin`, es el ancho del título, en `wrap`, el ancho máximo del título. En todos los demás, es un incremento del margen izquierdo.
- `margensuperior` es el margen que se deja encima del título.
- `margenposterior` es el margen que se deja después del título y antes del texto de la sección. En tipos como `hang`, `block` o `display`, que el texto está debajo del título, corresponde a un margen vertical, medido entre la parte baja del título y la parte alta del texto. En tipos como `runin`, `drop`, `wrap`, `leftmargin` o `rightmargin`, que el texto se encuentra a la derecha del título de la sección, corresponde al margen horizontal del lado derecho del título, medido entre la parte más a la derecha del título y la parte más a la izquierda del texto.
- `margenderecho` corresponde con un parámetro opcional para determinar el margen horizontal derecho en los tipos `hang`, `block` o `display`.

Todos estos valores pueden ser positivos –aumentando el margen– o negativos –reduciendo el espacio que ya deje L<sup>A</sup>T<sub>E</sub>X 2<sub>ε</sub>; por ejemplo, en márgenes verticales, por el salto de párrafo–. Todos estos valores también deben estar con sus unidades, tal como explicamos éstas en la página 113 de este libro.

Podemos utilizar el comando `\titlespacing` en principio en cualquiera de los tipos de título; pero ojo: siempre que hayamos incluido un comando `\titleformat` en el preámbulo anteriormente.

Ahora que conocemos el comando `\titlespacing` podemos ver algunos ejemplos de su uso. Estos ejemplos los vamos a ver sobre el tipo de sección `section` en lugar de sobre `chapter`; pero reiterando que en principio estos dos comandos y todos los tipos pueden ser empleados sobre cualquiera de las secciones de un texto en L<sup>A</sup>T<sub>E</sub>X 2<sub>ε</sub>.

El ejemplo del tipo `leftmargin` sería:

```
\titleformat{\section}[leftmargin]
               {\bfseries\Large\itshape}
               {\S \thesection . }{.5em}{}[]
\titlespacing{\section}{6pc}{1.6ex}{1pc}
```

Obteniendo:

---

§3.2.  *Lorem*    Lorem ipsum dolor sit amet, consectetur adipiscing elit. Mauris eu euismod metus. Aliquam imperdiet massa leo, eu elementum leo dictum sit amet. Aenean laoreet ac purus eget sagittis. Curabitur vel elit augue. Ut quis placerat nunc. Morbi lacinia nibh ac imperdiet dionissim. litora dui

---

Finalmente, el ejemplo del tipo `rightmargin` sería:

```
\titleformat{\section}[rightmargin]
               {\bfseries\Large\itshape}
               {\S \thesection . }{.5em}{}[]
\titlespacing{\section}{6pc}{1.6ex}{1pc}
```

Siendo el resultado:

---

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Mauris eu euismod metus. Aliquam imperdiet massa leo, eu elementum leo dictum sit amet. Aenean laoreet ac purus eget sagittis. Curabitur vel elit augue. Ut quis placerat nunc. Morbi lacinia nibh ac imperdiet dionissim. litora dui

§3.2.  *Lorem ipsum*

---

El paquete `titlesec` es muy completo, y se dispone de mucha más información sobre él aquí:

<https://www.ctan.org/pkg/titlesec?lang=en>

## 11.1.6. Definiendo nuestros propios identificadores de secciones

También nos podemos definir nuestros propios identificadores de secciones en el preámbulo. La función para hacer esto es:

```
\@startsection{nombre}{nivel}{indentación}
               {saltoanterior}{saltoposterior}{estilo}
```

eget laoreet ex. Phasellus placerat dolor in imperdiet ornare.

Obtendremos:

---

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Mauris eu euismod metus. Aliquam imperdiet massa leo, eu elementum leo dictum sit amet. Aenean laoreet ac purus eget sagittis. Curabitur vel elit augue. Ut quis placerat purus. Morbi lacinia, nibh ac imperdiet dignissim, ligula dui faucibus nibh, vel ullamcorper tortor lorem at massa. Maecenas aliquet suscipit nunc quis sagittis. Fusce ipsum magna, convallis non facilisis ac, venenatis vitae magna. Nunc gravida placerat finibus. Nulla facilisi. Phasellus consequat lobortis blandit. Donec quis malesuada nisi, eget laoreet ex. Phasellus placerat dolor in imperdiet ornare.



---

A la hora de utilizar este paquete, debemos recordar que no hace milagros. Los flotantes grandes siempre son problemáticos de maquetar –en L<sup>A</sup>T<sub>E</sub>X 2<sub>ε</sub> y en cualquier otro sistema–; y si utilizamos `wrapfigure`, al problema que ya teníamos con flotantes “altos” tenemos que añadir el problema de los flotantes “anchos”. No podemos pretender que con una fuente de tamaño 12 dejemos dos centímetros para texto al lado del flotante, y que quede bien. No lo hará. Debemos dejar espacio suficiente a la izquierda o a la derecha del flotante rodeado para que se pueda maquetar de forma estética una línea con el tamaño de fuente que usamos en el documento.

### 11.2.2. Revisitando las listas

Las listas las hemos revisitado ya anteriormente. Dando por válido cómo L<sup>A</sup>T<sub>E</sub>X 2<sub>ε</sub> las deja desde el punto de vista estético. Pero se pueden personalizar para que se adapten a nuestras necesidades.



### 11.2.2.1. Listas compactas

De las distintas cosas que se necesitan normalmente de las listas en L<sup>A</sup>T<sub>E</sub>X 2<sub>ε</sub> es que puedan hacerse más compactas; ya que cuando la lista es de palabras sueltas, el espaciado estándar queda poco estético.

Por ejemplo, una lista normalmente queda así:

---

Son mamíferos, los:

- Perros
  - Gatos
  - Humanos
  - Delfines
- 

Hay un excelente paquete, denominado `enumitem`, que nos permite un control muy fino de la apariencia que tienen las listas. Añadimos este paquete con:

```
\usepackage{enumitem}
```

Si queremos obtener un resultado más compacto, añadimos como opción de `itemize` el modificador `noitemsep`; por ejemplo, si hacemos:

```
Son mamíferos, los:
\begin{itemize}[noitemsep]
\item Perros
\item Gatos
\item Humanos
\item Delfines
\end{itemize}
```

Quedará:

---

Son mamíferos, los:

- Perros
  - Gatos
  - Humanos
  - Delfines
-

```
& cada uno & 0'01{\euro} \\
\addlinespace[1mm]
Piedra& por kilo & 1'50{\euro} \\
\addlinespace[1mm]
Agua & por litro & 0'80{\euro} \\
& por m$^3$ & 765'00{\euro} \\
\bottomrule[2pt]
\end{tabular}
```

Resultando:

Producto		
Nombre	Unidad	Precio
Papel	por resma	8'23€
	cada uno	0'01€
Piedra	por kilo	1'50€
Agua	por litro	0'80€
	por m <sup>3</sup>	765'00€

11.2.4.1. Tablas apaisadas

La inclusión de tabla apaisada es tan sencilla como encapsular la tabla en un entorno `sidewaystable`.

Esto es gracias a un paquete, denominado `rotating`, y que incluimos con:

```
\usepackage{rotating}
```

Este paquete nos crea varios entornos nuevos. Estos son:

- `sideways`: rota un objeto arbitrario 90 grados. Normalmente encapsulamos lo que queremos rotar en un `minipage`, encapsulación que veremos más adelante en este mismo capítulo.
- `turn`: rota un objeto arbitrario los grados que se le añade como parámetro al entorno, con la sintaxis `\begin{turn}{grados}`, añadiendo el espacio necesario para que el texto no “pise” el objeto rotado. También lo utilizamos con `minipage` normalmente.

```
uno & dos & tres \\
uno & dos & tres \\
uno & dos & tres \\
uno & dos & tres \\
uno & dos & tres \\
\hline
\end{tabular}
```

Recordando que el `\definecolor` puede estar definido en el cuerpo del documento con anterioridad al uso del `color`, o en la cabecera del documento. El resultado para un 90 % de gris será:

---

uno	dos	tres
uno	dos	tres
uno	dos	tres
uno	dos	tres
uno	dos	tres

---

Y para un 95 % de gris será:

---

uno	dos	tres
uno	dos	tres
uno	dos	tres
uno	dos	tres
uno	dos	tres

---

### 11.2.4.3. Tablas inmensas

Finalmente llegamos al gran problema: las tablas inmensas. Tablas que ocupan varias páginas, que queremos que se corten entre páginas, y que además queremos que cada trozo cortado de tabla tenga una cabecera y un pie de bloque específico.

La forma que tendremos de resolver este problema es mediante el uso del paquete `longtable`, que incorporamos con:

```
\usepackage{longtable}
```

Este paquete nos dota del nuevo entorno `longtable`. Este entorno son tablas con las que podemos hacer lo mismo que hemos aprendido

```
\endlastfoot
uno & dos & tres \\
uno & dos & tres \\
uno & dos & tres \\
uno & dos & tres \\
```

Suponiendo que tenemos en nuestra tabla muchas más filas que contengan `uno & dos & tres \\`, y que nuestra tabla termina con:

```
uno & dos & tres \\
uno & dos & tres \\
uno & dos & tres \\
\end{longtable}
```

Obtendremos en la cabecera de la primera página de la tabla:

---

Primero	Segundo	Tercero
uno	dos	tres
uno	dos	tres
uno	dos	tres
uno	dos	tres
uno	dos	tres

---

Este pie de todas las páginas de la tabla, salvo la última página:

---

uno	dos	tres
uno	dos	tres
uno	dos	tres
uno	dos	tres
uno	dos	tres
Primero	Segundo	Tercero
Próxima página...		

---

En la cabecera de todas las páginas de la tabla, salvo la primera página:

---

...Página anterior		
Primero y segundo		Tercero
uno	dos	tres
uno	dos	tres
uno	dos	tres
uno	dos	tres
uno	dos	tres

---

```

D & E & F \\
\hline
G & H & I \\
\end{tabular}
\end{minipage}

```

Resultando:

1	2	3	A	B	C
4	5	6	D	E	F
7	8	9	G	H	I

Para que entendamos lo que estamos haciendo, y reiterando la potencia de este sistema. El código L<sup>A</sup>T<sub>E</sub>X 2<sub>ε</sub>:

```

\begin{minipage}[t]{0.4\textwidth}
Lorem ipsum\footnote{Esto es un pie de página en minipage}
y más texto.
\end{minipage}
\begin{minipage}[t]{0.1\textwidth}
\includegraphics[width=\textwidth]{escudo.png}
\end{minipage}
\begin{minipage}[t]{0.4\textwidth}
Y otro texto\footnote{Esto es otro pie de página en
minipage} y más texto.
\end{minipage}

```

Nos genera lo siguiente:

Lorem ipsum<sup>a</sup> y más texto.

---

<sup>a</sup>Esto es un pie de página en minipage



Y otro texto<sup>a</sup> y más texto.

---

<sup>a</sup>Esto es otro pie de página en minipage

Es decir, que cada `minipage` se comporta como una página independiente; que puede hasta tener sus propios pies de página. Es importante observar que el comando `footnote` dentro del `minipage` no pone los pies de página al pie de la página dónde aparece, sino al pie

de la `minipage` dónde aparece, como si el documento que contiene la `minipage` no existiera.

## 11.4. Depurando en L<sup>A</sup>T<sub>E</sub>X 2<sub>ε</sub>

### 11.4.1. Operadores y comandos frágiles

Vamos a comentar un concepto que mencioné lateralmente en el tema ecuaciones en L<sup>A</sup>T<sub>E</sub>X 2<sub>ε</sub>. Se trata de los operadores y los comandos frágiles.

Los operadores y los comandos frágiles son aquellos operadores y comandos de L<sup>A</sup>T<sub>E</sub>X 2<sub>ε</sub> que pueden fallar si aparecen como argumentos de otro comando<sup>9</sup>. Hay algunos comandos catalogados como frágiles, y que por ello no funcionarán en títulos de sección, ni en pies de página, por poner ejemplos. Esto significa que es mejor usar sustitutos no frágiles. Por ejemplo, si hacemos:

```
\section{Demostración de que  $\forall b \mid b \in \mathbb{R}$ 
    } b < a \rightarrow 0 < b < a < \varepsilon}
```

La línea fallará, ya que estamos usando los comandos frágiles `\"` y `\"` dentro del argumento del comando `\"section\"`. Por otro lado, si queremos hacer lo mismo pero que funcione, hacemos:

```
\section{Demostración de que  $\forall b \mid b \in \mathbb{R}$ 
    } b < a \rightarrow 0 < b < a < \varepsilon}
```

Que genera como salida:

---

**12.3. Demostración de que  $\forall b \mid b \in \mathbb{R} b < a \Rightarrow 0 < b < a < \varepsilon$**

---

<sup>9</sup>La explicación de porqué fallan los comandos frágiles es bastante arcana. L<sup>A</sup>T<sub>E</sub>X 2<sub>ε</sub> genera código intermedio en T<sub>E</sub>X, y los comandos frágiles son comandos que cuando los pones como argumentos de otro comando, el código T<sub>E</sub>X intermedio que genera L<sup>A</sup>T<sub>E</sub>X 2<sub>ε</sub> no compila en T<sub>E</sub>X. Esto ocurre con relativa frecuencia con comandos “móviles”. Son comandos que realmente generan información en el fichero `aux`, para que L<sup>A</sup>T<sub>E</sub>X 2<sub>ε</sub> lea esa información en una pasada posterior. De cualquier forma, esta explicación técnica precisa no te debe preocupar, ya que no afectará ni a cuando se te plantea el problema, ni a cómo lo vas a solucionar.

Si queremos que lea de consola un número, le calcule el factorial, e imprima el resultado en pantalla, hacemos:

```
\documentclass[spanish,twoside,a4,11pt]{book}
\usepackage[utf8x]{inputenc}
\usepackage{forloop}
\usepackage{calc}
\begin{document}
\newcounter{ac}
\setcounter{ac}{1}
\newcounter{inc}
\newcounter{fin}
\typein[\entr]{Introduzca valor a calcular su factorial:}
\setcounter{fin}{\entr}
\typeout{\thefin}
\stepcounter{fin}
\forloop{inc}{1}{\value{inc} < \value{fin}}
{
\setcounter{ac}{\value{ac} * \value{inc}}
}
\typeout{\theac}
\end{document}
```

Así puesto, parece que tiene poco uso; pero recordemos que lo podemos integrar dentro de  $\text{\LaTeX}2_{\epsilon}$ . Por ejemplo, en el ejemplo ya estudiado de `longtable`, en lugar de repetir 70 veces la línea que aparece en cada fila uno & dos & tres `\`, podemos hacer así:

```
\definecolor{gray90}{gray}{.90}
\rowcolors{0}{}{gray90}
\begin{longtable}{p{3cm} p{3cm} | p{3cm}}
\hline
Primero & Segundo & Tercero \\\hline
\endfirsthead
\hline
\multicolumn{2}{1}{\dots Página anterior}& \\\hline
\multicolumn{2}{1}{Primero y segundo}& Tercero \\\hline
\endhead
```