

LyX, la otra forma de escribir

Edición 2022

Una de las mejores herramientas para trabajar
con el editor de documentos más potente

RICARDO G. BERLASSO

$\text{D}\grave{\text{o}} \text{ documentclass } y \text{ entra}$

@
@@@ @ @@@
X₃TEX
E₃TEX

LyX



<https://elpinguinotolkiano.wordpress.com/>

\LaTeX y derivados como $X\TeX$ tienen la enorme ventaja de su potencia y el gran problema, al menos en su forma pura, de su difícil *curva de aprendizaje*.

El proyecto LyX resuelve la dificultad de utilizar \LaTeX sin comprometer el poder del sistema, regalándonos un programa robusto y simple de utilizar que satisfará tanto a usuarios noveles como a veteranos.

En este libro se explora el poder de LyX , \LaTeX , $X\TeX$ y OpenType, mostrando hasta dónde se puede llegar.

LyX, la otra forma de escribir

*D*í `documentclass` *y* *entra*

Ricardo Berlasso

Edición 2022

© 2017-2022 Ricardo Gabriel Berlasso

Esta obra se distribuye bajo licencia Creative Commons Atribución-CompartirIgual 4.0 Internacional (CC BY-SA 4.0)

(<http://creativecommons.org/licenses/by-sa/4.0/>)



ATRIBUCIÓN: Usted debe darle crédito a esta obra de manera adecuada, proporcionando un enlace a la licencia, e indicando si se han realizado cambios. Puede hacerlo en cualquier forma razonable, pero no de forma tal que sugiera que usted o su uso tienen el apoyo del licenciente.



COMPARTIR IGUAL: Si usted mezcla, transforma o crea nuevo material a partir de esta obra, usted podrá distribuir su contribución siempre que utilice la misma licencia que la obra original.

NO HAY RESTRICCIONES ADICIONALES: Usted no puede aplicar términos legales ni medidas tecnológicas que restrinjan legalmente a otros hacer cualquier uso permitido por la licencia.

Cualquier marca mencionada en este libro pertenece a su respectivo propietario.

Los enlaces a sitios web de terceros se proporcionan de buena fe y sólo a título informativo. El autor declina toda responsabilidad por los materiales contenidos en cualquier sitio web de terceros al que se haga referencia en este trabajo.

La última versión de este libro se encuentra siempre en la siguiente dirección:

<https://elpinguinoTolkiano.wordpress.com/latexetexlyx/>

Índice general

I	Los cimientos	1
1	Introducción	3
1.1	Para quién es este libro	3
1.2	La estructura de este libro	6
1.3	Un poco de historia	6
1.4	Bases mínimas	7
1.5	Convenciones utilizadas en este libro	8
1.6	Enlaces a tener a mano	9
2	Definiendo \LaTeX	11
2.1	Qué es \LaTeX	11
2.2	\LaTeX como lenguaje de etiquetas (o casi...)	13
2.3	Clases de documentos	16
2.4	La construcción del documento	17
2.5	Y <i>Open Type</i> llegó a \LaTeX	17
3	Qué es lo que hace que LyX sea tan especial	19
3.1	El concepto «WYSIWYM»	21
3.2	Tipografías y «formatos de salida»	23
3.3	Características principales de LyX	23
3.4	Cómo es trabajar con LyX	24

3.5	LyX para los irreducibles de \LaTeX	26
3.6	Cuándo sí, cuándo no	27
II	Configurando, que es un gerundio	29
4	La interfaz gráfica de LyX	31
4.1	Descripción general de la interfaz	31
4.2	Los atajos de teclado	34
4.3	Muchos documentos, en muchas formas	35
5	Construir documentos	37
5.1	Trabajar en LyX	37
5.1.1	Formato general del documento	37
5.1.2	Texto	41
5.1.3	Listas	42
5.1.4	Matemáticas	43
5.1.5	Cuadros, ¡que no se llaman tablas!	44
5.1.6	Marcos	46
5.1.7	Figuras, cuadros y otros <i>flotantes</i>	47
5.1.8	Referencias cruzadas	48
5.1.9	Índices	50
5.1.10	Generar PDF <i>dinámicos</i> con el paquete <i>Hyperref</i>	52
5.1.11	Navegar y organizar el documento	55
5.2	Apéndices	56
5.3	Bibliografía	56
5.4	Idioma	57
5.4.1	«Comillas “anidadas” dinámicas»	58
5.4.2	Diccionarios	58
5.4.3	Babel vs. Polyglossia	60
5.4.4	Pasar opciones a Babel en \LaTeX / <i>LyX</i>	60
5.5	Importar y exportar documentos	62
5.6	Copiar información desde otros programas	62
5.7	Control de versiones	63
5.8	Trabajar con <i>ramas</i>	64
5.9	Documentos maestros	67

III	Más allá de la configuración básica	69
6	Algunos comandos	71
6.1	El preámbulo \LaTeX	71
6.2	La caja \TeX	72
6.3	Una cuestión de tamaño	73
6.4	Una cuestión de forma (tipográfica)	74
6.5	¡Números!	76
6.6	¡Reconfigurar!	76
7	\LaTeX, OpenType y la tipografía	77
7.1	Introducción	77
7.2	OpenType para la fuente predefinida	78
7.3	Definir una familia de fuentes tipográficas	81
7.4	OpenType para todo el documento	82
7.5	Distintas fuentes para distintas formas	83
7.6	Distintas fuentes para distintos tamaños	84
7.7	Distintas opciones OpenType para cada forma	86
7.8	OpenType para las matemáticas 1: <code>unicode-math</code>	87
7.9	OpenType para las matemáticas, 2: <code>mathspec</code>	89
7.10	Interlineado	90
7.11	Separación silábica	93
7.12	Microtipografía	94
7.13	Sub- y superíndices	96
8	Notas al pie y notas al final	99
8.1	Formato de las notas al pie y sus «marcas»	99
8.2	Notas al pie en columnas con texto principal sin columnas	101
8.3	Notas al pie sin quiebres de párrafo	101
8.4	Reiniciar la numeración de las notas al pie en cada página	101
8.5	«Numerar» las notas con símbolos	102
8.6	Volver desde una nota al pie	103
8.7	Notas del editor	104
8.8	Notas al final	106
8.9	Una <i>yapa</i> : notas al margen	107

9 Módulos, formato local y macros matemáticas	109
9.1 Módulos	109
9.2 Crear nuevas instrucciones	110
9.3 Crear estilos de carácter en $\text{L}\text{\AA}\text{X}$	112
9.4 ¿Entorno de párrafo?	115
9.5 Macros en modo matemático para $\text{L}\text{\AA}\text{X}$	116
10 Listas numeradas	121
10.1 Modificar todas las listas del documento	121
10.2 Modificar algunas listas numeradas	122
11 Sobre los índices	125
11.1 Títulos cortos	125
11.2 Notas al pie en títulos	126
11.3 Cambiar el nombre del índice	126
11.4 Índices parciales para cada capítulo	127
11.5 Corregir espaciado en listas de figuras o cuadros	127
11.6 Evitar las líneas de puntos en los índices	128
11.7 Agregar una línea con texto al índice	129
11.8 Quitar el número de página de las entradas para las Partes	129
11.9 Unas (pocas) palabras sobre los índices alfabéticos	130
12 Títulos	131
12.1 Renombrar los entornos Parte y Capítulo	131
12.2 Numerar capítulos con números romanos	132
12.3 Iniciar los capítulos en cualquier página	132
12.4 Evitar los quiebres de página al comenzar un capítulo	132
12.5 Cambiar el formato de los títulos en $\text{L}\text{\AA}\text{X}$	133
12.5.1 Los paquetes a utilizar	133
12.5.1.1 <code>xcolor</code>	134
12.5.1.2 <code>titlesec</code>	134
12.5.2 Un ejemplo completo	134
12.6 Reiniciar la numeración luego de una Parte	136

13	Cabeceras, pie y estilos de página	139
13.1	Preparativos	139
13.2	El paquete fancyhdr	140
13.3	Un ejemplo	142
13.4	Crear y utilizar estilos de página	143
13.5	Primera página del capítulo	146
13.6	Páginas en blanco	147
13.7	Cambiar el estilo de página del entorno Parte	148
14	Figuras y otros flotantes	151
14.1	Tamaño de las imágenes insertadas	151
14.1.1	Ajustar el tamaño de la imagen en el documento compilado	151
14.1.2	Tamaño de la imagen en la ventana de edición	153
14.1.3	Por no repetir: grupos de imágenes	153
14.2	Figuras envueltas	155
14.3	Configurar las leyendas de figuras y cuadros	157
14.4	Elegir estilo de página para un flotante particular	159
14.5	Renombrar flotantes	159
14.6	Cambiar el tipo de numeración de figuras y cuadros	160
15	Letras capitales y afines	161
15.1	Algo tradicional	161
15.2	¡Atención al número de líneas!	163
15.3	Una variante: letra capital «alzada»	164
15.4	Un desafío	164
15.5	Una alternativa: primera línea de un párrafo en versalitas	165
16	Extras	167
16.1	Describir teclas y menús: el paquete menukeys	167
16.1.1	Qué es menukeys	168
16.1.2	Usar menukeys en L ^A T _E X	169
16.2	Otra forma de representar teclas	171
16.3	Leyendas laterales para figuras/cuadros	172
16.4	Agregar un nuevo flotante	174
16.5	Cajas coloreadas	175

16.6	Cosas de las que no he hablado	176
16.6.1	Soporte Bib \LaTeX	176
16.6.2	Listado de código con listings	176
16.6.3	Formularios PDF	177
16.6.4	Sweave y knitr	177
16.6.5	Sistemas de álgebra computacional	177
16.6.6	Y mucho, mucho más	177
IV	Apéndices	179
A	Sobre fuentes tipográficas	181
A.1	Las fuentes utilizadas en este libro	181
A.2	Más sobre fuentes tipográficas	182
B	Formato local	183
C	Preámbulo \LaTeX	187
D	Otros formatos... y un saludo	191
D.1	Otros formatos utilizados en este libro	191
D.2	Y sí, también formato directo	192
D.3	Saludo final	193

Parte I

Los cimientos

Donde se definen los conceptos generales sobre los que se basará todo lo que sigue

On the shoulders of giants

Capítulo 1

Introducción

Hace ya muchos años existió una revista digital en español que hablaba de software libre llamada *Begins*. Para esta revista escribí varios artículos, algunos sobre Writer, otros sobre LyX. De los primeros nació el libro *Domando al escritor*¹, de los segundos el curso básico de LyX en mi blog², el cual dio lugar a más y más artículos que son ahora presentados, en forma actualizada, ordenada y por sobre todas las cosas ampliada, en este libro.

1.1 Para quién es este libro

Decir que LyX es una «interfaz gráfica para L^AT_EX» no solo es impreciso, también resulta tramposo.

LyX es un procesador de textos construido *sobre* L^AT_EX (y derivados, que en este libro hablaremos mucho sobre X_YL_AT_EX) que ofrece a sus usuarios una variedad de herramientas gráficas y de texto que satisfarán incluso a los más exigentes.

En principio (que la vida siempre es complicada) resultaría posible el utilizar LyX sin conocer qué hay debajo, pero claramente esto sería una enorme limitación ya que, como te mostraré a lo largo del libro, este programa nos ofrece mucho más que una serie de menús y botones.

LyX puede ayudar tanto a los usuarios noveles que nunca han utiliza-

1. <https://elpinguinotolkiano.wordpress.com/domando-al-escriptor-2/>

2. El pingüino tolkiano: <https://elpinguinotolkiano.wordpress.com/>

do \LaTeX como a los expertos que no tienen problemas a la hora de *programar*³ sus documentos: los usuarios noveles encontrarán que su ingreso en el mundo \LaTeX será más tranquilo que con otras alternativas mientras que los avanzados verán que resulta increíblemente sencillo el implementar sus viejos trucos en \LaTeX .

Y aquí llegamos a un punto fundamental en este discurso, ya que tarde o temprano algo de \LaTeX tendrás que aprender, te guste o no. De hecho, muchas herramientas gráficas de \LaTeX como la creación de estilos de carácter o de párrafo exigen un conocimiento más o menos importante de \LaTeX . Entonces, ¿vale la pena el esfuerzo?

Pues como todo en la vida, *depende*.

\LaTeX es un sistema increíblemente poderoso, pero como dice el viejo cliché «un gran poder viene con una gran responsabilidad». Y en este caso la responsabilidad está en aprender a utilizar el sistema.

Si alguien te proporciona la *documentclass* de base (la *plantilla* del documento) y solo te toca completar el contenido, \LaTeX resulta *muy* simple de utilizar: los comandos de base son sencillos y fáciles de comprender, la estructura del documento resulta lógica y clara, cuando te acostumbras a escribir expresiones matemáticas estas fluyen naturalmente, la numeración y las referencias cruzadas son simples de realizar y mantener, etcétera. Y todo esto se vuelve idílico en un programa como \LaTeX donde no tienes que preocuparte de paréntesis olvidados y el código no molesta en la lectura de lo que estás escribiendo.

Si nadie te proporciona una *documentclass*, pero estas conforme con uno de los predefinidos y no necesitas cambiar nada, \LaTeX sigue resultando bastante simple. Nuevamente, disponer de una interfaz gráfica bien pensada te ayudará en muchos de los detalles *pesados*, como definir la página y sus márgenes, las fuentes tipográficas, idiomas, etcétera. Un usuario novel podría tener algunos inconvenientes a la hora de crear páginas especiales como la dedicatoria o el copyright, pero seguramente podrá resolverlo en pocos minutos preguntando a su buscador de internet preferido.

Si tienes que ir más allá de la configuración básica... Todo dependerá

3. \TeX , la base sobre la cual se construye \LaTeX , es un lenguaje de programación *Turing completo*: básicamente cualquier procedimiento que pueda ser escrito como un algoritmo puede ser resuelto con \TeX . Es decir, nadie dice que sea fácil, pero es *posible*.

de qué tan lejos quieres llegar. Ciertamente L^AT_EX te ayudará en el camino, pero también es claro que no puede hacerlo todo por ti: L^AT_EX es *demasiado* grande.

Para darte una idea, el preámbulo L^AT_EX de este documento tiene unas de 120 líneas de código, el «formato local» (definición de estilos) tiene más de 90 líneas y en muchos lugares he recurrido a código en el documento mismo (las famosas «cajas T_EX» de las que te hablaré repetidamente). También es necesario prestar mucha atención a los paquetes utilizados ya que algunos de ellos podrían entrar en conflicto y no funcionar apropiadamente. Por ejemplo, `titlesec`, un paquete para dar formato a los títulos, no se lleva bien con `tocloft`, un paquete para dar formato a los índices⁴.

En definitiva, L^AT_EX es tan potente que se vuelve un universo en sí mismo, un universo por momentos contradictorio en el cual no siempre es sencillo el conocer sus reglas.

L^AT_EX te permite realizar todo lo que quieras en términos de formato. No solo la facilidad de escribir expresiones matemáticas lo vuelve imprescindible en la escritura de documentos técnicos, su versatilidad te permite buscar tanto de la belleza del contenido como la de su presentación. Opciones tipográficas avanzadas, simplicidad a la hora de manejar figuras y cuadros⁵ (sí, eso que tantos dolores de cabeza causa en Writer o en Word resulta increíblemente sencillo en L^AT_EX), todo está a nuestro alcance.

Por lo tanto la pregunta de si vale la pena el aprender L^AT_EX no es correcta, deberías preguntarte si te interesa alcanzar los resultados que L^AT_EX puede ofrecerte y si estás dispuesto a realizar el esfuerzo que ese aprendizaje exige.

Y eso es algo que solo tu, lector, puedes responder.

Te invito por lo tanto a acompañarme en esta aventura tipográfica, así quizás se aclaren tus ideas sobre estos temas. No temas experimentar: siendo L^AT_EX software libre solo tienes que instalarlo y ver si cumple con tus expectativas.

-
4. Esto seguramente está relacionado con el hecho de que el paquete `titlesec` viene junto a otro paquete llamado `titletoc` que sirve para configurar los índices. El problema está en que `titletoc` no es ni remotamente tan potente como `tocloft`.
 5. Algunos las llaman «tablas», pero no por eso dejan de ser «cuadros».

1.2 La estructura de este libro

El libro está separado en tres partes más algunos apéndices. Te llevaré desde lo más simple a lo más complejo, comenzando con una introducción superficial a qué es \LaTeX , siguiendo con una descripción rápida de la interfaz de LyX y de cómo se trabaja en este programa y terminando por temas avanzados de construcción del documento.

Como ya comenté, algunas de las herramientas gráficas de LyX requieren conocer algo de código \LaTeX antes de poder utilizarlas, por lo que su explicación quedará relegada a la parte final. Un ejemplo de esto es la creación de estilos de carácter: solo luego de saber cómo aplicar las distintas opciones tipográficas a mano, de ser capaz de crear nuevas instrucciones y comandos \LaTeX , de perderle el miedo al preámbulo \LaTeX , solo entonces podrás ir al *formato local* para crear tus estilos, los cuales serán posteriormente accesibles con un simple clic derecho.

Esto hará que durante el desarrollo del texto algunos temas parezcan ser más difíciles de lo que realmente son, pero no hay alternativa: antes de poder correr tienes que aprender a caminar sin tropezarte.

1.3 Un poco de historia

Siguiendo con la broma de la cubierta de este libro podría escribir

*Las puertas de LyX, señor de los textos. Di, documentclass, y entra.
Matthias Ettrich me ha creado, la comunidad me ha hecho crecer.*

Pero para llegar aquí muchas cosas han sido necesarias. Daré ahora un exageradamente rápido y absurdamente simplificado repaso de las principales.

1977 Cuando Donal Knuth recibió la prueba de impresión de la segunda edición de su libro, *The Art of Computer Programming*, la juzgó horrible. En esos tiempos él ya había comenzado a interesarse en los sistemas tipográficos digitales por lo que la pobre presentación de su libro lo empujó a crear su propio sistema tipográfico, $\text{T}_{\text{E}}\text{X}$. Pasarían muchos años antes de que el sistema estuviera listo para uso general (más de los que él previó en un principio), pero todo comenzó en aquel momento.

- 1985 \LaTeX fue creado por Leslie Lamport como una forma de simplificar el uso de \TeX a través de una serie de *macros* que realizaran tareas específicas como crear notas al pie, títulos, etcétera, además de dar herramientas para compilar los documentos.
- 1995 Matthias Ettrich comenzó a desarrollar una interfaz gráfica para \LaTeX llamada Lyrix. Poco después el proyecto cambió nombre para evitar conflictos con un ahora desaparecido procesador de textos comercial. El nombre finalmente elegido, LyX , proviene de la extensión de los archivos utilizados por Lyrix. Al año siguiente, Matthias Ettrich crearía el proyecto por el cual es más conocido: KDE.
- 2004 Se publica $\text{Xe}\TeX$, un sistema basado en \LaTeX con soporte Unicode y que puede utilizar tecnologías tipográficas modernas como OpenType, AAT y Graphite.
- 2011 A partir de la versión 2.0 LyX ofrece soporte directo para $\text{Xe}\TeX$, $\text{Lua}\TeX$ y XHTML.

Y el resto es historia.

Más información:

<https://en.wikipedia.org/wiki/TeX>

<https://en.wikipedia.org/wiki/LaTeX>

<https://en.wikipedia.org/wiki/LyX>

<https://en.wikipedia.org/wiki/XeTeX>

https://en.wikipedia.org/wiki/Donald_Knuth

https://en.wikipedia.org/wiki/Leslie_Lamport

https://en.wikipedia.org/wiki/Matthias_Ettrich

<http://www.lyx.org/misc/archaeology/>

http://tolkiengateway.net/wiki/Doors_of_Durin

1.4 Bases mínimas

Daré por supuesto, lector, que tienes *al menos* las mismas herramientas que he utilizado para escribir este libro. Estas son:

- L^AT_EX 2.4 o posterior
- Una distribución L^AT_EX equivalente a T_EXLive 2021 o posterior
- (Opcional) Pandoc 2.17 o posterior

También daré por supuesto que sabes, o al menos tienes a quién preguntarle, cómo se instalan los paquetes de esa distribución L^AT_EX. Esto implica claramente que doy por supuesto que sabes utilizar tu computadora.

Y lo más importante, daré por cierta tu voluntad de experimentar y de profundizar por tu cuenta los temas aquí tratados. En este libro no encontrarás detalladas guías sobre cómo hacer esto o aquello, o ejercicios para acostumbrarte a utilizar el programa: para eso ya existe la excelente documentación incluida en L^AT_EX y que encuentras bajo el menú Ayuda. Por esto y si bien muchos temas serán tratados en forma más o menos exhaustiva muchos otros serán solo *presentados*, dejando como tarea para tu curiosidad el familiarizarte con la documentación y el tratar de ver más allá de las opciones predefinidas. ¡La mejor forma de *aprender es hacer!*

1.5 Convenciones utilizadas en este libro

Las rutas a los distintos menús serán indicadas mediante un formato especial⁶. Por ejemplo, la ruta para llegar a la edición del «preámbulo L^AT_EX» (menú «Documento», luego a «Configuración» y por último «Preámbulo L^AT_EX») será indicada así:

Documento » Configuración » Preámbulo LaTeX

Para indicar elementos *dentro* de la ventana a la que nos ha llevado el menú, usaré VERSALITAS.

Los atajos de teclado también tendrán un formato especial⁷. Por ejemplo la combinación de teclas «control» y «E» se presentará como

Ctrl E

mientras que la secuencia «Alt» con «P», luego «2» se mostrará en forma similar a lo siguiente:

Alt P, luego 2.

6. Cortesía del paquete menukeys. Volveré sobre esto en el apartado 16.1.2.

7. Utilizando un estilo de carácter con la fuente Libertinus Keyboard.

Soy consciente de que el sistema operativo de las «mac» requiere adaptar estos atajos, pero no tengo acceso a ese sistema por lo que no puedo proporcionar las equivalencias.

Para indicar código o nombres de paquetes se utilizará una fuente de ancho fijo:

```
\usepackage{emptypage}
titlesec
```

Para indicar direcciones, ya sean estas en internet o en el sistema de archivos (mayormente para indicar la localización de la documentación de un paquete particular) se utilizará una fuente palo seco (*sans serif*):

```
https://www.lyx.org/
/usr/share/texmf/doc/latex/fancyhdr/
```

Sobre esto último, escribo desde un sistema Linux⁸, por lo que las direcciones que daré serán las de mi sistema. Si utilizas otro sistema operativo o algo diferente de T_EXLive, deberás adaptar las direcciones, pero no será difícil el hacerlo.

Si tienes la versión PDF de este documento, los enlaces a internet estarán en color azul mientras que los enlaces internos al documento (índice, referencias cruzadas, etcétera) se mostrarán en color rojo. Todos los enlaces son «activos» en el PDF: un clic en ellos te llevará al lugar enlazado, ya sea abriendo el navegador de internet en la página indicada o desplazándose dentro del documento.

1.6 Enlaces a tener a mano

La página del proyecto Ly_X se encuentra (en inglés) en

<https://www.lyx.org/>

Su wiki (también en inglés) en

<https://wiki.lyx.org/>

Para obtener soporte sobre Ly_X tienes la lista de correo en español. Para suscribirse es suficiente enviar un correo electrónico a:

8. O GNU/Linux, si prefieres. O GNU/BSD/MIT/Apache/MPD/etcétera/Linux, para ser más precisos. Yo uso openSUSE Leap.

lyx-es-subscribe@lists.lyx.org

Luego de confirmar la suscripción será suficiente escribir a:

lyx-es@lists.lyx.org

Si te llevas bien con el inglés, puedes probar la lista:

lyx-users@lists.lyx.org

a la que te puedes suscribir de la misma forma que la otra, enviando un mensaje a:

lyx-users-subscribe@lists.lyx.org

Es importante notar que las listas de correo están configuradas para «responder al remitente» por lo que al mantener una conversación allí no olvides agregar la lista como destinatario. Ni te cuento las veces que lo he olvidado.

Recuerda que son listas públicas, por lo que es conveniente utilizar una dirección de correo dedicada a comunicarse con listas y no la dirección privada.

El sistema para reportar errores del programa se encuentra en

<https://www.lyx.org/trac/wiki/BugTrackerHome>

A medida que avancemos en este libro será necesaria cada vez más información sobre \LaTeX y el lugar apropiado para obtener la documentación de cualquier paquete —además de nuestra instalación local, se entiende— es el repositorio CTAN (*Comprehensive $T_{\text{E}}X$ Archive Network*):

<http://ctan.org/>

Si quieres profundizar en el sistema que hace que $\text{L}_{\text{Y}}\text{X}$ funcione existe un «curso no convencional de \LaTeX » en español en el siguiente sitio:

<https://ondiz.github.io/cursoLatex/>

o bien, en «formato blog», aquí:

<https://ondahostil.wordpress.com/curso-no-convencional-de-latex/>

Y por supuesto, el lugar donde todas las respuestas sobre \LaTeX pueden encontrarse (en inglés) es *$T_{\text{E}}X$ - \LaTeX Stack Exchange*:

<https://tex.stackexchange.com/>

Si nadie allí sabe resolverlo, es imposible. Y como casi nada es imposible en \LaTeX , allí estará la respuesta.

Capítulo 2

Definiendo \LaTeX

Como ya te comenté en la introducción, *en principio* no es necesario conocer \LaTeX (o $X\TeX$) para utilizar LyX , este último ofrece una agradable interfaz gráfica que simplifica el uso del primero. Pero también es cierto que conocer algo, que saber qué es y cómo funciona \LaTeX te permitirá controlar la apariencia de tus documentos en un nivel de detalle inimaginable por procesadores de texto tradicionales.

Y es por esto que el libro comienza con una introducción superficial de \LaTeX , que tener en claro qué es lo que sucede «debajo del cofre» te permitirá llegar más lejos.

En la tercera parte de este libro, a partir del capítulo 6, volveré sobre estos temas, que para sumergirnos en la parte avanzada no queda otra alternativa que escribir código. Pero por ahora bastará con recorrer en forma relajada y superficial algunos conceptos básicos.

2.1 Qué es \LaTeX

La idea fundamental en \LaTeX y derivados como $X\TeX$ o LuaTeX (que no siempre se logra, todo hay que decirlo) es separar el *contenido* del *formato*.

Para indicar el título de un capítulo dices solamente «capítulo» (en lenguaje \LaTeX , se entiende) y no te preocupas de la tipografía, del tamaño o de la numeración: de esos detalles se encarga \LaTeX en forma automática. ¡Solo debes preocuparte por el título en sí mismo!

Y es que el formato se encuentra definido separadamente en un archivo especial llamado *documentclass* (o clase de documento) que es llamado en nuestro documento, junto a otras opciones de configuración, en lo que se conoce como *preámbulo L^AT_EX*.

En este sentido L^AT_EX es como un lenguaje de programación interpretado en donde se dan instrucciones que el sistema luego usará para construir el documento. Estas instrucciones pueden ser escritas en un archivo de texto normal, por lo que en principio no se necesitaría de un programa especial para escribir un documento L^AT_EX, si bien es altamente recomendable el usar uno. Y para esto tenemos al protagonista de este libro: LyX.

Esta doble estructura, *archivo de contenido + archivo con definiciones de formato* es a la vez la mayor fuerza y el principal problema de L^AT_EX y derivados.

La mayor fuerza porque es posible cambiar de una *documentclass* a otra sin mayores inconvenientes, porque tener los estilos separados del contenido hace que el formato del documento sea consistente, porque si alguien te provee la *documentclass* tu único trabajo será escribir ya que la estructura del archivo con el contenido es estándar y fácil de comprender...

El mayor problema porque alguien te tiene que dar esa famosa *documentclass* ya que es sumamente difícil el definir una por tu cuenta, porque utilizar otras fuentes tipográficas que las predefinidas puede tener sus be-moles, porque tratar de desviarse aunque más no sea ligeramente de lo que ofrece la *documentclass* elegida puede resultar en mucho trabajo extra.

Afortunadamente, cualquier instalación estándar de L^AT_EX como T_EX Live (para Linux) o MiK_TE_X (para el «sistema de la ventana»¹) traen empaquetadas una infinidad de *documentclass* que cubren prácticamente todos los gustos. Además, con un poco de trabajo siempre podrás personalizar todo.

El problema está en que L^AT_EX está más cerca de un lenguaje de programación que de un sistema de redacción de textos tradicional: el usuario común, es decir, el no académico que no necesita escribir complejas fórmulas matemáticas, suele intimidarse con su estructura, ya que si bien las instrucciones L^AT_EX son simples e intuitivas siguen siendo *instrucciones*.

Pero no te preocupes que, como te mostraré en las próximas páginas,

1. No quiero infringir cuestiones de copyright.

este problema no es completamente real.

2.2 \LaTeX como lenguaje de etiquetas (o casi...)

El lenguaje de programación por etiquetas más famoso es quizás el html, por lo que si conoces algo al respecto podrás hacerte rápidamente una idea de qué estamos hablando: cada elemento de un documento \LaTeX viene envuelto en etiquetas que identifican las propiedades que queremos asignarle.

Por ejemplo, si quieres escribir una famosa expresión matemática como puede ser² $E^2 = (mc^2)^2 + (pc)^2$, para hacerlo incluido en un párrafo, en \LaTeX *tradicional* (con \LaTeX es más fácil) escribes

```
 $E^2 = \left(mc^2\right)^2 + \left(pc\right)^2$ 
```

mientras que si quieres que la expresión resalte en su propio párrafo, con un número que cambiará automáticamente al incluir otras ecuaciones

$$E^2 = (mc^2)^2 + (pc)^2 \tag{2.1}$$

entonces debes escribir

```
\begin{equation}
  E^2 = \left(mc^2\right)^2 +
  \left(pc\right)^2
\end{equation}
```

Por otra parte, si el número no te interesa puedes utilizar

```
\[
  E^2 = \left(mc^2\right)^2 +
  \left(pc\right)^2
\]
```

2. No, no está equivocada: *la otra* está equivocada (o mejor dicho, incompleta)

También en L^AT_EX se tienen instrucciones más semejantes a «funciones» (de allí lo de *casí*). Por ejemplo, para indicar el título de una sección (volveré sobre qué se entiende por sección más adelante) se escribe

```
\section{Esto es un título de sección}
```

Muchas instrucciones pueden anidarse, por lo que si quieres insertar una expresión matemática en el título de una sección simplemente escribes algo como esto:

```
\section{La energía relativista:
  $E^2=\left(mc^2\right)^2+
  \left(pc\right)^2$}
```

También puedes «darle un nombre» a la ecuación numerada de antes para después citarla con una referencia cruzada. Para esto se usa la instrucción

```
\label{nombre}
```

Con todo esto un típico documento L^AT_EX puro y duro sería parecido a lo siguiente (más algunas otras cosas que definen el tipo de documento y de las cuales hablaré más adelante):

```
\begin{document}
\section{La energía relativista:
  $E^2=\left(mc^2\right)^2+
  \left(pc\right)^2$}
La famosa relación entre energía, masa y
el valor de la velocidad de la luz
elevado al cuadrado, $E=mc^2$ es
en realidad un caso particular de
una expresión más importante, la
cual presentamos, numerada
y en toda su gloria, aquí:
\begin{equation}
E^2=\left(mc^2\right)^2+
```

```

\left(pc\right)^{2}
\label{eq:Einstein}
\end{equation}
La ecuación \ref{eq:Einstein} trajo no
pocos dolores de cabeza a los padres de
la mecánica cuántica.
\end{document}

```

Insisto en que con \LaTeX no será necesario el escribir todo esto, solo te estoy mostrando cómo funcionan las cosas debajo del cofre.

Como puedes comprobar, en el citado código se ha hecho una referencia cruzada (como esta: ver ecuación 2.1) a la ecuación numerada usando la instrucción

```
\ref{nombre}
```

Es importante notar que tanto `\label` como `\ref` sirven para mucho más que llamar ecuaciones. Estas instrucciones son sumamente flexibles y si colocas `\label` junto al título de una sección (o junto a una figura o un cuadro), `\ref` comprenderá dónde se encuentra la referencia a la cual está llamando y la mostrará en el modo correcto.

De este simple (y claramente incompleto) ejemplo resulta claro que durante la escritura de un documento \LaTeX te ocupas solamente del contenido, no de su forma; incluso la numeración de las secciones y de las ecuaciones (y de las figuras y de los cuadros) se realiza en forma automática.

Otra cosa que en \LaTeX se realiza en forma automática y que es una verdadera bendición en documentos muy complejos es la ubicación de figuras y cuadros: no solo cambiarán de página según sea necesario, sino que además el texto en torno a ellas se moverá automáticamente en forma tal de no dejar huecos en la página.

Sí, léiste bien, sin mayor esfuerzo por parte del escritor³ desaparecen en \LaTeX las eternas luchas con los espacios en blanco que quedan cuando una figura pasa a la página siguiente: el sistema, usando criterios que pueden personalizarse fácilmente, se encarga de que todo quede perfecto.

3. Más adelante veremos los límites de todo esto, que en la vida nada es perfecto, pero no hay que preocuparse que son fáciles de superar.

2.3 Clases de documentos

Lo comentado en la entrada anterior, me dirás, de pensar más en el contenido de un documento que en su forma está muy bien, pero tarde o temprano tendré que preocuparme también sobre cómo se verá lo que estoy escribiendo. ¿Cómo decido qué tipo de letra usar, sus tamaños y todas esas cosas que siempre son necesarias?

Como ya comenté, \LaTeX toma la definición de cada estilo en particular de una *documentclass*, o clase de documento, la cual no es más que un archivo especial en el que se definen todas las cuestiones de formato.

Los formatos definidos son de distintos tipos. Tienes *entornos* que definen cómo se comportan párrafos determinados, o fragmentos de texto; tienes los formatos de las páginas; de las notas al pie; de todo lo que construye el documento.

Así, en las clases de documento de artículo puedes separar el contenido en secciones, subsecciones, subsubsecciones, párrafos, subpárrafos y, por supuesto, el cuerpo del texto. En las clases de libros se agregan, por encima de estos niveles, los capítulos, y por encima de estos últimos, las partes. Así, qué tamaño y característica de letra debe utilizarse cuando la instrucción `\section` se presenta está determinado por la *documentclass*.

Para elegir un determinado tipo de documento, como por ejemplo *artículo* (artículo), y el idioma en el que queremos escribir, por ejemplo español, en un documento \LaTeX puro debemos escribir al principio (antes de `\begin{document}`):

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

Para crear una *documentclass* de la nada se necesitan conocimientos avanzados de \LaTeX , pero afortunadamente no es necesario el hacerlo ya que toda distribución \LaTeX moderna cuenta con un enorme número de *documentclass* y en el repositorio CTAN puedes encontrar muchas más. Esto, sumado a la posibilidad de realizar ajustes finos en los valores por defecto de la *documentclass*, te permitirá darle a tus documentos el formato que quieras casi sin problemas.

Por ejemplo, si en lugar de utilizar un tamaño de papel A4 que es el formato por defecto de la *documentclass* «`article`» quieres usar A5, la instrucción de arriba cambia a:

```
\documentclass[a5paper,spanish]{article}
```

En esta parte del documento anterior al `\begin{document}`, parte que viene llamada *preámbulo* \LaTeX , pueden definirse muchas cosas. A partir del capítulo 6 haremos uso intensivo del mismo, por lo que ahora te dejaré descansar de estos códigos. Pero solo por ahora.

2.4 La construcción del documento

Dado que la idea de este libro es introducir una herramienta gráfica para manejar \LaTeX , no entraré en detalles técnicos sobre cómo se construye el documento ya que \LaTeX lo hará por ti: solo delinearé las ideas básicas.

Tu documento es hasta ahora solo una sucesión de instrucciones \LaTeX que deben ser *compiladas* (generalmente, más de una vez) para crear el resultado final. Esta compilación se realiza llamando el programa «`latex`» u otros asociados (`pdflatex`, `xelatex`, etcétera) con una serie de parámetros entre los que se encuentra, claramente, el nombre del archivo por ti escrito. \LaTeX genera entonces un archivo `.pdf` que puede ser visto e impreso con Okular u otros programas semejantes y en donde todas las instrucciones de la *documentclass* resultan implementadas. También es posible, utilizando el paquete `hyperref`, obtener un documento dinámico en el que los enlaces, ya sean estos internos al documento o externos a otros documentos o sitios web, serán activos (hablaré de esto más adelante).

2.5 Y *OpenType* llegó a \LaTeX

Algo de lo que no he hablado hasta ahora es de cómo seleccionar fuentes tipográficas. Esta omisión se debe a dos razones, siendo la primera que hacer esta selección desde \LaTeX es infinitamente más simple que hacerlo manualmente.

La segunda razón es un tanto más pesada: \LaTeX utiliza su propio sistema de fuentes tipográficas, el cual no solo es completamente incompatible con todo otro sistema sino que además es realmente complicado.

Las fuentes en \LaTeX no se llaman por familia o tipo de fuente, sino por

nombre de archivo⁴. Y para complicar aún más las cosas, \LaTeX en su forma tradicional solo recientemente ha adquirido soporte Unicode. Esto último significa que para dar soporte a lenguajes diferentes del inglés se deben dar algunos rodeos.

Existen variantes de \LaTeX que permiten solucionar estos dos problemas, sobre todo el del soporte Unicode. En este libro me centraré en el que es quizás el más maduro de todos los proyectos, $X\TeX$, pero lo que comento aquí será mayormente aplicable a \LuaTeX también.

$X\TeX$, inicialmente desarrollado por la gente de Sil internacional justamente para dar soporte a lenguajes no occidentales, es una variante de \LaTeX que no solo soporta Unicode sino que además nos permite utilizar las fuentes del sistema. De hecho $X\TeX$ es uno de los pocos sistemas libres en dar soporte completo a fuentes *OpenType*⁵, permitiendo utilizar todas sus características avanzadas como ser sustitución automática de caracteres (por ejemplo, ligaduras tipográficas), uso de verdaderas versalitas, verdaderos sub- y superíndices... todo lo que las fuentes provean.

$X\TeX$ llama las fuentes en forma distinta de como lo hace \LaTeX , pero dado que no he entrado en detalles de cómo lo hace este último tampoco hablaré sobre cómo lo hace el primero.

Al menos no por el momento: volveré sobre estos temas brevemente en el capítulo 5 y en gran profundidad (carga tus tanques de oxígeno) en el capítulo 7.

Pero bueno, ya está bien de *bases teóricas* y de esta historia de ver qué hay *debajo del cofre*: este libro es sobre LyX por lo que en el próximo capítulo comenzaré a hablar de sus posibilidades.

Sírvete un poco más de café, estimado lector, que ya estamos en camino.

4. Antes de que los puristas \LaTeX me protesten, sí, ya sé que existen *paquetes* que permiten seleccionar grupos de fuentes rápidamente. Pero esto en realidad refuerza lo que estoy diciendo: manejar fuentes *metafont* a mano no es tarea sencilla.

5. Tanto LibreOffice como Scribus ofrecen un soporte bastante completo para esta tecnología.

Capítulo 3

Qué es lo que hace que LyX sea tan especial

Existe una larga guerra entre quienes creen que la mejor forma de crear documentos es con editores de texto como se hace con \LaTeX puro y quienes creen que nada supera una buena interfaz gráfica con sus menús desplegables y sus botones por todas partes como... bueno, seguramente conoces *al menos* un producto de ese tipo.

Los defensores de la edición basada en texto dicen, y suelen tener razón, que cuando se conocen los códigos correctos es mucho más simple escribir que andar haciendo clic por aquí y allá y que en los editores gráficos se pasa uno más tiempo buscando la función que se necesita que usándola.

Los defensores de los programas de edición gráficos dicen, y suelen tener razón, que recordar una infinidad de comandos por más simples que estos sean implica un esfuerzo la mayor parte de las veces injustificado y que la *curva de aprendizaje* de un programa gráfico es siempre más suave que en un programa basado en texto.

A esta guerra se agrega un tercer grupo formado por aquellos que dicen que los otros dos grupos son unos idiotas y que nada es mejor que un buen atajo de teclado, a lo cual los dos grupos anteriores responden, y suelen tener razón, que hacer todo con atajos de teclado requiere unas habilidades tanto en los dedos como en la memoria propias de un pianista virtuoso.

Pues bien, LyX ha resuelto este antiguo conflicto en la forma más elegante posible.

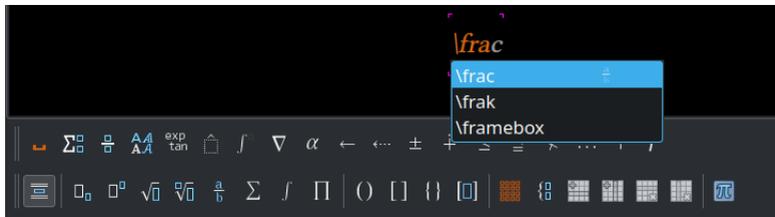


FIGURA 3.1: *el editor de ecuaciones en acción*

Por una parte esas acciones pesadas en \LaTeX que deben realizarse solo una vez por cada documento (la selección del tipo de documento, fuentes tipográficas, formato de página, etcétera) se manejan a través de menús.

Por otra parte aquellas acciones que solo se deben realizar una vez en documentos particulares (cambiar la numeración de las páginas del índice, insertar páginas especiales) se manejan a través de código.

Finalmente, y aquí reside la gran belleza del programa, *todo lo demás puede manejarse ya sea a través de código, menús/botones o atajos de teclado.*

Efectivamente, \LaTeX ofrece lo mejor de *todos* los mundos, dejando contentos a los usuarios más variados.

Considera por ejemplo la figura 3.1.

Aquí se muestra el editor de ecuaciones de \LaTeX , pero el concepto es válido también para otras funciones. Como puedes ver se tiene una barra de herramientas que permite insertar una fracción presionando un botón como en los editores gráficos, pero si en lugar de buscar botones o menús comienzas a escribir la instrucción `\frac`, \LaTeX no solo comprende lo que estás escribiendo sino que además muestra un menú con las posibles instrucciones que pueden usarse con lo que se ha escrito hasta el momento. Siguiendo con el ejemplo: `\frac` para fracciones, `\frac` para utilizar el tipo de letra Fraktur sobre un carácter o `\framebox` para insertar un rectángulo en torno al texto escrito.

Pero concentrémonos por un momento en `\frac`: si luego de terminar de escribir la instrucción presionas la barra de espacio, esta será reemplazada por una verdadera fracción con dos campos de entrada listos para recibir tanto al numerador como al denominador. Ahora bien, si en lugar de escribir `\frac` (o de utilizar el botón correspondiente) presionas **[Alt]** **[M]**, seguido

de **F** obtendrás el mismo resultado.

Y así con todo: tanto `\sqrt` como el botón con la raíz cuadrada como la combinación **Alt** **M**, luego **R** te dará la raíz cuadrada que estabas buscando, con el campo libre listo para aceptar el argumento.

Como ya te comenté, esta versatilidad no está limitada a las ecuaciones. Por ejemplo, en el texto normal puedes utilizar la caja de arriba a la izquierda para seleccionar el tipo de entorno Sección (o Capítulo, o párrafos especiales... los entornos disponibles dependerán de la *documentclass* elegida) o utilizar el atajo **Alt** **P**, seguido de **2**.

Así es, la interfaz de LyX te permite trabajar en las tres modalidades: texto, menús/botones y atajos de teclado. Lo que prefieras.

3.1 El concepto «WYSIWYM»

Lo que muestra LyX durante la edición del documento *no* es exactamente igual a lo que sale una vez que este es compilado, pero te da una idea exacta de lo que el documento *quiere decir*. Por ejemplo, LyX no muestra los quiebres de página ya que para trabajar con el *contenido* son completamente irrelevantes (pero sí, obviamente, los genera al compilar).

Otra cosa que *no* se ve en la ventana de edición son los códigos \LaTeX : todos los `\documentclass`, `\begin{document}`, `\end{document}`... están ocultos y son manejados por LyX sin que el escritor tenga que intervenir directamente. Incluso los títulos de los capítulos y las secciones se ven como títulos verdaderos (si bien en un formato diferente del resultado final) y no con la instrucción `\section`.

Otras cosas a las que los nuevos usuarios de LyX tienen que acostumbrarse, pero que tienen su lógica:

- No hay reglas en la página.
- Si presionas la barra de espacio dos veces seguidas, la segunda será ignorada. Lo mismo sucede con la tecla **←**: los espacios son manejados automáticamente por los estilos, el autor no necesita preocuparse.
- La tecla **↔** solo sirve para moverse entre partes del documento (partes de una expresión matemática, celdas de una tabla...).

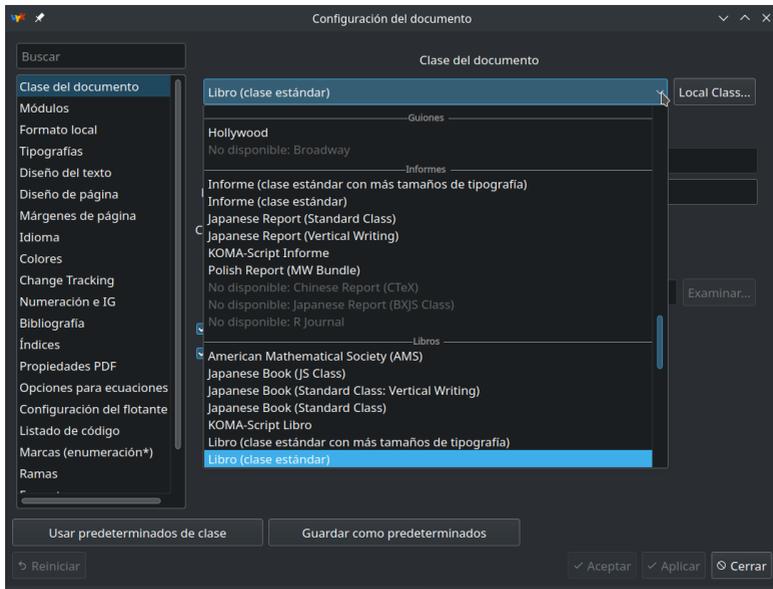


FIGURA 3.2: *eligiendo la documentclass*

- Todo lo que sea espaciado, sangrado de párrafos, formato del texto, contenido de las cabeceras y los pie de página, posición de figuras y cuadros... todas estas cosas se manejan en LyX de forma automática y de acuerdo al contexto, basándose en la *documentclass* elegida en Documento Configuración Clase del documento.

Como se ve en la figura 3.2, existen muchas clases de documentos a tu disposición y siempre puedes conseguir más a través de tu distribución $\text{L}\text{A}\text{T}\text{E}\text{X}$. También tienes, en otros menús, muchas formas de personalizar la elección cambiando por ejemplo la forma en la que se separan los párrafos (sangrado o espacio vertical), el tamaño de la página y sus márgenes, la ubicación de los *flotantes* (ya te diré más adelante qué son), la numeración, etcétera. Las opciones disponibles dependerán de la *documentclass* elegida, pero son siempre tan numerosas que satisfarán incluso al más empedernido de los detallistas.

Además, la forma en la que cada parte del documento se construye no está gobernada solamente desde allí ya que más allá de estas opciones ge-

nerales es siempre posible realizar ajustes finos en partes determinadas del documento. Por ejemplo, en el menú **Editar** » ... puedes cambiar las características de un párrafo en particular, o de una palabra aislada, o de una página en especial, mientras que desde el menú **Insertar** » ... puedes usar saltos de página, espacios especiales entre palabras y párrafos, lo que sea necesario.

¡Ha! **Ctrl** **B** activa y desactiva la negrita mientras que **Ctrl** **E** activa y desactiva la cursiva, al menos eso es (casi) igual que siempre.

3.2 Tipografías y «formatos de salida»

Entre las muchas opciones que el menú **Documento** » **Configuración** ofrece (DISEÑO DE PÁGINA, MÁRGENES DE PÁGINA, IDIOMA...) existen dos entradas que son de interés para este libro: TIPOGRAFÍAS y FORMATOS.

Estas dos opciones te permitirán configurar LyX para que utilice como motor de texto X_YTeX (o LuaTeX, pero no hablaré de esa opción aquí) en la construcción de tus documentos.

Si seleccionas **Tipografías** » **Usar tipografías no TeX (via XeTeX/LuaTeX)** es posible elegir las fuentes que tengas instaladas en el sistema en lugar de las fuentes L^ATeX. Con esta opción activa, LyX utilizará por defecto X_YTeX, pero también puedes ir al menú **Formatos** » **Formato de salida predeterminado** para hacer esta selección explícita eligiendo «PDF (XeTeX)». Como puedes ver existen allí otras opciones, pero no entraré en ellas.

3.3 Características principales de LyX

LyX puede cargar varios documentos simultáneamente, abriéndolos en pestañas separadas al estilo Firefox o bien en una modalidad de *ventana dividida* (volveré sobre esto más adelante).

El programa recuerda en qué parte del documento estabas trabajando al cerrarlo, por lo que al abrirlo nuevamente te llevará allí. Cuenta también con un panel que te muestra la estructura del documento, el cual puede ser activado desde **Ver** » **Panel del esquema** (también tienes un botón en la barra de herramientas principal). Este panel funciona como un índice del documento que te permite no solo llegar rápidamente al punto que te interesa sino también cambiar con poco esfuerzo el orden de los distintos apartados

o modificar el *nivel* del título seleccionado (*degradar* una sección a subsección, por ejemplo). La casilla de verificación MANTENER es interesante para evitar que el árbol de niveles se cierre cuando haces doble clic en un nivel superior. Volveré sobre todo esto en la sección 5.1.11.

El programa ofrece además un sistema de registro de versiones, el cual permite registrar, aceptar o rechazar cambios facilitando así el trabajo cooperativo en un único documento (lo comentaré brevemente en la sección 5.7). Puede usar Enchant (opción por defecto) o Hunspell para la corrección ortográfica en tiempo real y posee un tesoro que puede utilizar los diccionarios de sinónimos de LibreOffice (sección 5.4.2). Permite trabajar con documentos escritos en varios idiomas. Ofrece una forma increíblemente simple y robusta de trabajar con documentos maestros (sección 5.9)...

Algo importante de remarcar es que LyX trabaja con Unicode sin problemas por lo que, incluso en el caso de que decidas no utilizar X_YTeX, no tendrás que preocuparte a la hora de insertar símbolos o acentos ya que LyX se encargará de traducir todo para que L^ATeX lo comprenda sin dificultad.

Debes tener en cuenta que las imágenes no son incorporadas en los archivos LyX, solo son vinculadas y los archivos originales quedan en su lugar. Por lo tanto si tienes que trabajar en un mismo documento desde varios sistemas te tocará un poco de trabajo extra para mantener todo sincronizado. Para simplificar la tarea de mover tu proyecto a otro sistema puedes exportar un paquete comprimido como archivo tar.gz que contenga el documento LyX y todas las imágenes desde  »  » Archivo LyX (tar.gz): simplemente descomprimes el paquete en una carpeta vacía del sistema de destino y ya estás listos para continuar trabajando.

3.4 Cómo es trabajar con LyX

Volveré sobre todo esto en gran detalle a partir del próximo capítulo, pero veamos ahora rápidamente el método de trabajo necesario con LyX.

Comenté antes que la ubicación de las figuras y cuadros se realiza automáticamente. Para esto se debe primero insertar un *flotante* ( o el correspondiente botón de la barra de herramientas) apropiado y dentro de él colocar la figura o cuadro junto con su *leyenda*. Estos flotantes, como así también las notas al pie de página y otros elementos especia-

les se muestran en la ventana de edición de L^AT_EX como una caja roja con una etiqueta que explica de qué se tratan. Estas cajas tienen la particularidad de poder cerrarse dejando visible solo la etiqueta cuando no necesitamos ver su contenido: esto puede hacerse en cada caja particular simplemente haciendo clic sobre su etiqueta o globalmente con  Cerrar todos los recuadros. Muchas de estas cajas ofrecen menús contextuales para configurar su comportamiento.

A este punto es importante notar que L^AT_EX puede trabajar mayormente solo con imágenes postscript encapsuladas (EPS), X_YL_AT_EX o pdfL_AT_EX solo con imágenes PDF... pero a no preocuparse, que L^AT_EX se sirve de distintas librerías para aceptar virtualmente cualquier tipo de imagen, convirtiéndola en el formato apropiado cuando sea necesario. En general, si se trata de diagramas será mejor utilizar imágenes EPS, PDF o SVG ya que al ser formatos vectoriales darán mejores resultados, pero cualquier tipo de imagen de mapa de bits (png, jpg, lo que sea) funcionará a la perfección.

Un concepto importante en L^AT_EX es el de *entorno*. Los entornos sirven para distinguir las diferentes partes del documento, como por ejemplo el título de un capítulo del texto simple que le sigue. Estos entornos se asignan con el menú que se encuentra a la izquierda de la primera barra de herramientas (o con el atajo de teclado correspondiente), justo debajo del menú ARCHIVO.

Los entornos que tengas a disposición dependerán de la *documentclass* elegida. Por ejemplo, al escribir un artículo o una carta no tendrás a disposición Capítulo, como sí lo tendrás al escribir un libro.

Ya te he hablado de los diversos atajos de teclado que ofrece L^AT_EX. Estos atajos son de dos tipos: atajos directos como las combinaciones   y    para introducir una ecuación, o bien secuencias de teclas como la ya comentada   seguido de  para introducir una fracción, o  , luego  seguido de una letra (a, b...) para introducir un carácter griego (α , β ...). Como ya vimos la asignación de los principales entornos puede realizarse desde el teclado a través de la secuencia   seguida de un número del 0 al 6 (o de *b*, o de *e*, ¡prueba a ver qué se obtiene!). Es importante notar que alguno de estos atajos podrían no hacer nada dependiendo de la *documentclass* elegida.

El último paso importante en el trabajo con L^AT_EX es ciertamente la com-

pilación del archivo. Seguramente habrás visto la barra de herramientas que se encuentra justo debajo del menú ENTORNO: de izquierda a derecha tenemos «ver», «actualizar», «ver documento maestro»... los dos primeros botones serán seguramente los más utilizados, al menos para aquellos usuarios que se inician en LyX.

3.5 LyX para los irreducibles de L^AT_EX

Si bien LyX utiliza su propio formato de archivo basado en XML para almacenar el trabajo a no preocuparse, fanáticos L^AT_EX, que la conversión al formato L^AT_EX es siempre perfecta y por una buena razón: LyX *siempre* convierte a formato L^AT_EX cada vez que compila el documento por lo que es algo que *necesariamente* debe funcionar. Por lo tanto, si alguien por alguna razón necesita obtener un archivo en formato L^AT_EX legible por seres humanos, pues que es suficiente ir a  .

Como ya vimos, en LyX es posible introducir directamente código no solo en el modo matemático. Fuera del modo matemático contamos de hecho con el botón T_EX (atajo de teclado:  ) que abre una caja especial donde puedes escribir tu fantástico código. Por ejemplo, si usando alguna de las *documentclass* para libros (todas las variantes de *book* que nos da L^AT_EX) insertas con la caja roja el comando

```
\frontmatter
```

justo delante del índice (  ) y del mismo modo insertas

```
\mainmatter
```

justo después, lograrás que las páginas en las que se encuentra el índice sean numeradas con números romanos, mientras que las páginas siguientes tendrán números arábigos que comenzarán desde 1 en el primer capítulo.

Ya volveré sobre estas (y otras) cosas más adelante.

LyX permite además ver el código L^AT_EX en tiempo real, ya sea del párrafo que se está editando o de todo el documento, simplemente habilitando la opción  .

Por último, aunque no menos importante, LyX ofrece un modo realmente sencillo de editar el preámbulo \LaTeX , es decir (casi) todo el código \LaTeX que puede ser escrito entre las instrucciones `\documentclass` y `\begin{document}` y que afecta al comportamiento general del documento:

Documento » Configuración » Preámbulo \LaTeX .

Las posibilidades son casi ilimitadas y en las próximas páginas te daré varios ejemplos.

3.6 Cuando sí, cuándo no

No existe la herramienta perfecta capaz de hacerlo todo y ciertamente no se utiliza un destornillador para clavar un clavo. LyX es poderoso para una enorme variedad de trabajos, pero podría resultar molesto en otros.

Si quieres escribir un libro, especialmente uno de carácter técnico con fórmulas matemáticas, gráficos, cuadros, etcétera, resulta perfecto (estoy tentado a decir *imprescindible*), pero si quieres realizar un póster o una presentación o diseñar una revista o cualquier otro documento que necesite un control manual de todos los elementos será mejor buscar otras opciones. Es decir, es *posible* realizar un póster o una presentación en LyX (\LaTeX cuenta con más de una *documentclass* para presentaciones), pero no resulta difícil el encontrar argumentos para decir que no es la herramienta más adecuada para el trabajo.

A pesar de ser uno de los más viejos sistemas electrónicos de edición de texto, \LaTeX y sus derivados siguen siendo por lejos los más poderosos.

Sus limitaciones son pocas mientras que sus beneficios son enormes, especialmente gracias a LyX que hace fácil las cosas molestas de \LaTeX como la selección de la *documentclass*, la creación de figuras y cuadros y el ajuste de los detalles (márgenes, formato de párrafos) entre otras cosas.

Si bien algunos programas —por ejemplo LibreOffice Writer con su filosofía basada en estilos y su amplio soporte OpenType— se están acercando a las características de \LaTeX , todavía tienen un largo camino por recorrer, especialmente cuando hablamos de documentos técnicos complejos.

El público más amplio de este sistema se encuentra ciertamente entre la gente de ciencia (físicos, matemáticos, astrónomos...) e ingeniería, pero no se limita solo a este especializado grupo de personas: \LaTeX o X_{\LaTeX} + LyX

está abierto a todo aquel que tenga algo para decir a través de un texto y que quiera obtener documentos bellos con (relativamente) poco esfuerzo.

Y aquí llegamos a un punto filosófico: *¿existe algún motivo que nos obligue a preocuparnos por la belleza de nuestros documentos?*

De hecho, si vas a una librería y tomas de un estante un libro cualquiera te encontrarás casi seguramente con una presentación realmente pobre. Y no me refiero solamente a una mala encuadernación, que incluso a esta altura del siglo y de la tecnología muchas de las editoriales populares siguen sin preocuparse tan siquiera de utilizar ligaduras tipográficas, ni hablemos entonces de alternativas contextuales o microtipografía. ¡Incluso he tenido en mis manos libros que presentaban un horrible interletraje¹!

Cada persona tendrá su propia reacción ante este hecho tristemente indiscutible. La mía es *¿y qué?*

¿Existe algún motivo que me obligue a preocuparme por lo que los demás *no hacen*?

Ciertamente no. A mi me gusta que un texto esté bien presentado. Me encantan las ligaduras tipográficas. Adoro la microtipografía. ¿Por qué tendría yo que privarme de utilizar las potentes herramientas que LyX, Xe_{La}TeX y OpenType me ofrecen?

¿Por qué tendrías que privarte tú, estimado lector?

Si no quieres renunciar a vestir de belleza el contenido de tus documentos, acompáñame, que este viaje recién comienza.

1. <https://es.wikipedia.org/wiki/Interletraje>
<https://www.xkcd.com/1015/>

Parte II

Configurando, que es un gerundio

Donde se comienza a trabajar sobre LyX, nivel básico a medio

Don't Panic!

Capítulo 4

La interfaz gráfica de LyX

LyX presenta una típica interfaz de menús, barras de herramientas y paneles. Pero hay mucho más de lo que aparece a simple vista.

4.1 Descripción general de la interfaz

La interfaz gráfica de LyX es bastante tradicional en apariencia, con una serie de menús arriba seguidos por un par de barras de herramientas. Con un clic derecho sobre las barras de herramientas puedes modificar el tamaño de los iconos y en el menú `Herramientas >> Preferencias` tienes varias opciones para modificar la apariencia general del programa.

Pero por supuesto esto es solo el comienzo, tal y como se puede apreciar en la figura 4.1.

Un espacio de la interfaz de LyX que suele ser pasado por alto es la delgada barra que se encuentra justo debajo del área de edición. Aquí LyX presenta información importante de *lo que está pasando*:

- La función del botón de la barra de herramientas sobre la que se encuentra el puntero del ratón.
- El formato del texto sobre el que se encuentra el cursor.
- Las opciones que tenemos para completar un atajo de teclado.
- El comando que se ha apenas utilizado desde algún menú.

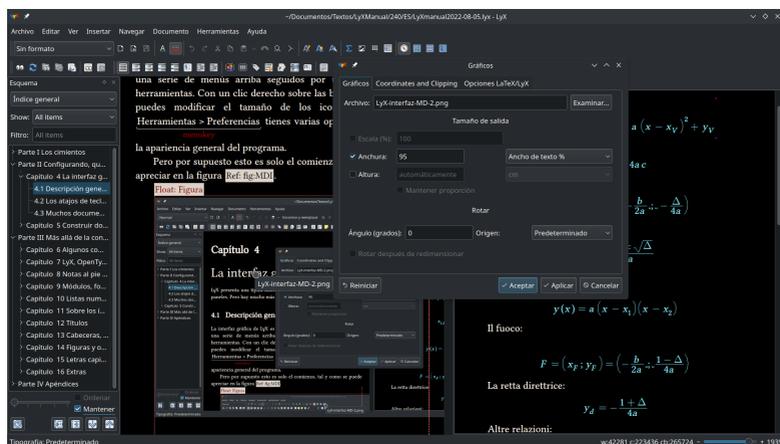


FIGURA 4.1: *LyX* y su interfaz multidocumento mostrando el panel de esquema y un cuadro de diálogo con la configuración de una imagen insertada

- Notificaciones del compilador (advertencias, errores, etcétera).

Algunas de estas cosas pueden verse en la figura 4.2.

La combinación **Ctrl** **F** nos dará la herramienta de buscar y reemplazar básica, mientras que **Ctrl** **⇧** **F** nos dará un panel con opciones de búsqueda avanzadas en el que podremos utilizar expresiones regulares, buscar expresiones matemáticas, etcétera.

No voy a profundizar en la búsqueda avanzada. Si quieres sumergirte en este tema puedes consultar la sección 6.13 del manual que se encuentra en **Ayuda** **»** **Guía del usuario**.

Desde el menú **Ver** **»** **...** tienes acceso a varias opciones.

Hablaré sobre el **PANEL DE ESQUEMA** en la sección 5.1.11.

El **PANEL DE CÓDIGO FUENTE** te muestra en tiempo real el código **TeX** generado por **LyX**.

El **PANEL DE MENSAJES** te muestra los mensajes producidos por **LyX** mientras trabaja. Solo lo necesitarás si encuentras muchos errores a la hora de tratar de compilar el documento.

Bajo **BARRAS DE HERRAMIENTAS** puedes activar o desactivar otras barras. Debes tener en cuenta que la opción *automático*, activa por defecto en

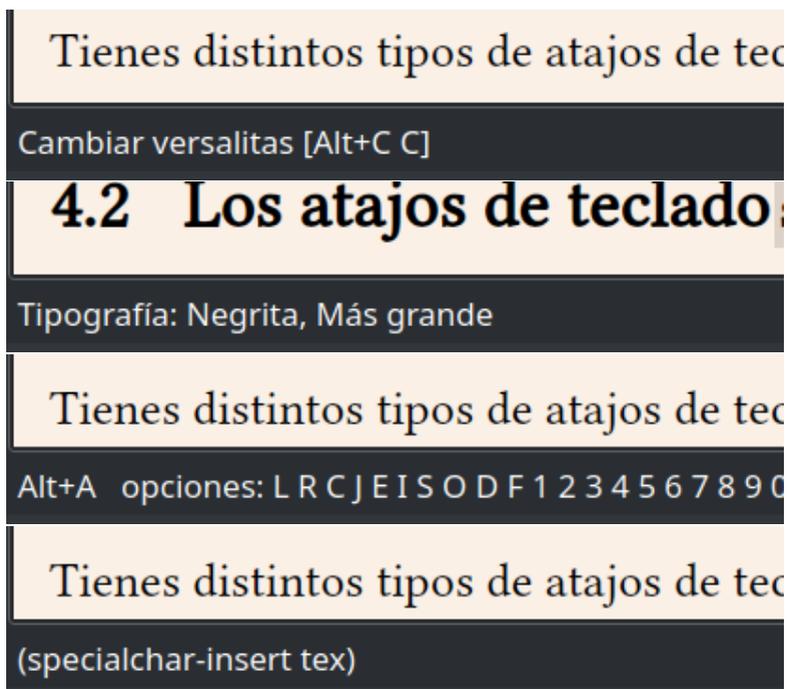


FIGURA 4.2: de arriba a abajo, la primera captura presenta la barra de estado mostrando la función de un botón de la barra de herramientas sobre el que se encuentra el puntero del ratón (si la acción tiene asociada un atajo de teclado, este se mostrará). La segunda captura presenta información del formato sobre el cual se encuentra el cursor. La tercera captura muestra cómo al activar la primera parte de algunos atajos de teclado LyX nos presenta todas las opciones disponibles. En la última captura se ve información relevante a una acción realizada a través de los menús y que no tiene (aún, ver la siguiente sección) un atajo de teclado asignado.

muchas barras como las relacionadas con el editor de ecuaciones, hace que la correspondiente barra se active solo cuando sea necesaria.

4.2 Los atajos de teclado

Tienes distintos tipos de atajos de teclado:

- Los *tradicionales* como **Ctrl** **B** para pasar a negrita.
- Los *secuenciales* donde se utiliza un atajo de activación que luego viene seguido por otras teclas o combinaciones de teclas. Por ejemplo **Alt** **P** seguido de **2** activa el entorno sección.

Tal y como muestra la figura 4.2, comenzando a escribir un atajo secuencial la barra de estado te ofrece las opciones disponibles para completarlo.

Claramente no todas las funciones de LyX tienen un atajo de teclado asignado. Para crear un nuevo atajo para una opción que utilizas mucho debes dirigirte a **Herramientas** **Preferencias** **Edición** **Atajos de teclado**. Allí eliges de la lista la función que te interesa, haces clic en **MODIFICAR**, presionas la combinación elegida y luego aceptas: LyX te avisará amablemente si el atajo seleccionado ya está en uso por lo que solo tendrás que cambiarlo por otro.

Ahora bien, no todas las funciones de LyX están en esta lista, ¿es posible crear un atajo de teclado para esas funciones ausentes? ¡Ciertamente! Aquí tienes que prestar atención nuevamente a la barra de estado que mencioné al final de la sección anterior, ya que como *también* se vio en la figura 4.2, cuando una acción es realizada desde el menú allí se mostrará el código de la misma.

Supongamos que quieres crear un atajo de teclado secuencial para insertar el logo de LyX. Este se encuentra en **Insertar** **Carácter especial** **Logotipos** **LyX logo** y al insertarlo desde allí verás en la barra de estado el siguiente código:

```
specialchar-insert lyx
```

Esto es lo que necesitas para crear el nuevo atajo. Eso sí, ¡atención, que hay que respetar las mayúsculas!

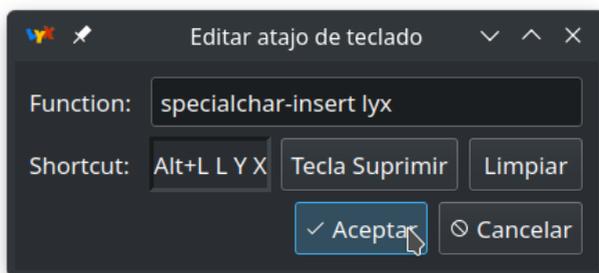


FIGURA 4.3: crear un nuevo atajo de teclado

Vuelve a [Herramientas](#) [»](#) [Preferencias](#) [»](#) [Edición](#) [»](#) [Atajos de teclado](#), pero haciendo clic sobre NUEVO. En FUNCIÓN escribe el código que leíste en la barra de estado y en la casilla correspondiente introduce el atajo, tal y como se ve en la figura 4.3

Este nuevo atajo es un tributo a los atajos secuenciales: **Alt** **L** y luego la secuencia **L**, **Y** y **X**.

Pero puedes hacer más: imagina que quieres un atajo de teclado para insertar el texto, con todo y cursiva, «*typography rulez!*», el comando a insertar sería

```
command-sequence font-emph;
self-insert typography rulez!; font-emph
```

y allí ya lo tienes.

Puedes consultar el manual [Ayuda](#) [»](#) [Funciones de LyX](#) para una extensa lista de instrucciones disponibles.

4.3 Muchos documentos, en muchas formas

LyX permite trabajar simultáneamente con varios documentos separados en pestañas en la misma ventana o con varias instancias del mismo documento en forma de pantalla dividida.

Cada vez que abres un nuevo documento este se agrega en una nueva pestaña, como en los navegadores de internet. Para dividir la vista y tener si-

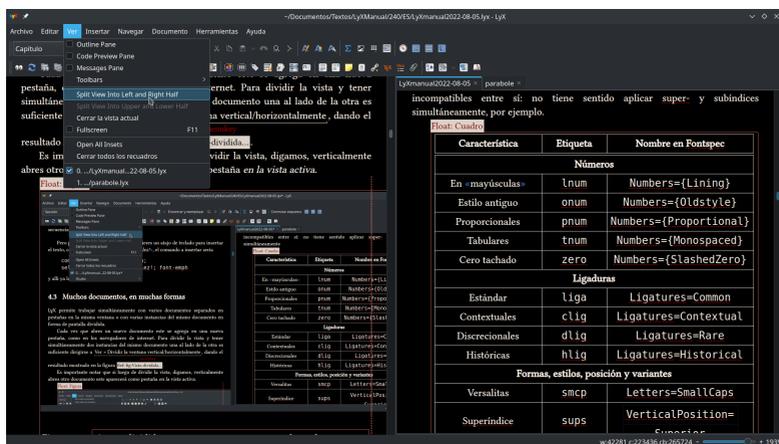


FIGURA 4.4: *vista dividida + pestañas, para todos los gustos*

multáneamente dos instancias del mismo documento una al lado de la otra es suficiente dirigirse a `Ver >> Dividir la ventana vertical/horizontalmente`, dando el resultado mostrado en la figura 4.4.

Es importante notar que si luego de dividir la vista, digamos, verticalmente abres otro documento este aparecerá como pestaña *en la vista activa*.

Te dejaré jugar con estas opciones. Eso sí, no te olvides de probar el clic derecho sobre una pestaña.

Capítulo 5

Construir documentos

En este capítulo te mostraré rápidamente (que para algo está la documentación del programa) cómo crear un documento simple desde cero, elegir su tipo, navegar y modificar su contenido. Presentaré las herramientas básicas que ofrece LyX y cómo trabajar con ellas. Para todo esto supondré ya has leído el capítulo anterior.

No entraré aún en los «detalles L^AT_EX» (que para eso tenemos los capítulos más avanzados), pero lo que aquí se presenta resulta esencial para comprender el manejo del programa.

5.1 Trabajar en LyX

Crear un documento nuevo LyX no es diferente que en otros programas: puedes utilizar , **Ctrl** **N** o el correspondiente botón de la barra de herramientas. Pero claro, eso es solo el comienzo.

Por defecto un documento nuevo de LyX utiliza la *documentclass* Artículo, una de las más simples clases de documento que ofrece L^AT_EX.

5.1.1 Formato general del documento

Todo (bueno, casi, pero ya volveremos sobre esto) lo que necesitas se encuentra en el menú , tal y como puedes ver en la figura 5.1.

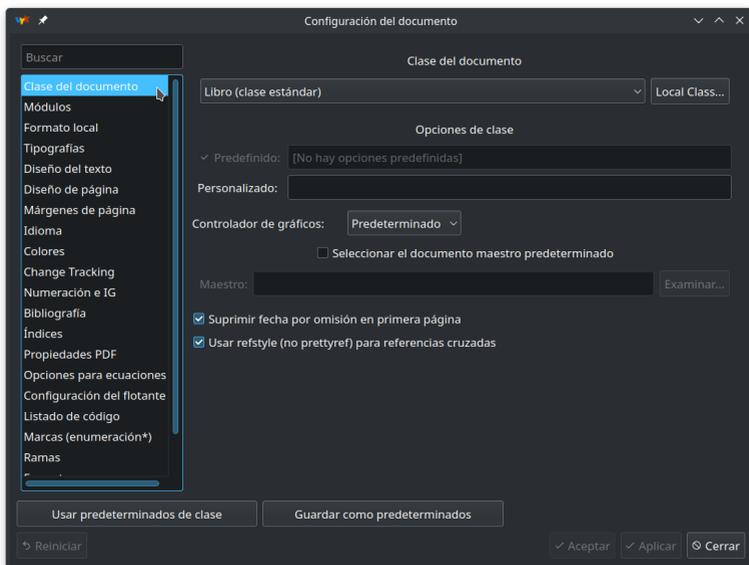


FIGURA 5.1: el menú `Documento` \gg `Configuración`

En la primera voz puedes seleccionar la *documentclass* que te interesa. Desafortunadamente la única forma de poder saber cuál te interesa es crear un documento de prueba para «ver cómo se ve», por lo que no entraré en detalles sobre cómo elegir el tipo correcto.

Eso sí, he aquí un consejo general: un par de *documentclass* están *repetidas*, ofreciendo una versión estándar y otra «con más tamaños de tipografías». Por ejemplo, «Libro (clase estándar)» y «Libro (clase estándar con más tamaños de tipografía)». Olvídate de la segunda.

Si bien volveré a hablar sobre las cuestiones de tamaño en la sección 6.3, veamos en qué se diferencian estas clases.

Si te diriges a la voz TIPOGRAFÍAS, bajo TAMAÑO BASE te encontrarás con una serie de posibilidades. Además de la opción PREDETERMINADO las clases normales ofrecen tres tamaños de fuente para el texto: 10 puntos (generalmente, la opción predefinida), 11 y 12, mientras que las clases con más tamaños ofrecen, pues, más opciones.

Para este documento he utilizado «Libro (clase estándar)» seleccionan-

do el tamaño de 11 puntos, pero las otras opciones son perfectamente válidas. Cada clase de documento viene acompañada de su documentación. Recuerda que CTAN es tu amigo.

En esta primera voz también tienes la opción SUPRIMIR FECHA POR OMISIÓN EN LA PRIMERA PÁGINA que hace... justamente eso. Por defecto en las clases de documento de libro \LaTeX agrega la fecha en la que se compila el documento, lo cual no siempre es lo que se quiere. De hecho, casi nunca es lo que se quiere. Pues bien, con solo marcar esa opción tal y como se ve en la captura de pantalla la fecha no se presenta.

Algo muy importante que también se encuentra allí es el apartado OPCIONES DE CLASE. Esta caja te permite pasar parámetros a la *documentclass* que afecten el comportamiento del documento. Por ejemplo, en las clases de documento de libros, cuando seleccionas la opción A DOS CARAS en la configuración de la página, tienes que por defecto los capítulos comenzarán en páginas impares (derechas): si quieres que los capítulos puedan comenzar en cualquier página puedes agregar en PERSONALIZADO la instrucción `openany`. Las opciones disponibles dependerán de la *documentclass* elegida por lo que te invito a consultar la documentación correspondiente.

Hablaré sobre los MÓDULOS y el FORMATO LOCAL en el capítulo 9.

La voz TIPOGRAFÍAS es, como resulta fácil imaginar, el lugar donde se seleccionan las fuentes tipográficas. El elemento más importante allí es la caja USAR TIPOGRAFÍAS NO TEX (VIA XETEX/LUATEX): esta opción te permitirá seleccionar fuentes OpenType instaladas en el sistema en lugar de las fuentes propias de \LaTeX . Volveré sobre estos temas en el capítulo 7.

En un documento \LaTeX tienes tres familias básicas de fuentes (más adelante te mostraré cómo definir más): la *Romana* que es la fuente base, *palo seco* y *ancho fijo*. Como ya comenté más arriba, la caja TAMAÑO BASE te permite definir el tamaño de la fuente de base mientras que el tamaño de las otras dos familias puede determinarse en forma proporcional al tamaño de base: por ejemplo, en este libro he utilizado una escala del 80% para la fuente de ancho fijo¹.

Las voces DISEÑO DEL TEXTO, DISEÑO DE PÁGINA y MÁRGENES DE PÁGINA son simples... salvo por el interlineado. Pero ese tema tiene sus vueltas por lo que lo discutiré más adelante, también en el capítulo 7.

1. Y es que DejaVu Sans Mono tiene un tamaño *generoso*.

Solo una cuestión a remarcar aquí, importante para quien venga de trabajar con LibreOffice Writer: en un documento \LaTeX /LyX los márgenes indican la distancia del borde de la página al área de texto efectiva, por lo que las cabeceras y pie de página caen *dentro* del margen. Es por esto que en la voz MARGENES DE PÁGINA existen entradas específicas para configurar la posición y la altura de las cabeceras o pie de página.

Sobre la parte del idioma hablaré más abajo.

La voz COLORES te permite seleccionar, bien, los colores del documento (texto, notas, fondo de página).

NUMERACIÓN E IG se refiere a hasta dónde queremos numerar la estructura del documento. Por defecto en las clases de libros se tiene numeración automática para Parte, Capítulo, Sección, SubSección y SubSubSección, pero los párrafos no son numerados. Además, el índice general muestra por defecto solo los niveles que tienen numeración. Pues bien, deslizando las dos barras superiores puedes regular qué se numera y qué no, o qué se muestra en el índice y qué no aparece.

Es importante notar que en LyX puedes utilizar plantillas: como puedes ver en el menú  puedes *cargar* una plantilla o bien *crearla*. Esto quiere decir que si quieres reutilizar las opciones elegidas para un documento particular tienes dos opciones:

- El botón GUARDAR COMO PREDETERMINADO que tienes en el menú de configuración del documento hará que la configuración elegida se aplique a *todos* los documentos nuevos.
- Salvar una plantilla vacía por cada grupo de opciones (una para los libros, otro para los artículos, etcétera) y luego llamar esos documentos desde .

LyX ya viene con varias plantillas predefinidas para crear documentos especiales (publicaciones específicas como *Springer* y *Scientific American*, por ejemplo). Una de estas plantillas, PDF (recortado) bajo GENERAL, te permite crear un documento PDF a partir de una fórmula matemática con el tamaño exacto de esa fórmula matemática, que luego podrás utilizar como imagen en otras aplicaciones.

En próximos capítulos hablaré de algunas de las otras voces del menú de configuración. Discutiré las PROPIEDADES PDF y las RAMAS al final de este

capítulo, la parte de **FORMATOS** en el capítulo 7 y el **PREÁMBULO L^AT_EX...** en muchos lugares.

5.1.2 Texto

Ya has elegido fuentes, página, márgenes y demás, ¡a escribir!

Un concepto fundamental a la hora de utilizar L^AX es el de *entorno*. Con el menú (o con atajos de teclado) que se encuentra a la izquierda de la barra de herramientas principal, justo debajo del menú **ARCHIVO**, puedes asignar a cada párrafo la *función* que quieras darle: título de capítulo, de sección, texto normal, lista numerada, etcétera.

Todo está perfectamente clasificado y comprensible por lo que te dejaré la tarea de jugar un poco con esas opciones para acostumbrarte a sus funciones. Lo único importante a comentar es que los elementos *sin numerar* (los que tienen el asterisco) no serán incluidos en el índice: volveré sobre esto en el capítulo 13.4.

De esta forma el documento mostrará una estructura bien definida y coherente.

L^AX ofrece algunos estilos de texto predefinidos: **Ctrl** **E** activa y desactiva la *cursiva* (el comando L^AT_EX `\emph{}`), **Ctrl** **B** activa y desactiva la **negrita** (el comando L^AT_EX `\textbf{}`) mientras que la secuencia **Alt** **C**, luego **C** hace lo propio con las **VERSALITAS** (`\noun{}`). Estas tres opciones también pueden manejarse desde la barra de herramientas.

NOTA: para volver al formato predeterminado, además de volver a aplicar el estilo al texto (piensa al acto de aplicar el formato como un interruptor que lo enciende o lo apaga) tienes la combinación **Alt** **C**, luego .

Para esas raras situaciones en las que el formato directo pueda ser necesario considera estos menús: **Editar** **>>** **Propiedades del texto** **>>** **Personalizado** y **Editar** **>>** **Configuración del párrafo**, ambos accesibles desde el clic derecho.

Para introducir el guion mediano – y la raya — (lo que en inglés se llaman *n-dash* y *m-dash*) es suficiente escribir el guion normal dos y tres veces seguidas, respectivamente

5.1.3 Listas

Directo desde la barra de herramientas o desde el menú desplegable Entorno puedes elegir entre los entornos Enumeración (**Alt** **P**, luego **E**), Itemizado (**Alt** **P**, luego **B**), Etiquetado (**Alt** **P**, luego **L**) y Descripción (**Alt** **P**, luego **D**).

Las dos primeras son las clásicas «listas numeradas» (la primera) y «viñetas» (la segunda) a las que otros procesadores de texto nos han acostumbrado. Estas pueden anidarse y mostrar varios niveles. Para pasar de un nivel al otro puede utilizarse ya sea la barra de herramienta o los atajos de teclado **Alt** **⇧** **→** y **Alt** **⇧** **←** para subir un nivel y para bajarlo, respectivamente. Algo así:

- Primer nivel
 - Segundo nivel
 - Otro segundo nivel
 - ◊ Tercer nivel
 - Un nuevo segundo
- Un nuevo primero

Para salir de la lista es suficiente seleccionar el entorno «normal» o presionar **↵** dos veces.

Tanto los etiquetados como las descripciones separan una «palabra a definir» del texto que la definen, cambiando solamente en los detalles del formato del párrafo y cómo la descripción viene separada de la palabra clave. Te dejo jugando con estas opciones. Solo una cosa: si quieres más de una palabra en la etiqueta tendrás que unir las con espacios de no separación (**Ctrl** **[**) para que LyX las considere como una única palabra.

LyX te permite configurar en forma realmente simple las viñetas de tu documento. Para esto simplemente te diriges a **Documento** **»** **Configuración** **»** **»** **Marcas (enumeración*)** donde podrás seleccionar el símbolo para los cuatro posibles niveles, ya sea de las seis tipografías propuestas o insertando tus propios símbolos. También puedes cambiar el tamaño de los símbolos seleccionando una de las opciones que va desde *diminuta* hasta *gigante*².

2. Volveré al significado de esos nombres en el capítulo 7, donde hablaré sobre tipografía.

Configurar las listas numeradas no es tan inmediato, por lo que lo dejaré para la última parte de este libro, en el capítulo 10.

5.1.4 Matemáticas

Existen dos estilos para mostrar expresiones matemáticas: *en línea* y *presentada*, esta última pudiendo ser numerada o no.

La diferencia principal entre estos dos estilos es que las expresiones en línea presentan signos (integral, sumatoria) más pequeños para integrarse mejor como parte de un párrafo. Compara cómo se ve esta expresión en línea $\int_a^b f(x)dx$ con esta otra presentada:

$$\int_a^b f(x)dx$$

La forma en línea es accesible desde la barra de herramientas principal mientras que ambas formas pueden insertarse desde el menú **Insertar** **Ecuación**, si bien la forma más simple de insertar expresiones matemáticas es seguramente con los atajos de teclado **Ctrl** **M** para las expresiones en línea y **Ctrl** **⇧** **M** para las presentadas. Estos atajos de teclado pueden utilizarse para transformar un tipo de expresión en otra: presionando **Ctrl** **⇧** **M** dentro de una expresión en línea esta se transformará en una presentada.

Las ecuaciones numeradas pueden crearse desde **Insertar** **Ecuación** **Numerada** o bien puedes transformar una expresión presentada con **Alt** **M**, luego **N** (esta combinación también quita la numeración):

$$\int_a^b f(x)dx \tag{5.1}$$

Más adelante en este capítulo te mostraré cómo realizar referencias cruzadas a expresiones matemáticas numeradas.

Por defecto las expresiones presentadas vienen centradas en la página, pero LyX ofrece soporte nativo para la instrucción `mathindent`, lo cual facilita el lograr que las expresiones estén sangradas en lugar de centradas. Para esto es suficiente ir a **Documento** **Configuración** **Opciones para ecuaciones** y

modificar la opción correspondiente. Allí también resulta posible seleccionar de qué lado de la ecuación se presentará el número.

En la figura 3.1 en la página 20 vimos al editor de ecuaciones en pleno trabajo mostrando cómo pueden utilizarse menús, botones o código puro. Claramente muchísimos atajos de teclado se encuentran también disponibles. Muchos de ellos comienzan con la combinación **Alt** **M**: **Alt** **M**, luego **G**, luego una letra nos dará un carácter griego, **Alt** **M**, luego **F** una fracción, **Alt** **M**, luego **(** los paréntesis escalables, **Alt** **M**, luego **L** pasa los límites de una integral de la derecha a que se vean arriba y abajo (la instrucción `\limits`), etcétera. Al pasar el puntero del ratón sobre los botones de las barras de herramientas podrás ver los atajos correspondientes para así ir aprendiéndolos cómodamente.

Las posibilidades que ofrece el editor de ecuaciones de LyX son casi infinitas y realmente no tiene sentido que aquí me dedique a duplicar lo que ya está magníficamente explicado en la documentación del programa, por lo que luego de esta breve introducción solo me queda invitarte a consultar el manual que se encuentra en **Ayuda** **»** **Ecuaciones**: no te preocupes por sus más de ochenta páginas que la lectura es amable y rápida.

5.1.5 Cuadros, ¡que no se llaman tablas!

Antes de entrar en el tema específico de este apartado, una declaración importante: la relación que en castellano existe entre las palabras *cuadro* y *tabla* es la misma que en inglés existe entre *table* y *tabular*. En castellano se puede hablar de *la tabla periódica de los elementos*, pero si tienes datos tabulados con una leyenda numerada, pues que eso es un *cuadro*.

En fin, bien sé que es esta una batalla perdida. Te contaré más sobre los cuadros al hablar de *flotantes* en el apartado 5.1.7), ahora solo te mostraré cómo crear los datos tabulados que irán dentro de esos cuadros.

A lo nuestro: ya sea desde la barra de herramientas estándar o desde el menú **Insertar** **»** **cuadro/tabla** te encontrarás con las clásicas opciones para elegir el número de filas y columnas que te sirven.

Con el cursor dentro del cuadro/tabla tendrás a disposición en la parte baja de la pantalla de edición la barra de herramientas de cuadros, la cual te permite insertar o quitar filas o columnas, unir celdas (multicolumnas o multifilas, los dos últimos botones a la derecha), seleccionar bordes, girar

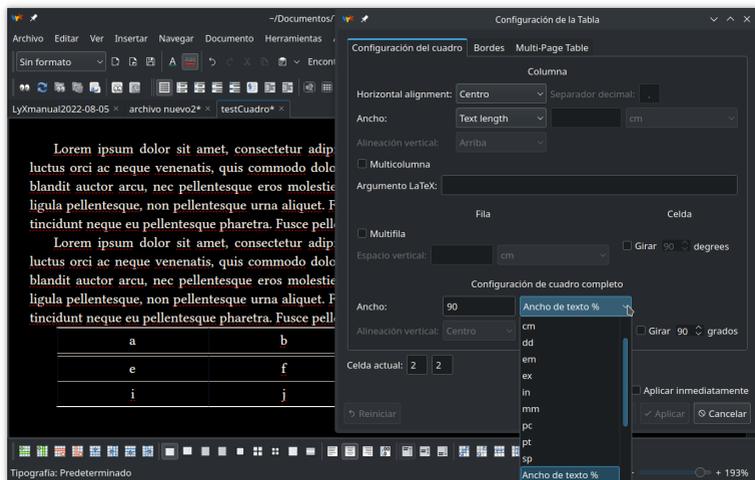


FIGURA 5.2: configurar el ancho de un cuadro

una celda o todo el cuadro de 90°, etcétera.

Los cuadros/tablas pueden *anidarse*: colocando el cursor en una celda puedes insertar otro cuadro.

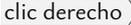
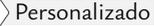
Con un clic derecho \gg Configuración tienes muchas más opciones como girar el contenido de una celda en un ángulo arbitrario o configurar un cuadro multipágina cuyo contenido pueda dividirse en varias páginas en forma automática.

En la pestaña BORDES de la configuración del cuadro tienes a disposición varios estilos de líneas. Es posible configurar un estilo predefinido para todos los cuadros nuevos en Documento \gg Configuración \gg Diseño del texto.

Ciertamente puedes crear el formato que quieras para tu cuadro, y a partir de 2.4 es posible definir nuevos estilos.

Algo importante de recordar es que también puedes modificar el ancho del cuadro: por defecto tendrá el ancho mínimo necesario para acomodar el contenido, pero puedes fácilmente hacer que ocupe un porcentaje determinado del área del texto, por ejemplo, como puede verse en la figura 5.2. De hecho, fijado el ancho del cuadro/tabla podrás luego fijar los anchos relativos de cada columna.

Ahora bien, ¿cómo se logra dar *color* a una tabla?

Puedes cambiar el color *del texto* con un    o bien utilizando un estilo de carácter (volveré sobre eso en el capítulo 9). ¿Pero y el *color de fondo* de una celda? Para eso se necesita el paquete `LATEX colortbl` y un poco de trabajo: todo está explicado en la sección 2.11 del manual que encuentras en  , pero antes de intentarlo lee la tercera parte de este libro, que en este tema no te podrás escapar del código `LATEX` puro y duro.

5.1.6 Marcos

En el menú   tienes varias opciones (sin borde, borde sencillo, oval...), la pregunta, claro está, es qué hace cada una de ellas.

Las opciones presentadas en ese menú insertan algo que en `LATEX` se llama *minipage* (minipágina): básicamente te da una pequeña página flotando sobre la página real. Esta pequeña página puede tener algún borde o incluso un color de fondo. Es importante notar que estos marcos no se dividen por lo que si el contenido no cabe en la página real tendrás algunos problemas. El uso de estos marcos es por lo tanto el de insertar notas, por ejemplo, o pequeñas aclaraciones que complementen el contenido principal.

Ahora bien, si lo que hace   es insertar una minipágina, ¿no debería llamarse «insertar minipágina»? La respuesta es, claramente, que no, que tenemos más opciones.

Con un clic derecho en la etiqueta de la caja que indica el marco obtienes un menú que no solo te permite cambiar entre las opciones disponibles, sino que también te da acceso a la configuración de ese marco.

En este menú de configuración puedes cambiar varias cosas, como el ancho del marco, etcétera. Pero por sobre todas las cosas puedes elegir el tipo de *marco interior*, tal como se ve en la figura 5.3.

Tienes tres opciones allí:

MINIPÁGINA Lo ya discutido más arriba

NINGUNO Para textos sumamente breves que deben ocupar solamente una línea

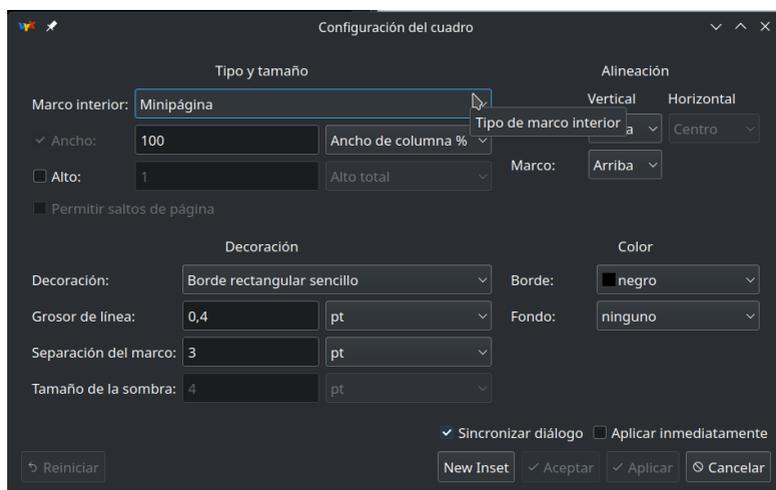


FIGURA 5.3: configuración de un marco

MARCO DE PÁRRAFO Para enmarcar un párrafo particular

Las opciones de este menú son claras, por lo que otra vez te dejaré jugar un poco con ellas.

5.1.7 Figuras, cuadros y otros *flotantes*

En programas como LibreOffice Writer, si quieres una imagen o un cuadro (que ya dije que no son tablas) con una leyenda, generalmente insertas primero la imagen y luego la leyenda. Pues bien, en LyX esto se hace exactamente al contrario.

Ya sea con el correspondiente botón de la barra de herramientas o desde el menú **Insertar >> Flotante >> ...** puedes insertar un *flotante* en el cual escribir la leyenda e insertar luego la imagen o el cuadro.

Estos flotantes hacen exactamente lo que sus nombres indican: *flotar* sobre la página. A dependencia del espacio disponible en la página estos flotantes se colocarán ya sea en la parte alta o baja de la página o pasarán a la página siguiente (si no hay espacio) permitiendo al texto fluir cómodamente alrededor de ellos sin mayor trabajo por parte del usuario.

El comportamiento de los flotantes puede ser configurado ya sea para cada flotante en particular (clic derecho sobre la etiqueta del mismo) como en forma general para el documento desde Documento » Configuración » Configuración del flotante. Generalmente dejar que L^AT_EX haga este trabajo por nosotros (usar ubicación predeterminada) es una buena idea ya que lo hace *realmente* bien.

Ahora bien, LyX te permite elegir fácilmente si quieres que la imagen o el cuadro esté por encima o por debajo de su título simplemente moviendo el cursor hacia arriba o hacia abajo de este último, tal y como muestra en la parte superior de la figura 5.4.

Otra cosa que muestra la figura 5.4 es la posibilidad de trabajar con subflotantes: figuras (o cuadros) dentro de figuras, numeradas como «figura 3A», etcétera. Simplemente insertas flotantes dentro del flotante principal hasta satisfacer tus necesidades. Estos subflotantes pueden ser puestos en cualquier posición: uno arriba y otro abajo, uno al lado del otro... otra vez, te dejo la responsabilidad de explorar esta posibilidad³.

Es importante notar que si quieres un flotante *sin* leyenda simplemente tienes que borrar la caja correspondiente. Si más tarde quieres reintroducirla, con el cursor en el flotante te diriges a Insertar » Leyenda.

Te contaré más sobre figuras y otros flotantes en el capítulo 14.

5.1.8 Referencias cruzadas

LyX no cuenta con un mecanismo de etiquetado automático por lo que para realizar una referencia cruzada, ya sea esta al título de un capítulo, a una ecuación o a una figura numerada tendrás que realizar dos simples pasos.

Primero tienes que crear la etiqueta, ya sea con el botón INSERTAR ETIQUETA o desde Insertar » Etiqueta, junto (o, en el caso de los flotantes, *dentro*) al elemento al cual quieres hacer referencia. Con la etiqueta lista, ahora puedes llamarla con la opción INSERTAR REFERENCIA CRUZADA, otra vez desde el menú o del correspondiente botón.

Como vimos en la introducción, las etiquetas pueden ser usadas en cualquier tipo de contenido: figuras, cuadros, ecuaciones numeradas, capítulos, partes o secciones, notas al pie de página, lo que sea necesario. LyX compren-

3. Acostúmbrate, que esta es la tendencia.

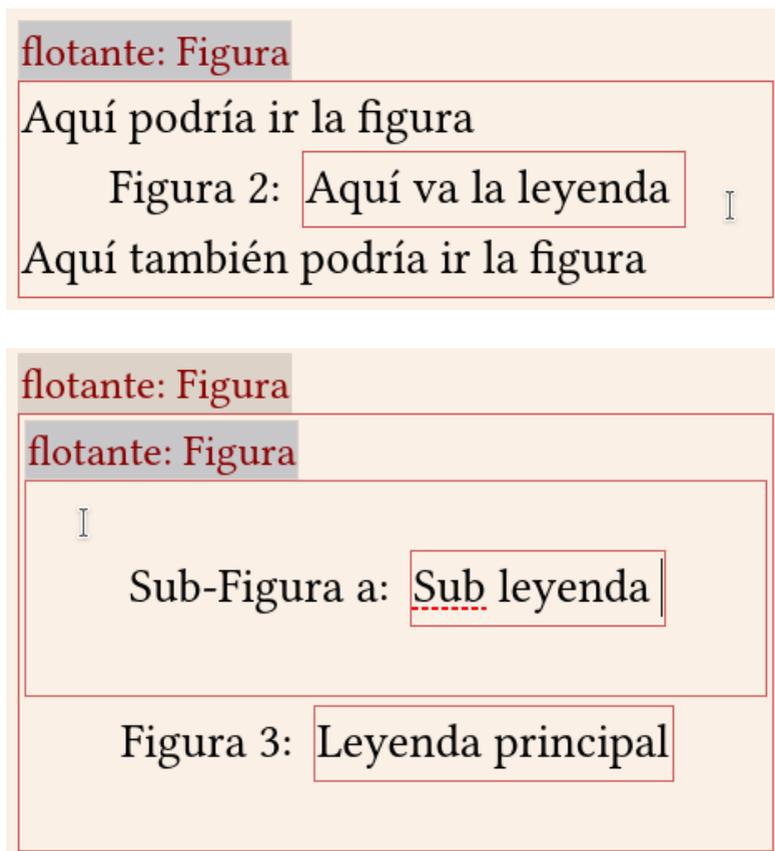


FIGURA 5.4: trabajar con flotantes. En la parte superior de la figura se ve cómo se presenta el flotante en la ventana de edición de LyX, indicando con texto dónde podría colocarse la imagen. En la parte inferior de la figura se ve que resulta muy simple trabajar con subfiguras simplemente insertando los flotantes necesarios dentro del flotante principal. Como puede verse en esta figura, no me agradan las subfiguras.

derá sin problemas qué tipo de contenido se está etiquetando por lo que en la referencia cruzada la numeración será siempre correcta.

Es importante al insertar la etiqueta darle un nombre que luego puedas reconocer sin problemas, por lo que resulta conveniente aceptar el esquema que te ofrece LyX por defecto anteponiendo unas pocas letras para saber si estamos llamando a un capítulo, figura, ecuación, etcétera.

Ahora bien, supón que se ha cambiado el título de un capítulo previamente etiquetado, o que se transformó una subsección a sección, *¿es posible cambiar el nombre de la etiqueta?* ¡Por supuesto! *¿Y sin temor a arruinar las referencias cruzadas a la misma?* ¡Faltaba más! Con un simple clic sobre la etiqueta puedes cambiar más tarde y en total tranquilidad su nombre por otro más apropiado que las referencias cruzadas a la misma se actualizarán automáticamente.

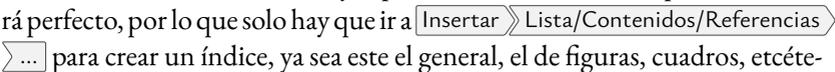
El menú que se presenta al insertar las referencias es más que claro, a continuación solo indicaré sus características más interesantes:

- En la parte superior te permite elegir entre los documentos que tengas abiertos, lo cual simplifica el trabajo de realizar referencias cruzadas entre documentos.
- Sobre la lista de etiquetas disponibles tienes un filtro que facilita el encontrar la que buscas.
- En la parte inferior tienes varias opciones para las referencias (número o texto, la página, ambas cosas...).

En la figura 5.5 puedes ver ambos diálogos, el de crear la etiqueta y el de llamarla.

5.1.9 Índices

Configurar un índice es quizás una de las tareas más pesadas en L^AT_EX. Existen varias formas de hacer esto y por si fuera esto poco incluso varios métodos de crear uno.

Afortunadamente en la mayor parte de los casos el índice por defecto será perfecto, por lo que solo hay que ir a  para crear un índice, ya sea este el general, el de figuras, cuadros, etcétera.

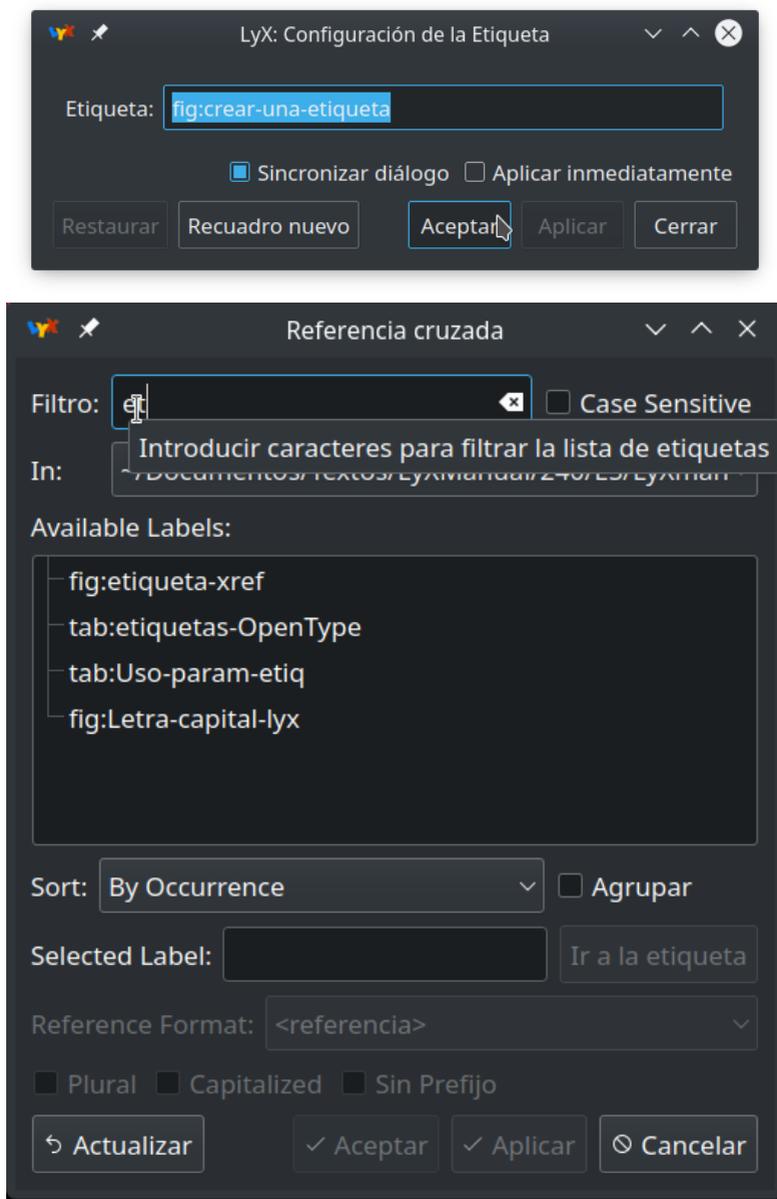


FIGURA 5.5: crear una etiqueta (arriba) e insertar una referencia (debajo)

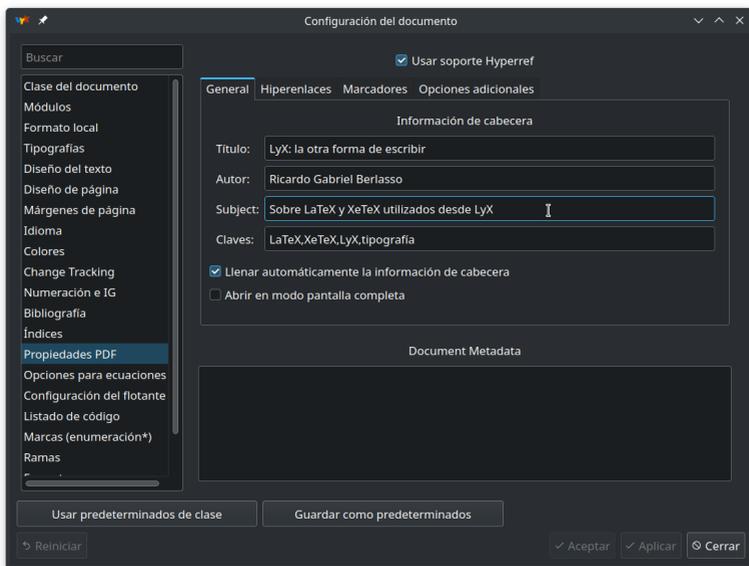


FIGURA 5.6: configurar el comando `Hyperref`

Siendo la configuración de los índices un tema avanzado de \LaTeX volveremos sobre esto en la parte final de este libro, en el capítulo 11.

5.1.10 Generar PDF *dinámicos* con el paquete `Hyperref`

El paquete `hyperref` permite configurar la salida a PDF de documentos \LaTeX de modo tal que se tengan enlaces vivos, ya sea dentro del documento, a sitios web o a otros archivos.

LyX permite configurar varias opciones del paquete `Hyperref` simplemente seleccionando algunas casillas de control, mientras que las otras opciones (como el elegir los colores particulares de los enlaces) se pueden configurar escribiendo algunos parámetros en una línea de texto.

Si te diriges a `Documento` \gg `Configuración` \gg `Propiedades PDF` encontrarás algo parecido a lo que se muestra en la figura 5.6.

Como puedes ver, la pestaña `GENERAL` te permite no solo activar el paquete `Hyperref` para ese documento, sino también establecer las propie-

dades generales del PDF que se generará, como el nombre, autor, si el PDF debe abrirse a pantalla completa, etcétera.

Desde la pestaña HIPERENLACES puedes acceder a algunas opciones del paquete `Hyper ref`, como colorear los enlaces... pero no verás una opción gráfica para seleccionar los colores.

En la pestaña llamada OPCIONES ADICIONALES, bajo `HYPERSETUP`, es donde deben escribirse, separadas por comas, las opciones de `Hyper ref` que no pueden configurarse gráficamente.

Como ejemplo, luego de marcar ENLACES COLOREADOS en la pestaña HIPERENLACES puedes determinar los colores de cada tipo de enlace utilizando las siguientes instrucciones:

citecolor fija el color de los enlaces a la bibliografía

filecolor fija el color de los enlaces a archivos

linkcolor fija el color de los enlaces internos del documento (referencias cruzadas, índice)

urlcolor fija el color de los enlaces a direcciones web

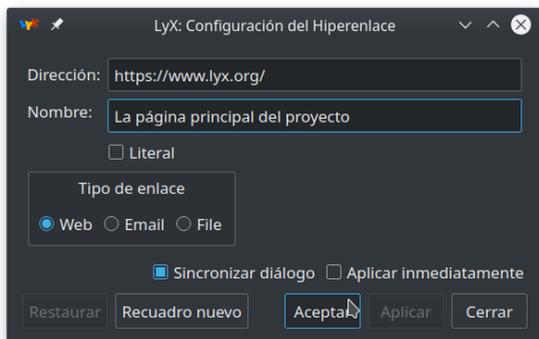
Los colores pueden indicarse ya sea por nombre (en inglés: *red*, *green*, *cyan*, *magenta*, *black*) o por valores RGB con tres números entre cero y uno. Yo suelo marcar la opción «sin marcos alrededor» ya que detesto que los enlaces estén recuadrados, pero si te gusta es posible determinar también el color de los marcos con instrucciones del tipo `linkbordercolor` y similares.

La documentación del paquete `Hyper ref` suele instalarse en `/usr/share/texmf/doc/latex/hyperref/manual.pdf`
Escribiendo algo como

```
citecolor=cyan,filecolor=green,linkcolor=red,  
urlcolor=blue
```

tendrás un PDF no solo *vinculado* sino también *colorido*.

Algo importante para marcar en la pestaña HIPERENLACES es la opción QUEBRAR ENLACES EN LÍNEAS, ya que de lo contrario un texto que contenga un enlace podría seguir y atravesar impunemente el margen derecho.

FIGURA 5.7: *crear un enlace en el documento*

En la pestaña MARCADORES no debes olvidar de tener marcado no solo GENERAR MARCADORES (IG) sino también la opción MARCADORES NUMERADOS: de esta forma el índice del PDF estará numerado apropiadamente.

Otro detalle importante a notar es que con la configuración indicada hasta ahora, en el índice general solo los títulos serán convertidos en enlaces, no los números de página. Agregando en la pestaña OPCIONES ADICIONALES la instrucción `linktocpage`, los números serán convertidos en enlaces, *pero los títulos dejarán de serlo*. Para tener ambos, títulos y números de página como enlaces (pero no los «puntos guía»), la opción a utilizar es `linktoc=all`.

Para tener *todo* el renglón como enlace... pues que no es sencillo. Las posibles *soluciones* (digamos, trucos sucios) dependen de la clase de documento utilizada, por lo que no hablaré de ello en este libro.

Y es que la configuración de los índices es *complicada*.

Como último comentario, recuerda que en LyX existe una herramienta gráfica para insertar enlaces externos, por lo que no es necesario utilizar una caja T_EX para eso (ver figura 5.7).

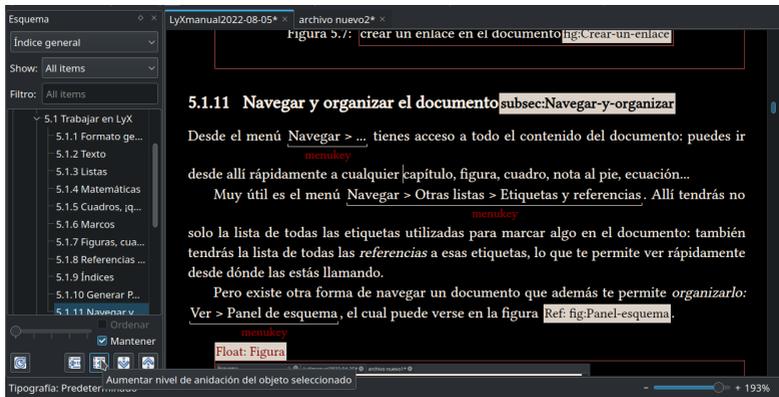


FIGURA 5.8: *el panel de esquema durante la edición de este libro. La casilla «Mantener» resulta muy útil cuando es necesario ir y venir por el documento mientras que los cuatro botones de abajo a la derecha permiten modificar rápidamente tanto el orden como la jerarquía de los capítulos, secciones y demás elementos del documento.*

5.1.11 Navegar y organizar el documento

Desde el menú **Navegar > ...** tienes acceso a todo el contenido del documento: puedes ir desde allí rápidamente a cualquier capítulo, figura, cuadro, nota al pie, ecuación...

Muy útil es el menú **Navegar > Otras listas > Etiquetas y referencias**. Allí tendrás no solo la lista de todas las etiquetas utilizadas para marcar algo en el documento: también tendrás la lista de todas las *referencias* a esas etiquetas, lo que te permite ver rápidamente desde dónde las estás llamando.

Pero existe otra forma de navegar un documento que además te permite *organizarlo*: **Ver > Panel de esquema** (o el botón correspondiente), el cual puede verse en la figura 5.8.

Como muestra la captura de pantalla, en la parte superior puedes elegir qué tipo de información quieres navegar o modificar (el índice general, la lista de figuras, etcétera). Debajo de esta opción tienes un campo que te permite filtrar la lista que sigue, la cual está organizada jerárquicamente en una vista en árbol.

Es importante notar que haciendo clic sobre un elemento de la lista el

punto de edición se moverá allí, *pero también colapsará esa rama de la jerarquía*. Para evitar que esto suceda tienes el botón «Mantener» que hace que las ramas puedan abrirse o cerrarse solo haciendo clic en las flechas a la izquierda de la entrada.

La barra deslizable a continuación sirve para elegir en forma global hasta qué nivel las ramas estarán abiertas.

Finalmente, los botones de abajo a la derecha (que solo estarán disponibles para el índice general) te permiten cambiar el nivel de un título o mover el mismo sobre o debajo de otro del mismo nivel, permitiendo cambiar el orden de las entradas del documento en forma rápida y simple.

5.2 Apéndices

Con   se crea una nueva parte del documento, indicada en la ventana de edición de LyX con una caja roja que dice «apéndice» en la línea superior, que contendrá los apéndices.

Para insertar un apéndice o un subapéndice simplemente tienes que utilizar los entornos Capítulo y Sección, respectivamente.

Las referencias cruzadas a los apéndices y a todo lo que esté en ellos funcionará correctamente sin más trabajo.

Es importante notar que por defecto (¿o por definición?) los apéndices van al final del documento. Es *posible* crear apéndices «por capítulo» o «por parte», pero aún no he encontrado un motivo lo suficientemente bueno como para preocuparme de hacerlo.

5.3 Bibliografía

En un documento LyX tienes dos formas de trabajar con referencias bibliográficas: interna al documento o utilizando una base de datos bibliográfica externa como BIB_{TEX}.

LyX ofrece soporte para BIB_{TEX}, pero entrar en tales temas realmente se escapa del propósito general de este libro por lo que solo daré una breve introducción al método interno. Para conocer el otro método puedes consultar la sección 6.5 de la guía del usuario.

Todo comienza con el entorno «bibliografía». Tienes, luego de insertarlo, una nueva sección llamada Referencias y una primera entrada vacía con una etiqueta en gris llamada `key-1[]`. Con un clic sobre esa etiqueta podrás cambiar la clave por algo más descriptivo o darle un nombre que será utilizado para mostrar la cita en el documento: si la clave se deja en blanco, la cita será numerada.

Es importante notar que en ese diálogo no se aceptan espacios o caracteres no ASCII.

A continuación de la etiqueta es donde debes escribir el texto de la referencia, dándole formato a mano (itálica para el título, negrita para el número de página, etc.).

Finalmente debes ir a `Insertar >> Cita` para seleccionar la(s) cita(s) que necesites, seleccionar el formato, agregar texto a la cita (por ejemplo, [5, capítulo 6]), etcétera.

Dos cosas importantes a notar sobre las referencias así insertadas:

- Siendo un entorno siempre podrás salir del mismo para agregar otro entorno como un capítulo o sección *después*. Esto significa que podrías crear un grupo de referencias nuevo por cada capítulo o sección del documento.
- La lista de referencias quedará en el exacto orden en la cual fue introducida, por lo que es necesario tener cuidado de insertar las entradas en el orden correcto. Recuerda que siempre puedes mover párrafos hacia arriba o abajo con un simple clic derecho. Este problema se resuelve utilizando una base de datos como `BIBTEX`, pero ya dije antes, no entraré en este tema.

5.4 Idioma

En `Documento >> Configuración >> Idioma` encontrarás unas pocas opciones, pero todas ellas de suma importancia. En esta sección te contaré sobre cada una de ellas.

5.4.1 «Comillas “anidadas” dinámicas»

Muchos idiomas, y el castellano no es excepción en esto, pueden utilizar comillas anidadas como las que puedes ver en el título aquí arriba. Cada idioma tiene sus reglas para establecer cuáles son las combinaciones correctas⁴ y desde este menú LyX nos permite configurar estas combinaciones fácilmente.

Las comillas externas se insertan normalmente (en el teclado español, con  ), mientras que para insertar las comillas internas solo se necesita agregar  a la combinación de teclas. Seleccionando la opción UTILIZAR COMILLAS DINÁMICAS, las comillas se adaptarán automáticamente si luego cambias el estilo seleccionado en forma global: sin marcar esta opción, quedarán fijas.

Con las comillas dinámicas activas podrás también cambiar el estilo de comillas en forma manual para un caso particular con un simple clic derecho.

5.4.2 Diccionarios

Para los usuarios Linux probablemente los diccionarios necesarios para corregir la ortografía estén ya instalados (los habrá arrastrado LibreOffice), pero podrían faltar los diccionarios de sinónimos. O quizás quieras utilizar otros diccionarios más actualizados que los que ofrece tu distribución. Incluso podría darse que alguien tenga que utilizar LyX en otros sistemas operativos.

En esta sección verás cómo obtener diccionarios actualizados y cómo utilizarlos en LyX.

Primero, tienes que descargar las extensiones de los diccionarios, por ejemplo desde la página de las extensiones de LibreOffice⁵, o bien, si quieres alguna de las variantes locales del español, desde el repositorio del proyecto «Recursos lingüísticos abiertos del español» (RLA-es) en GitHub⁶. El siguiente paso es cambiar los nombres de estas extensiones de <diccionario>.oxt a <diccionario>.zip para a continuación descomprimirlos, lo que

4. <https://es.wikipedia.org/wiki/Comillas>

5. <https://extensions.libreoffice.org/extensions?getCategories=Dictionary>

6. <https://github.com/sbosio/rla-es/releases/>

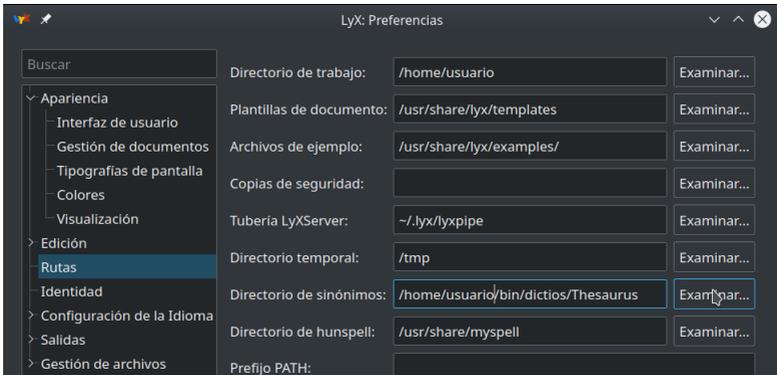


FIGURA 5.9: *modificar las rutas a los diccionarios*

te dará varios archivos y posiblemente algunas carpetas. De todo esto debes preocuparte solo por los archivos con las siguientes extensiones:

- **.dic** y **.aff**, que son los archivos que contienen los diccionarios de corrección ortográfica.
- **.dat** y **.idx**, los archivos que contienen los diccionarios de sinónimos.

Copiando estos archivos en una carpeta cualquiera, abres LyX y te diriges a **Herramientas** » **Preferencias** » **Rutas** como se ve en la figura 5.9.

Ahora LyX usará los diccionarios que copies a los directorios elegidos. En la captura solo he cambiado la carpeta de los diccionarios de sinónimos, pero cambiar el directorio de los diccionarios ortográficos funciona de la misma forma.

En el menú **Preferencias** » **Configuración de idioma** puedes elegir que la corrección ortográfica marque los errores encontrados en tiempo real, mientras que el atajo de teclado **⇧ F7** te abrirá el diccionario de sinónimos en la palabra sobre la cual se encuentre el cursor. Es importante notar que si tienes los archivos **.aff**, el diccionario de sinónimos reconocerá palabras modificadas (plurales, verbos conjugados, etcétera), si bien solo dará sugerencias sin modificar (ofrecerá singulares, verbos en infinitivo, etcétera).

5.4.3 Babel vs. Polyglossia

La definición del idioma del documento afecta a mucho más que a la corrección ortográfica: desde cuestiones inmediatas como llamar a los capítulos *Capítulo* y no *Chapter* hasta temas más sutiles como decidir si luego de un título se mantiene la sangría de primera línea son todos temas altamente relacionados al lenguaje utilizado.

Históricamente \LaTeX ha siempre usado un programa externo llamado Babel para tomar esas decisiones. El problema estaba en que, también históricamente, Babel solía tener problemas con lenguajes que utilizan sistemas de escritura distintos del latino.

Esta situación empeoró con la llegada de Unicode, lo que llevó a que surgiera un proyecto alternativo llamado Polyglossia. De hecho, en documentos que utilizan las fuentes del sistema LyX utiliza por defecto Polyglossia en lugar de Babel.

Ahora bien, no solo en las últimas versiones Babel ha resuelto casi todos sus problemas, para el castellano esos problemas nunca existieron. Si a esto sumamos el hecho de que para el español Babel ofrece muchas más opciones que Polyglossia te resultará claro el porqué en todo lo que sigue hablaré mayormente sobre Babel, si bien algunos trucos que te mostraré más adelante también funcionarán con Polyglossia.

5.4.4 Pasar opciones a Babel en \LaTeX / LyX

Como te conté más arriba, el paquete `spanish` de Babel ofrece innumerables opciones que te permitirán cambiar fácilmente algunas de las configuraciones de idioma del documento.

NOTA: Las opciones por defecto de babel para el español *son correctas desde el punto de vista ortotipográfico*⁷. Esto también se aplica a otras opciones, como mantener el sangrado de primera línea también en párrafos que siguen a un título: el quitarlo es característico de documentos escritos en inglés británico, mientras que en inglés americano se deja... Pero bueno,

7. <http://www.tex-tipografia.com/spanish2.html>

para gustos los colores, después de todo algunas de las opciones ofrecidas por el `spanish` de `babel` son realmente interesantes.

Para pasar parámetros a `Babel` desde `LyX` simplemente debes ir al menú `Documento` \gg `Configuración` \gg `Idioma` y, bajo Paquete de idiomas, seleccionar Personalizado.

En el casillero a derecha debes escribir la instrucción para `babel`, que debe ser del tipo:

```
\usepackage[spanish,opción1,opción2]{babel}
```

donde `opción1`, `opción2`... son las opciones que quieres (puedes usar solo una o más de dos). ¿Qué cuáles son esas opciones? Puedes encontrar una lista de las más importantes en este enlace:

<http://www.tex-tipografia.com/spanishopt.html>

También tienes la documentación del paquete en:

`/usr/share/texmf/doc/generic/babel-spanish/spanish.pdf`

Por ejemplo, para hacer que no se utilice la sangría de primera línea en párrafos que siguen a títulos y que los números romanos estén escritos en versalitas (para así evitar los horribles números de página en números romanos en minúscula de las páginas del índice), la instrucción sería

```
\usepackage[spanish,es-noindentfirst,
es-ucroman]{babel}
```

También resulta interesante la opción `es-noindentdot`, que evita los puntos que siguen a la numeración de secciones.

Otra forma de pasar estas opciones es, bajo `PAQUETE DE IDIOMAS`, seleccionar `NINGUNO` y escribir la instrucción directamente en el Preámbulo `LaTeX`.

La mayor parte de las veces no será necesario tocar estas cosas, como ya te conté las opciones por defecto son correctas, pero es posible utilizar instrucciones que llaman `Babel` para modificar algunas características fijas del documento, como los nombres de ciertos elementos. Volveré sobre esto más adelante, cuando sea necesario.

5.5 Importar y exportar documentos

Uno de los grandes mitos de la informática moderna es la *interoperatividad*: la idea de que se puede trabajar sin problemas en distintos sistemas y distintos programas sobre un único documento.

No entraré en discusiones del porqué esta interoperatividad es prácticamente imposible fuera del mundo web (me alejaría demasiado del contenido de este libro), pero la triste realidad es que si trabajas sobre un documento de texto con un programa y luego quieres pasar a otro *siempre* tendrás problemas.

Pero eso no quiere decir que sea imposible.

LyX ofrece integración con el programa Pandoc, el cual permite convertir entre distintos tipos de documentos. Con Pandoc y otros filtros ya presentes puedes importar y exportar hacia y desde LyX documentos odt, doc, docx, xhtml... simplemente desde el menú **Archivo** » **Importar / Exportar**. En la versión 2.4 se ha agregado una opción para exportar documentos al formato de libro electrónico ebook. Eso sí, no hay que esperar milagros: si los documentos son complejos seguramente será necesario algo de trabajo extra.

Evidentemente importar y exportar documentos \LaTeX funciona a la perfección. De hecho, si copias al portapapeles un fragmento de un documento \LaTeX tendrás la opción de hacer un pegado especial que convertirá todo a formato LyX sin esfuerzo.

5.6 Copiar información desde otros programas

Copiar y pegar información entre distintos programas siempre ha sido, es y será problemático y con LyX esto no es una excepción. Pero difícil no es sinónimo de imposible, por lo que he aquí algunos pequeños trucos para simplificar la ingrata tarea.

Pegando con **Ctrl** **V** en LyX desde, por ejemplo, LibreOffice Writer significa que además de perder todo el formato, cualquier quiebre de párrafo o línea será ignorado. Con **Ctrl** **⇧** **V** se mantienen los quiebres de párrafo. Bajo el menú **Editar** » **Pegado especial** tienes un par de opciones más, pero más allá de «pegar de \LaTeX » (siempre que copies desde un do-

cumento \LaTeX , se entiende) no hay que poner muchas esperanzas en ellas. De hecho, copiar imágenes u objetos resulta casi imposible y siempre tendrás que recrearlos.

Algo que *sí* puede copiarse (mayormente) sin problemas son los cuadros/tablas, tanto desde documentos HTML mostrados en el navegador de internet (fácil), planillas de cálculo como LibreOffice Calc o Calligra Sheets (igual de fácil) o incluso desde tablas en procesadores de texto como LibreOffice Writer (con algunas vueltas).

Imagina que en un documento HTML o en una planilla de cálculo tienes un cuadro de, digamos, 6 filas por 4 columnas. Luego de copiarlo te diriges a \LaTeX y creas un cuadro vacío de las mismas dimensiones. Finalmente, colocando el cursor en la primera celda de arriba a la izquierda haces **Ctrl**  **V**. Listo: el formato se perderá, pero el contenido irá en las celdas correctas.

Copiar un cuadro desde Writer tiene más problemas. Como en el caso anterior tienes que crear un cuadro en \LaTeX de las dimensiones correctas. Ahora te diriges al documento Writer, seleccionas el cuadro y haces **Tabla**  **Convertir**  **Tabla en texto**, seleccionando la opción de utilizar tabuladores como separador. Copias el texto resultante, vas a la primera celda del cuadro en \LaTeX y finalmente presionas **Ctrl**  **V**.

Paradójicamente, copiar desde \LaTeX hacia Writer es más simple: los entornos son automáticamente mapeados a estilos de párrafo apropiados (las secciones se convierten en «Título 2», por ejemplo) mientras que la negrita y la cursiva vienen respetadas sin problemas. ¡Incluso los cuadros/tablas se copian correctamente!

Eso sí, se generarán montones de *secciones* (en el sentido de la palabra que le da Writer⁸) que te convendrá eliminar y los vínculos a las imágenes se copian mal. Nada que no se pueda arreglar con unos pocos clics.

5.7 Control de versiones

\LaTeX tiene un sistema de control de versiones que permite registrar las diferentes etapas de la redacción de un documento y que puede resultar par-

8. Puedes profundizar en el tema aquí:

<https://elpinguinotolkiano.wordpress.com/domando-al-escriptor-2/>

ticamente útil cuando varias personas trabajan sobre el mismo archivo.

Todo comienza en `Documento` » `Seguimiento de cambios` » `Seguir cambios`, o bien con `Ctrl` `⇧` `E`. Los cambios se irán indicando con texto subrayado para indicar lo nuevo, tachado para indicar lo borrado, etcétera.

Al activar el control de versiones tendrás una nueva barra de herramientas que te permitirá aceptar o rechazar cambios, navegar entre ellos y demás.

El sistema es muy simple y está perfectamente explicado en la sección 6.16 de la guía del usuario que se encuentra en `Ayuda` » `Guía de usuario`.

Además, debo admitirlo, no es algo que utilice mucho. Podríamos decir que como escritor soy un lobo solitario... y desorganizado.

5.8 Trabajar con *ramas*

Imagina que quieres crear un documento con variantes que puedan ser seleccionadas rápidamente, tanto para ocultarlas como para mostrarlas en la versión final. El ejemplo más simple es un examen con varias preguntas y sus correspondientes respuestas: quieres que los alumnos solo vean las preguntas, pero también darle a los docentes que deben corregir el examen el documento completo con todo y respuestas.

O quizás tienes que escribir un manual para un producto con tres modelos similares pero no idénticos, por lo que seleccionando que se muestre solo lo correspondiente a un determinado modelo cuando creas un PDF obtendrás tres manuales distintos a partir de un único documento.

LyX ofrece una herramienta muy potente para lograr este objetivo: las ramas.

En un documento cualquiera puedes insertar una nueva rama desde el menú `Insertar` » `Rama` » `Insertar rama nueva`. Alguna rama tiene que ser la primera.

Si llamas a la primera rama «Preguntas», luego de aceptar te encontrarás con algo como lo que muestra la figura 5.10.

¿Notas el símbolo ✖ a la izquierda de la etiqueta? Ya volveré sobre eso.

En la caja roja puedes ahora insertar cualquier tipo de contenido: texto normal, títulos, figuras, cuadros, ecuaciones...

Si vuelves al menú `Insertar` » `Rama` » `Insertar rama nueva` verás que además de la posibilidad de crear otra rama nueva se ofrece también la rama



FIGURA 5.IO: nuestra primera rama. Lo del color verde lo discutimos luego.

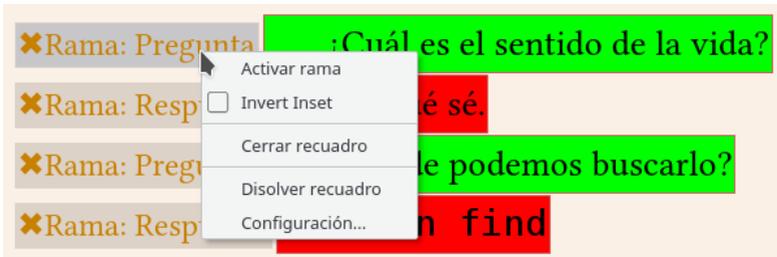


FIGURA 5.II: activar una rama

creada anteriormente.

Y así es como funciona la cosa: puedes tener varios bloques separados del documento, todos ellos agrupados bajo la misma rama en modo tal de que al ocultar esa rama todos esos bloques se oculten simultáneamente. Además, puedes tener varias ramas que puedan ser ocultadas o mostradas independientemente.

Ahora bien, por defecto las ramas están deshabilitadas por lo que si compilas el documento no se mostrarán. De hecho, si lo único que tienes en el documento son las ramas y todas ellas están deshabilitadas, *el documento no compilará*. Esto se muestra con el símbolo ✘ a la izquierda de la etiqueta de la cual te hablaba antes. Para activar o desactivar una rama es suficiente hacer clic derecho sobre una etiqueta cualquiera de las correspondientes a la rama de interés, como se ve en la figura 5.11.

Ahora la rama Preguntas mostrará un ✓ a la izquierda indicando que se ha activado y que aparecerá en el documento compilado.

En el menú apenas mostrado tienes cuatro opciones más:

INVERTIR RECUADRO ofrece la posibilidad de «invertir» un recuadro particular perteneciente a una rama para que se muestre solo cuando la rama está *oculta*.

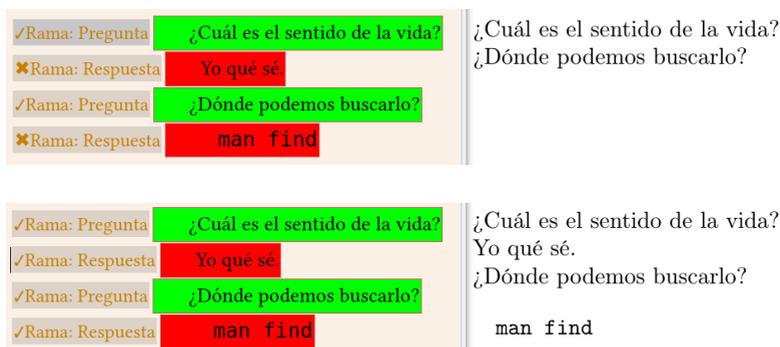


FIGURA 5.12: *dos ramas en acción con solo una activa (arriba) o ambas activas (abajo). En ambos casos se tiene la ventana de edición de LyX a la izquierda y un fragmento del PDF generado a la derecha.*

CERRAR RECUADRO simplemente oculta el contenido de ese bloque, dejando solo la etiqueta visible.

DISOLVER RECUADRO sirve para pasar el contenido de ese bloque al documento de base, quitándolo de la rama.

CONFIGURACIÓN te lleva a un simple menú que sirve para asignar el bloque a otra rama.

Otra forma de controlar las ramas, no solo para activarlas o desactivarlas sino también para unir las, cambiar nombre, asignar un color que solo se mostrará durante la edición⁹ ... es dirigirse a Documento » Configuración » Ramas.

Y así, con un par de clics, podrás pasar de la situación que se ve en la parte superior de la figura 5.12 (a la izquierda el documento LyX y a la derecha el correspondiente PDF) a lo que se ve en la parte inferior de la misma.

En definitiva, una magnífica herramienta para poder irse por las ramas con total tranquilidad.

⁹. Ahora ya sabes de dónde viene el color verde.

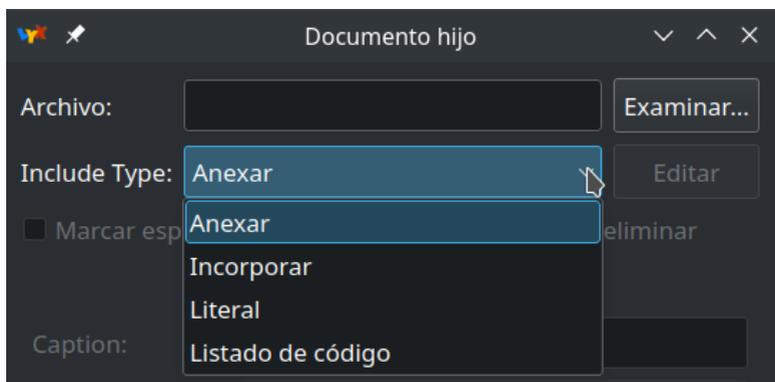


FIGURA 5.13: insertar un documento hijo

5.9 Documentos maestros

En LyX resulta increíblemente sencillo el dividir un documento en subdocumentos para combinar todo en un documento maestro. Trabajar en cada documento, realizar referencias cruzadas entre ellos, todo es un paseo. Te mostraré a continuación las opciones disponibles.

Primero que nada hay que aclarar que, a diferencia de otras aplicaciones como LibreOffice Writer, no existe una diferencia real entre un documento maestro y un documento hijo: podrías incluso tener varios niveles de documentos hijos, con un primer documento que funciona como maestro, un segundo que es el primer hijo y en este un documento *nieto*. Lo importante a notar es que la *documentclass* que prevalece será aquella del documento en el nivel más alto, por lo que si en un subdocumento utilizas alguna característica especial (algún módulo, por ejemplo) tienes que asegurarte de que la misma esté también habilitada en el documento principal.

Bueno, manos a la obra.

En el documento principal, con el cursor en el punto donde quieres insertar el documento hijo, te diriges a `Insertar >> Archivo >> Documento hijo` para obtener el menú que se ve, parcialmente, en la figura 5.13.

Este menú ofrece varias formas de incluir un documento hijo. En la primera, ANEXAR, el documento será representado en la ventana de edición por un botón centrado en su propio renglón: clic sobre ese botón y se vuel-

ve a acceder a las propiedades del documento Hijo, donde se tiene el botón EDITAR que lo abrirá para su edición. Al compilar el documento maestro el archivo así insertado irá a partir de su propia página y luego del mismo se tendrá otro quiebre de página.

Si utilizas INCORPORADO el documento irá en la misma página donde fue insertado, iniciando en un renglón nuevo.

Las dos últimas opciones, LITERAL y LISTADO DE CÓDIGO, introducen el *código* del documento hijo. Las dejo a tu curiosidad.

Algo realmente interesante si tienes todos los documentos abiertos simultáneamente, cada uno en su pestaña, es que la numeración de secciones y capítulos en la ventana de edición de LyX seguirá el orden real del documento completo, dándote una idea más clara de la estructura del documento terminado.

Hacer una referencia cruzada entre los distintos documentos y subdocumentos es increíblemente simple si tienes todos los documentos abiertos simultáneamente: *todos* los marcadores estarán disponibles desde *todos* los documentos.

Y eso es todo. No, en serio, no hay más: trabajar con documentos maestros e hijos (o nietos) en LyX es igual de fácil que trabajar en un único archivo que lo tenga todo.

Simplemente magnífico.

Hay más información en la sección 3.2 del manual que se encuentra en

[Ayuda](#) » [Características adicionales](#). También aquí:

<https://wiki.lyx.org/FAQ/Multidoc>

Parte III

Más allá de la configuración básica

Donde se libera todo el poder de L^AT_EX, X_YT_EX y OpenType desde L^YX

It's full of stars!

Capítulo 6

Algunos comandos

L^AT_EX no puede dar soporte para cada aspecto de \LaTeX en su interfaz gráfica, por lo que para pasar ciertas fronteras del formato no quedará otra opción que arremangarse y comenzar a dar instrucciones en \LaTeX puro y duro, ya sea en el preámbulo o en el cuerpo del documento.

Esto suena más difícil de lo que en realidad es por lo que ¡mantén la calma! Pero tampoco te relajes demasiado ya que hacer cosas sin comprender cómo estas funcionan es una invitación al desastre.

En este capítulo te mostraré cómo insertar código en el documento además de dar un rápido repaso a algunas de esas instrucciones que tienen la costumbre de ser necesarias muchas veces.

6.1 El preámbulo \LaTeX

Ya hablé del preámbulo \LaTeX en la introducción: es el lugar donde se pueden definir cosas que podrán ser utilizadas en todo el documento.

Accedes a él desde `[Documento]»[Configuración]»preámbulo LaTeX` y allí podrás escribir el código que te mostraré en los próximos capítulos.

Un tipo de instrucción que se repetirá muchas veces es la que permite activar un paquete \LaTeX . Por ejemplo, para poder insertar el logo de X_YL_AT_EX he debido activar el paquete `hologo`, para lo cual he escrito en el preámbulo

```
\usepackage{hologo}
```

Es decir, se utiliza una instrucción con un claro nombre (`\usepackage`) para indicar que queremos utilizar un paquete particular (`hologo`).

Es interesante notar que los *comentarios*, es decir, textos que se introducen en el código que *no* deben ejecutarse, se marcan con el signo `%`.

Así, si escribimos

```
\usepackage{hologo} % Activar el paquete
```

todo aquello que esté a la derecha del signo `%` será ignorado.

Algunos paquetes ofrecen la posibilidad de ajustar distintos parámetros para configurar su comportamiento. Esto puede hacerse directamente al llamar el paquete o mediante instrucciones especiales que estarán disponibles solo si el paquete ha sido activado, mientras que otros paquetes ofrecen, o incluso exigen, *ambas* posibilidades. En algunos casos tendrás que *parbear* un paquete con el comando `\renewcommand` e incluso te verás empujado a utilizar instrucciones lógicas para determinar cuándo (o si) un cambio debe efectuarse.

Desafortunadamente no todas las instrucciones son tan claras: por ejemplo `\makeatletter` puede fácilmente confundir a quien no esté en tema.

Simplificando un poco las cosas, el carácter `@` (*at*) es especial en L^AT_EX... pero solo a veces. Cuando se redefine un comando que utiliza `@` en su nombre, `\makeatletter` (*make @ a letter*) transforma `@` en un carácter «normal» (es un modo de decir), mientras que `\makeatother` (*make @ other*) lo vuelve «a lo que L^AT_EX dice que es».

En fin, nada que deba preocuparte mucho ya que no iré *tan* profundo en estas cosas, pero como puedes ver estamos ante un mundo complejo en el que es necesario moverse con pie firme.

6.2 La caja T_EX

Ya sea con el correspondiente botón en la barra de herramientas o simplemente presionando la combinación de teclas `Ctrl` `L` te encontrarás frente a una caja roja a la espera de código.

Siguiendo con el ejemplo de `hologo`, para hacer que aparezca el logo de X_YT_EX tendrás que escribir dentro de una de estas cajas el comando

```
\hologo{XeTeX}
```

lo cual en la ventana de edición de L^AT_EX se vería más o menos así:

```
\hologo{XeTeX}
```

NOTA: en este libro he utilizado un estilo de carácter en lugar de escribir cada vez esta instrucción. Volveré sobre esto en el capítulo 9 y en el apéndice B.

Las cajas pueden aceptar varias líneas de código y todas las instrucciones necesarias, por lo que no estás limitado a instrucciones pequeñas.

6.3 Una cuestión de tamaño

Existen instrucciones L^AT_EX para definir el tamaño del texto. Estas instrucciones pueden verse, ordenadas desde el valor más pequeño al mayor, en el cuadro 6.1 donde se presentan los valores del tamaño para las diferentes clases de documentos y para algunos valores del tamaño de texto de base¹.

Como puedes ver estos tamaños son fijos, proporcionales al tamaño de base definido para el documento, por lo que si bien tienes una amplia *variedad* que cubrirá casi todas las situaciones imaginables no te da una *libertad* absoluta. Para obtener tamaños de fuente arbitrarios existe el comando

```
\fontsize{tamaño-fuente}{altura-
línea}\selectfont
```

Si no se especifica una unidad, el comando sobreentiende que estamos hablando de «puntos». Ahora bien, volveremos a hablar de las alturas de las líneas en el apartado 7.10, pero para dar a entender que esto tiene sus vueltas... imagina que en un párrafo cualquiera escribes

Texto normal `{\fontsize{15}{30}\selectfont}` Texto grande `}`

Es decir, aplicas la instrucción solo a una pequeña parte del texto (por eso las llaves extras de apertura y de cierre). Pues bien, no importa lo que

1. Un detalle técnico: L^AT_EX usa una definición de «punto» (el «punto T_EX») diferente a la estándar (o «punto PostScript»). La diferencia es mínima, menor al 0,4%, pero existe: [https://en.wikipedia.org/wiki/Point_\(typography\)#American_points](https://en.wikipedia.org/wiki/Point_(typography)#American_points)

Tamaño	Clases estándar, <i>beamer</i>			Clases $\mathcal{A}\mathcal{M}\mathcal{S}$, <i>memoir</i>		
	[10pt]	[11pt]	[12pt]	[10pt]	[11pt]	[12pt]
<code>\tiny</code>	5	6	6	6	7	8
<code>\scriptsize</code>	7	8	8	7	8	9
<code>\footnotesize</code>	8	9	10	8	9	10
<code>\small</code>	9	10	10.95	9	10	10.95
<code>\normalsize</code>	10	10.95	12	10	10.95	12
<code>\large</code>	12	12	14.4	10.95	12	14.4
<code>\Large</code>	14.4	14.4	17.28	12	14.4	17.28
<code>\LARGE</code>	17.28	17.28	20.74	14.4	17.28	20.74
<code>\huge</code>	20.74	20.74	24.88	17.28	20.74	24.88
<code>\Huge</code>	24.88	24.88	24.88	20.74	24.88	24.88

CUADRO 6.1: valores usados por las distintas instrucciones en función del tamaño de base del texto. En esta tabla se ha ignorado la clase para generar transparencias (slides). Fuente: https://en.wikibooks.org/wiki/LaTeX/Fonts#Sizing_text

pongas en la altura de la línea —siempre que pongas algo— *el número será ignorado*. Ahora bien, si dejas en claro que la instrucción aplica a *todo* el párrafo colocando cada cosa en líneas separadas

```
{\fontsize{15}{30}\selectfont
```

¡Un texto muy grande y muy, pero muy, pero muy, pero muy, pero que muy, muy, muy largo!

```
}
```

¡allí sí que lo elegido para el interlineado será tomado en cuenta!

Para conocer más al respecto,

https://en.wikibooks.org/wiki/LaTeX/Fonts#Sizing_text

6.4 Una cuestión de forma (tipográfica)

Hablaré largo y tendido sobre tipografía en el próximo capítulo, pero en ocasiones no es necesario entrar en tanto detalle, muchas veces alcanza (y sobra) con indicar las *variantes* de las fuentes predefinidas que se desean.

Esto será de mucha utilidad en el capítulo 12.

Como ya comenté, en un documento \LaTeX se tienen tres familias de fuentes tipográficas: con gracias, también llamada romana (*roman*), sin gracias, también llamada palo seco (*sans serif*) y ancho fijo, también llamada teletipo (*teletype*).

Cada una de estas familias de fuentes tienen un «peso» que viene indicado mediante una «serie» (negrita, seminegrita, ligera) o una «forma» (normal, itálica, inclinada, versalitas).

Cada una de estas familias y cada una de estas formas puede ser llamada con una instrucción \LaTeX determinada. Así tienes tres tipos de instrucciones,

```
\XYfamily
\XYseries
\XYshape
```

donde XY son dos caracteres que representan la característica buscada.

Las familias son representadas por

- `rm` para romana: `\rmfamily`
- `sf` para sans serif: `\sffamily`
- `tt` para ancho fijo: `\ttfamily`

Las «series» son representadas por

- `bf` para negrita: `\bfseries`
- `md` para mediana: `\mdseries`
- `lf` para ligera: `\lfseries`

Por último, las «formas» son representadas por

- `it` para itálica: `\itshape`
- `sl` para inclinada (*slanted*): `\slshape`
- `sc` para versalitas (*Small Caps*): `\scshape`

Para una lista completa de estos comandos:

https://en.wikibooks.org/wiki/LaTeX/Fonts#Font_styles

6.5 ¡Números!

Sin duda se te presentará la situación de tener que elegir entre numerar con números arábigos, romanos o con letras. Cada tipo de numeración está asociada a una instrucción:

- `arabic` da los números arábigos de toda la vida: 1, 2, 3, 4...
- `Roman` da números romanos en mayúsculas: I, II, III, IV...
- `roman` da números romanos en minúsculas: i, ii, iii, iv... (¡si los vieran los romanos!)²
- `alph` da letras en minúsculas: a, b, c, d...
- `Alph` da letras en mayúscula: A, B, C, D...

Ya te mostraré en los ejemplos de más adelante cómo se utiliza todo esto.

6.6 ¡Reconfigurar!

En los próximos capítulos presentaré varios paquetes \LaTeX que serán necesarios para llevar adelante todos los trucos que llenarán las restantes páginas de este libro. Muchos de esos paquetes vienen instalados en forma predefinida en las distribuciones \LaTeX , pero no todos.

Ahora bien, no es suficiente instalar un paquete para poder utilizarlo desde LyX : tienes que avisar a este último que el paquete ha sido instalado. Para esto, luego de modificar cualquier cosa en la instalación \LaTeX debes siempre dirigirte a Herramientas » Reconfigurar, dar unos segundos para que LyX mire bien cómo están las cosas y reiniciar el programa cuando te pida amablemente de hacerlo.

². En castellano es una falta de ortografía usarlos, por lo que ya sabes: ¡no los uses!

Capítulo 7

LyX, OpenType y la tipografía

Antes de comenzar, una nota importante: en este capítulo y en los que sigan haré un uso extensivo de las «cajas T_EX» y del preámbulo L^AT_EX ya que no quedará otra alternativa que utilizar código directo. Pero no te preocupes, que no será difícil. De hecho, más adelante en este libro (capítulo 9) te mostraré cómo simplificar todo aún más con la creación de estilos.

Pues bien, al tema de este capítulo: ya tienes definida una fuente OpenType para el documento, pero quieres ir más allá de las opciones por defecto de la tipografía elegida y activar algunas *especialidades*.

Esto puede realizarse en diferentes niveles:

1. pocos y pequeños fragmentos de texto,
2. muchos y quizás no tan pequeños fragmentos de texto,
3. todo el documento,
4. las fórmulas matemáticas.

En este capítulo hablaré de estas posibilidades, pero también de algunos temas más generales: interlineado, separación silábica y microtipografía.

7.1 Introducción

En un documento L^AT_EX/X_YL^AT_EX se definen tres tipos de fuentes principales: la, para la redundancia, principal (llamada Romana en LyX), una

fuente palo seco (sans serif) y una fuente de ancho fijo.

Si bien LyX nos lo oculta detrás del menú **Documento** **Configuración** **Tipografías**, en un documento L^AT_EX la fuente principal se define en el preámbulo del documento con la instrucción `\setmainfont`, mientras que la fuente palo seco se define con la instrucción `\setsansfont` y la fuente de ancho fijo se define con `\setmonofont`.

Recuerda estas instrucciones, que más adelante tendremos que llamarlas explícitamente.

Por ahora, ve a **Documento** **Configuración** **Tipografías** y activa la opción **USAR TIPOGRAFÍAS NO TEX (VIA XETEX/LUATEX)**, seleccionando el tamaño de base y las tres fuentes principales como ya vimos en capítulos precedentes. No te preocupes ahora por la demás opciones disponibles, ya volveremos sobre ellas.

Para utilizar OpenType tienes que conocer cómo llamar a cada una de sus tablas de sustitución. En lo que sigue de este capítulo y del resto del libro tomaré como referencia la página de Dario Taraborelli:

<http://nitens.org/taraborelli/TeXOpenType>

y la página de la wikipedia:

https://en.wikipedia.org/wiki/List_of_typographic_features

pero debes estar atento al hecho de que en esas páginas falta algo de información que veremos más adelante.

A modo de ejemplo, en el cuadro 7.1 encontrás algunas de las etiquetas principales. Es importante notar que no todas las fuentes ofrecen todas estas opciones. Además, algunas de las más comunes estarán activas por defecto (como las ligaduras tipográficas) mientras que otras tienen que activarse manualmente. Y por si fuera poco, algunas opciones son incompatibles entre sí: no tiene sentido aplicar super- y subíndices simultáneamente, por ejemplo.

Ahora bien, esto no es toda la historia: algunas de estas etiquetas aceptan también parámetros. Pero como esto tiene sus vueltas, lo veremos más adelante.

7.2 OpenType para la fuente predefinida

Digamos que quieres activar una opción especial de la fuente predefinida en un punto particular del documento. Por ejemplo, y para impresionar

Característica	Etiqueta	Nombre en Fontspec
Números		
En «mayúsculas»	lnum	Numbers={Lining}
Estilo antiguo	onum	Numbers={Oldstyle}
Proporcionales	pnum	Numbers={Proportional}
Tabulares	tnum	Numbers={Monospaced}
Cero tachado	zero	Numbers={SlashedZero}
Ligaduras		
Estándar	liga	Ligatures=Common
Contextuales	clig	Ligatures=Contextual
Discrecionales	dlig	Ligatures=Rare
Históricas	hlig	Ligatures=Historical
Formas, estilos, posición y variantes		
Versalitas	smcp	Letters=SmallCaps
Superíndice	sup	VerticalPosition=Superior
Subíndice	sub	VerticalPosition=Inferior
Inferiores científicos	sinf	VerticalPosition=ScientificInferior
Ordinales	ordn	VerticalPosition=Ordinal
Sets estilísticos	ss01...ss20	StylisticSet=1...20
Alternativas estilísticas	salt	Style=Alternate
Swash	swsh	Style=Swash
Fracciones		
Numerador	numr	VerticalPosition=Numerator
Denominador	dnom	VerticalPosition=Denominator
Fracciones	frac	Fractions=0n
Fracciones alternativas	afrac	Fractions=Alternate

CUADRO 7.1: *algunas etiquetas OpenType y sus usos*

a tus lectores, podrías querer escribir un breve texto utilizando ligaduras tipográficas discrecionales — estándar actual histórico en lugar de estándar actual histórico— o quizás escribir fracciones del tipo $\frac{123}{7}$ sin utilizar el editor de ecuaciones.

Insisto en que más adelante te mostraré cómo definir estilos de carácter para así simplificar el uso de estas instrucciones, pero todavía tienes que aprender un par de cosas por lo que por ahora hay que ser rudos y escribir algo de código.

Se tiene que combinar la instrucción `\addfontfeature` con el nombre de la característica OpenType que necesitas, para lo cual puedes utilizar dos cajas \TeX rodeando el texto a modificar. Por ejemplo, las ligaduras discrecionales pueden llamarse con

```
\addfontfeature{RawFeature={+dlig}} históricas }
```

(el signo «+» *activa* la opción `dlig`) o bien con

```
\addfontfeature{Ligatures=Rare} históricas }
```

que el resultado será el mismo: históricas.

La llave que abre la primera caja \TeX y la solitaria en la segunda son importantes ya que aseguran que el código será aplicado solamente al texto que te interesa y no a todo lo que sigue.

Ciertamente es posible activar varias sustituciones simultáneamente: solo hay que separarlas con comas. Por ejemplo

```
\addfontfeature{RawFeature={+dlig, +onum}}
```

activará simultáneamente las ligaduras discrecionales y los numerales en estilo antiguo.

Es importante notar que estas instrucciones también pueden utilizarse para *deshabilitar* características habilitadas por defecto. Por ejemplo puedes deshabilitar las ligaduras contextuales, ya sea con

```
\addfontfeature{Ligatures=NoContextual}
```

o con

```
\addfontfeature{RawFeature={-clig}}
```

(el signo «-» *desactiva* la opción `clig`).

A partir de ahora, para identificar las características OpenType utilizaré las correspondientes etiquetas y no los nombres en Fontspec. Así escribo menos.

Volviendo al tema, aquí entramos en un problema que en la mayor parte de los casos no le interesará a nadie más allá de algún friki¹ tipográfico como yo, pero que para un par de fuentes particularmente completas podrá generar un poco de confusión: el uso de parámetros en las opciones OpenType.

Hasta donde sé lo que verás a continuación se aplica a poquísimas fuentes (por ejemplo, fuentes de SIL International y Junicode), pero ya sabes, para tener el panorama completo.

Según el estándar, puedes agregar un igual a una etiqueta y a continuación de la misma un número. Así, siempre según el estándar, puedes tener `+etiq=0`, `+etiq=1`, `+etiq=2` y así siguiendo. Estas etiquetas pueden activar distintas opciones. Siempre según el estándar, el valor «cero» *desactiva* la opción indicada por la etiqueta (vamos, que hace lo mismo que el menos), mientras que el 1 aplica la primera opción, etcétera.

El problema es que `fontspec` no sigue tan de cerca el estándar como lo hace, por ejemplo, LibreOffice Writer: con `fontspec` solo puedes desactivar la opción con el menos, mientras que el cero aplica la primera opción, el uno la segunda, etcétera.

Se ve que la gente que hizo esto eran programadores. En fin.

En el cuadro 7.2 te propongo un ejemplo con opciones aplicadas a la fuente Gentium Plus y al carácter Ŋ (U+014A), que en esta fuente ofrece cuatro variantes, todas aplicables con la opción `cv43`.

Insisto en que son muy pocas las fuentes que ofrecen varios parámetros para una etiqueta, y que incluso dentro de esas pocas fuentes son solo unas pocas las etiquetas que dan más de una variante, pero uno nunca sabe cuándo pueda necesitarlas.

7.3 Definir una familia de fuentes tipográficas

Lo que te conté en la sección anterior está relativamente bien solo si hay que hacerlo cada tanto: escribir todo ese código una y otra vez es ciertamente

1. No, no va en cursiva, que ya está incorporada en el diccionario: <https://dle.rae.es/friki>.

Estándar	Resultado	fontspec
cv43=0	ŀ (sin modificar)	-cv43
cv43=1	ŀ	+cv43=0
cv43=2	Ń	+cv43=1
cv43=3	Ń	+cv43=2

CUADRO 7.2: *ejemplo de uso de parámetros en opciones OpenType con la fuente Gentium Plus. Con fontspec, debes colocar la opción correspondiente dentro del «RawFeature», mientras que con Writer, que sigue el estándar... pues eso te lo cuento en otro lado, que este es un libro sobre LyX.*

molesto.

Por esto resulta conveniente definir una familia de fuentes extra que tenga la característica deseada. Además, no estarás limitado a la tipografía predefinida del documento.

Como ejemplo voy ahora a definir una nueva fuente basada en EB Garamond, pero con las ligaduras discrecionales activadas. En el menú Documento Configuración Preámbulo LaTeX es suficiente escribir lo siguiente:

```
\newfontfamily{\discrlig}[RawFeature={+hlig}]
{EB Garamond}
```

Para utilizar esta definición con código puro hay que hacer algo parecido a lo que comenté en la sección anterior (claramente, escribiendo menos), es decir, usar dos cajas TeX rodeando el texto a modificar:

```
{\discrlig{ históricas }}
```

¡No hay que olvidar la doble llave de cierre!

Y bien, esto es más sencillo que lo anterior, pero aún no resulta ideal. No te preocupes que volveré sobre esto en el capítulo 9 donde la cosa se volverá casi idílica.

7.4 OpenType para todo el documento

Imagina que quieres un documento que utilice la fuente tipográfica EB Garamond como fuente por defecto, pero con el set estilístico 6 (+ss06) activo.

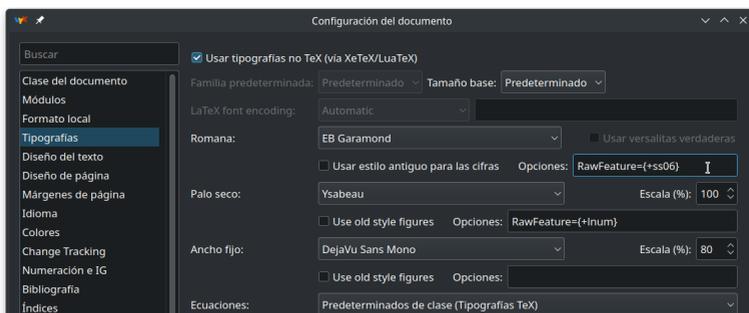


FIGURA 7.1: configuración de una característica OpenType para todo el documento

Desde la versión 2.4, LyX ofrece la posibilidad de pasar opciones OpenType a las tres fuentes principales directamente desde el menú de selección de la fuente: justo debajo de cada una de las fuentes puedes elegir no solo de utilizar numerales de estilo antiguo, también tienes una casilla llamada OPCIONES donde podrás escribir la instrucción que te sirve. Por ejemplo, escribiendo allí

```
RawFeature={+ss06}
```

para la fuente romana, ya lo habrás logrado.

Si quieres otra característica simplemente cambia `+ss06` por lo que corresponda dentro de las llaves que siguen a `RawFeature`. Si quieres varias características simultáneamente, es suficiente el escribirlas todas dentro de esas mismas llaves, separadas por comas.

7.5 Distintas fuentes para distintas formas

A partir de esta sección el menú de configuración de las fuentes nos quedará chico, por lo que pasaremos a escribir todo en el preámbulo.

Difícilmente llegues a necesitar lo que te cuento en esta sección, pero ya que estamos por aquí te lo cuento igual, así te vas haciendo una idea de la potencia del sistema. Imagina que tienes una fuente que te gusta bastante, pero que solo ofrece la versión «regular», nada de itálica o negrita. Por

ejemplo, podría ser Doulos SIL. Imagina también que te das cuenta de que Stix Two Text es bastante parecida —ambas son fuentes estilo *Times*— y que por alguna razón en lugar de utilizar directamente Stix Two como haría una persona normal quieres tener Doulos SIL para la regular y Stix Two Text para todo lo demás. No te recomiendo hacerlo, recuerda que esto es solo un ejemplo, pero sirve para ver cómo definir una «fuente virtual» que junte varias fuentes distintas para las diferentes «formas». A lo nuestro, luego de marcar el uso de fuentes no T_EX en el menú de las fuentes, dejando la casilla de la fuente romana en PREDETERMINADO, abres el preámbulo y escribes:

```
\setmainfont[BoldFont = STIX Two Text Bold,
             ItalicFont = STIX Two Text Italic,
             BoldItalicFont = STIX Two Text
                           Bold Italic,
             Mapping = tex-text]{Doulos SIL}
```

7.6 Distintas fuentes para distintos tamaños

¡Atención, que el ejemplo de esta sección podría incluso ser útil!

Supón que tienes un documento configurado con tamaño de fuente de 10 pts. Como se ve en el cuadro 6.1, el tamaño de fuente utilizado para las notas al pie de página será de 8 pts. Ahora bien, al reducir una fuente se reduce también cuán grueso es su trazado, resultando en un texto que parece más claro que el del cuerpo del documento. Para resolver este problema algunas fuentes vienen con «tamaños ópticos», versiones específicamente diseñadas para los distintos tamaños. Fuentes de este tipo son Latin Modern, TeX Gyre, etcétera. X_YT_EX soporta este tipo de fuentes, eligiendo automáticamente el tamaño apropiado en cada caso, lo cual está muy bien... El problema es que son pocas las fuentes que nos dan esta opción.

A no desesperar. Si bien fuentes con tamaños ópticos son pocas, fuentes con muchas variantes del peso ya son más.

Vayamos al ejemplo. EB Garamond, la fuente utilizada en este documento, en su anterior encarnación ofrecía una fuente pensada para 12 pts y otra pensada para 8 pts., lo cual está muy bien, pero no venía con negrita

o negrita cursiva. Octavio Pardo mejoró la fuente agregando no solo la negrita, sino también una mediana, una seminegrita y una extranegrita, todas con sus correspondientes cursivas, lo cual está magnífico, pero ya no da la versión pensada para 8 pts., la forma «regular» está basada en la vieja fuente para 12 pts. y no hay más.

Ahora bien, resulta que la versión mediana de la nueva EB Garamond tiene el mismo «color» —anchura del trazo— que la antigua versión para 8 pts., por lo que podrías preguntarte, como he hecho al construir este documento, ¿es posible decidir qué fuente utilizar para cada rango de tamaños?

¡Por supuesto! Otra vez eliges de usar fuentes no T_EX, pero dejando la romana en PREDETERMINADO y escribiendo en el preámbulo:

```
\setmainfont{EB Garamond}[
  Mapping=tex-text,
  RawFeature={+ss06},
  UprightFont = EB Garamond Regular,
  BoldFont = EB Garamond SemiBold,
  ItalicFont = EB Garamond Italic,
  BoldItalicFont = EB Garamond SemiBold Italic,
  UprightFeatures = { SizeFeatures = {
    {Size = {-10}, Font = EB Garamond Medium},
    {Size = {10-}, Font = EB Garamond}}},
  BoldFeatures = { SizeFeatures = {
    {Size = {-10}, Font = EB Garamond Bold},
    {Size = {10-}, Font = EB Garamond SemiBold}}},
  ItalicFeatures = { SizeFeatures = {
    {Size = {-10}, Font = EB Garamond Medium Italic},
    {Size = {10-}, Font = EB Garamond Italic}}},
  BoldItalicFeatures = { SizeFeatures = {
    {Size = {-10}, Font = EB Garamond Bold Italic},
    {Size = {10-}, Font = EB Garamond SemiBold
      Italic}}},
]
```

En el ejemplo de código —a esta altura, ya sabes dónde escribirlo—, luego de decir qué fuente utilizar para cada forma decido cuál utilizar para los

distintos rangos de tamaño: hasta 10 pts y a partir de 10 pts.

Si quieres hilar más fino, siempre puedes utilizar un rango intermedio, por ejemplo de 10 a 14, diciendo `Size = {10-14}`. El intervalo así definido incluye el valor inicial, pero no el final. En términos matemáticos, si x es el tamaño, $10 \leq x < 14$. Encuentras más información en el apartado III-6 del manual de `fontspec`, el cual se instala en

`/usr/share/texmf/doc/latex/fontspec/fontspec.pdf`

7.7 Distintas opciones OpenType para cada forma

Vuelvo con otro ejemplo alocado para mostrar el poder de todo esto. Voy ahora a activar algunas características OpenType para la fuente normal, otras diferentes para la cursiva, otras para la negrita... Para esto, a las instrucciones introducidas en secciones anteriores hay que agregar un poco de código.

Para dejar las cosas en claro te muestro un ejemplo completamente arbitrario y absurdo donde para la fuente *Vollkorn* activo el primer set estilístico solo para el texto normal, el set estilístico 17 solo para la cursiva, el 14 solo para la negrita y la opción «zero» (el cero tachado) solo para negrita cursiva:

```
\setmainfont{Vollkorn}[
  UprightFeatures={
    RawFeature=+ss01,
  },
  ItalicFeatures={
    RawFeature=+ss17,
  },
  BoldFeatures={
    RawFeature=+ss14,
  },
  BoldItalicFeatures={
    RawFeature=+zero,
  },
]
```

¡Atención a las comas!

Para agregar más opciones a cada tipo, simplemente se utilizan más líneas y más comas:

```
ItalicFeatures={
  RawFeature=+hlig,
  RawFeature=+swsh,
  Numbers=OldStyle,
},
```

Incluso podrías decidir que determinada forma tenga un *color* diferente... pero para eso te dejaré explorar el apartado III-6.1 de la documentación de `fontspec`.

7.8 OpenType para las matemáticas 1: **unicode-math**

Para completar la migración a fuentes OpenType, por supuesto falta cambiar también las fuentes utilizadas en las fórmulas matemáticas. Esto es simple de lograr en LyX gracias al paquete `unicode-math` y a varios proyectos tipográficos que ofrecen fuentes especialmente diseñadas para este propósito como el proyecto STIX Two, Libertinus, TeX Gyre, Asana Math, etcétera. Si lo deseas, puedes consultar la etiqueta `unicode-math` de mi blog personal, donde reviso todas las fuentes matemáticas que he encontrado:

<https://elpinguinotolkiano.wordpress.com/tag/unicodemath/>

Para utilizar esto en LyX es necesario realizar dos pequeños pasos, uno gráfico y el otro «de código». Primero hay que ir a `Documento` `Configuración` `Tipografías` `Ecuaciones` donde en el menú desplegable es necesario seleccionar la críptica opción `TIPOGRAFÍAS NO-TEX PREDETERMINADAS`. Ahora, en `Documento` `Configuración` `Preámbulo LaTeX` hay que escribir lo siguiente para utilizar las fuentes STIX Two:

```
\setmathfont{STIX Two Math}
```

Si en cambio quieres utilizar las fuentes Libertinus, pues:

```
\setmathfont{Libertinus Math}
```

O con DejaVu (a alguien podría gustarle):

```
\setmathfont{DejaVu Math TeX Gyre}
```

Si alguna de estas fuentes ofrecen sets estilísticos u otras especialidades OpenType es posible activarlas al momento de llamar la fuente. Por ejemplo para utilizar los glifos alternativos de algunas letras y obtener integrales más verticales con STIX Two Math (set estilísticos 2 y 8, respectivamente, consulta la documentación de la fuente para más información) puedes escribir

```
\setmathfont{STIX Two Math}[RawFeature={+ss02;+ss08}]
```

Otra cosa que puedes hacer es elegir distintas fuentes para diferentes *rangos*. Por ejemplo podrías insistir de utilizar Latin Modern Math para los caracteres caligráficos aún cuando la fuente matemática principal es otra. Simplemente hay que escribir

```
\setmathfont[range={\mathcal,
\mathbfcal}]{Latin Modern Math}
```

Quizás (yo nunca lo he visto), haciendo esto podrían presentarse problemas de espaciado entre los caracteres de la fórmula. Para resolver esto es suficiente agregar otra línea estableciendo un «rango vacío» para la fuente matemática de base:

```
\setmathfont[range={}]{STIX Two Math}
```

Listo, incluso nuestras ecuaciones utilizarán fuentes OpenType.

Eso sí, ¡recuérdate de instalar la fuente antes!

En Linux y con TeXLive, la documentación de `unicode-math` se instala en

```
/usr/share/texmf/doc/latex/unicode-math/unicode-math.pdf
```

Existe otra forma de seleccionar la fuente tipográfica utilizada en las fórmulas, te la presento en la siguiente sección.

7.9 OpenType para las matemáticas, 2: **mathspec**

El paquete `Unicode-math` que te presenté en la sección anterior tiene una pequeña limitación: el número de fuentes matemáticas disponibles no es ciertamente abundante.

Existe sin embargo una forma de utilizar *cualquier* fuente del sistema para escribir las fórmulas: el paquete `mathspec`.

Con este paquete los símbolos (integrales, paréntesis, un \pm) provienen de las fuentes estándar \LaTeX , pero tanto las variables como el nombre de las funciones y las letras griegas provienen de la tipografía elegida.

Utilizar `mathspec` en L^AT_EX tiene sus vueltas, pero conociendo el secreto no es un verdadero problema.

Primero que nada, es importante notar que `mathspec` llama al paquete `fontspec` que es el que permite configurar las fuentes en X_YL^AT_EX. Esto quiere decir que si activas la opción que ofrece L^AT_EX de utilizar fuentes no T_EX (opción que se dedica a configurar el paquete `fontspec`) junto con `mathspec` tendrás una larga serie de errores al tratar de compilar el documento ya que estarías llamando a `fontspec` *dos* veces. ¿Solución? *Olvidarnos de las opciones que L^AT_EX nos ofrece para configurar las fuentes tipográficas y hacer todo a mano en el preámbulo.*

En `Documento` \gg `Configuración` \gg `Tipografías` deja SIN marcar la opción `USAR TIPOGRAFÍAS NO TEX (VIA XEYTEX/LUAYTEX)`, asegurándote de que *todas* las opciones estén en `PREDETERMINADO/AUTOMÁTICO`. Luego, en `...` \gg `Formatos` \gg `Formato de salida predeterminado` elige de utilizar **PDF (X_E-T_EX)** (con **LuaT_EX** también debería funcionar).

Ahora, al Preámbulo \LaTeX . Para este ejemplo utilizaré la fuente `Cochineal`², pero puede ser cualquier otra:

```
\usepackage{mathspec}
\setmathfont{Digits, Latin, Greek}[] {Cochineal}
\setmathrm{Cochineal}

\setmainfont[Mapping=tex-text]{Cochineal}
\setsansfont[Mapping=tex-text]
```

2. <http://www.ctan.org/pkg/cochineal>

```
{Libertine Sans}
\setmonofont{DejaVu Sans Mono}
```

La primera línea dice de utilizar el paquete `mathspec`. La segunda línea configura la fuente a utilizar en las expresiones matemáticas para los distintos grupos de caracteres, pero sin pasar parámetros: simplemente dice de utilizar `Cochineal` para los números, el texto (variables, nombres de funciones, constantes, etcétera) y el alfabeto griego.

La siguiente instrucción es para hacer que las funciones como `sin`, `cos`, `tan`, `log`, etc. *también* aparezcan en la fuente requerida.

Es posible separar la instrucción para cada alfabeto y pasar algunas opciones. Por ejemplo, la instrucción

```
\setmathfont(Greek)[Uppercase=Plain,
Lowercase=Regular]{Cochineal}
```

se concentra en el griego y pasa algunas opciones, como hacer que las letras griegas en minúscula no se muestren en itálica (`Lowercase=Regular...` no es que esta opción me guste mucho, pero ya sabes, para dar un ejemplo).

Las últimas tres líneas solo definen la fuente principal, la fuente sans y la de ancho fijo, tal y como ya hemos visto que podía hacerse.

Ciertamente eres libre de elegir tus tipografías preferidas para cada opción, que el ejemplo aquí presentado es un tanto ecléctico³.

En Linux y con `TeXLive`, la documentación de `mathspec` se instala en `/usr/share/texmf/doc/xelatex/mathspec/mathspec.pdf`

Fuente de inspiración para esta sección:

<http://tex.stackexchange.com/questions/29100/using-old-style-numerals-in-xelatex-in-math-mode>

7.10 Interlineado

He aquí un dilema: tanto para el documento como un todo como para un párrafo particular, `LATEX` ofrece varias opciones de interlineado. Las tres

3. A esta altura podrías comenzar a sospechar que elijo muchos de mis ejemplos en modo tal de que no se puedan simplemente «copiar y pegar», que necesitan ser editados y personalizados antes de que resulten útiles en aplicaciones reales. Y eso.

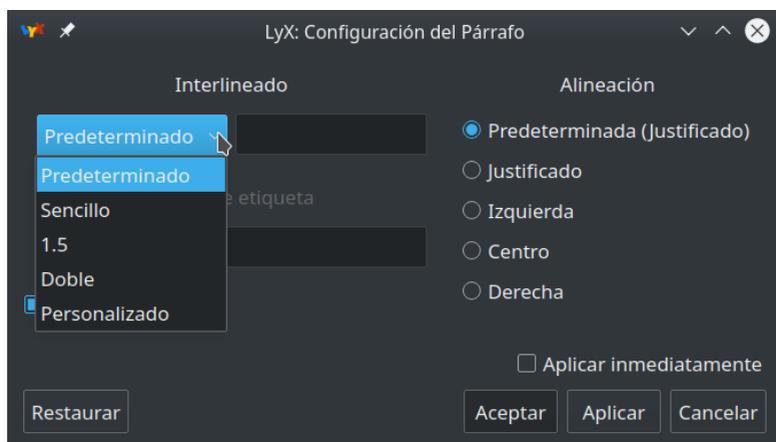


FIGURA 7.2: seleccionar el interlineado en un párrafo particular

principales («Sencillo», «Uno y medio» en la configuración del documento o «1.5» en la del párrafo —esta doble denominación resulta importante— y «Doble») parecen lo suficientemente claras y comprensibles, pero solo hasta que nos damos cuenta de que carecemos de la menor idea del significado de «Sencillo».

La cuestión no es menor y ha dado y dará muchos dolores de cabeza a quienes utilizan \LaTeX .

Te propongo un par de pruebas para comprender cómo funciona todo esto. Si sobre un párrafo haces `[clic derecho] >> Configuración del párrafo` obtendrás el menú presentado en la figura 7.2.

Bien, prueba ahora lo siguiente: crea un texto en «1.5» y otro con espacio personalizado de 1,5 (¡atención a la coma!). ¿Puedes ver lo mismo? ¿Verdad que no? ¡Personalizado de 1,5 da líneas más separadas que la opción «1.5»! Ahora vuelve a la configuración del párrafo, selecciona nuevamente PERSONALIZADO y escribe 1,25: verás cómo al presionar APLICAR la opción cambia por sí sola a 1.5. Por último selecciona nuevamente PERSONALIZADO y escribe 1,667: ahora pasa a «doble». ¿Y esto?

Las fuentes tipográficas modernas incluyen la definición de qué es una «línea sencilla»: quien diseña la fuente elige cuál es el espacio mínimo entre líneas para que estas no se choquen unas a otras. Este tamaño depende del

diseño de los caracteres (su altura, la forma en la que se apilan los acentos —sí, hay idiomas en los que una misma letra puede tener varios acentos simultáneamente—), las preferencias personales del autor... Procesadores de texto como LibreOffice Writer utilizan este parámetro interno de la fuente como base del interlineado.

Ahora bien, lo que sucede con \LaTeX es que este sistema fue creado décadas antes de Unicode, OpenType y la mar en coche. Y es por esto que \LaTeX calcula la distancia entre líneas *a su modo*.

La cosa va más o menos así: \LaTeX define la altura básica de la línea como el tamaño de la fuente. Es decir, una fuente de 10 puntos tendrá una altura de línea de 10 puntos y por esto «uno y medio» (o «1.5», según sea el menú que miremos) será de 15 mientras que la doble será de 20... PERO, ¡el interlineado sencillo no será de 10, sino de 12!

Efectivamente, dado que tener una distancia entre líneas igual al tamaño de la fuente dará un texto demasiado compacto e ilegible donde las letras más altas de una línea chocarán con las más bajas de la anterior, \LaTeX agrega automáticamente un 20% de esa altura de base, ¡pero solo al interlineado sencillo, el «uno y medio» y el doble se calculan *sin* ese aumento! Y por si fuera poco ¡el interlineado personalizado es un factor sobre el interlineado sencillo, el que ha sido aumentado!

Ahora te pediré que hagas un esfuerzo por recordar cuando en el colegio trabajabas con proporciones y porcentajes. Si sabes que una cantidad (la distancia entre dos líneas con «interlineado sencillo») es el 120% de una incógnita y quieres calcular el 150% de esa incógnita, ¿cuál es el factor que queda? Pues

$$\frac{1}{1,2} \cdot 1,5 = 1,25$$

Y si lo que quieres es tener el doble de la distancia de base, pues

$$\frac{1}{1,2} \cdot 2 \approx 1,667$$

Pues ya te digo yo que \LaTeX puede ser extraño.

Como puedes ver la confusión nace del hecho de que las opciones «uno y medio» (o «1.5») y «doble» están definidas sobre un valor diferente al valor utilizado para definir «personalizado».

Y antes de que lo preguntes, pues sí, esto también afecta a X_{\LaTeX} y a \LuaTeX .

Más información información sobre estos temas en:

<http://wiki.lyx.org/FAQ/Spacing>
<http://tex.stackexchange.com/questions/13742/what-does-double-spacing-mean>
<https://en.wikipedia.org/wiki/Leading>

7.11 Separación silábica

Cuando en un documento haces que el texto esté «justificado» (hacer que ambos márgenes, izquierdo y derecho, formen una línea) te enfrentas a un problema: si el texto está en columnas o la fuente tipográfica usada es grande en modo tal de tener pocas palabras por línea, el software que uses se verá obligado a variar la distancia entre las palabras para ajustarse a los márgenes, lo que dará espacios en blanco de tamaño impredecible que harán incómoda la lectura. Para evitar esto se recurre a la separación silábica de palabras a final de línea en modo tal de reducir esos espacios... pagando el precio de tener muchas palabras *cortadas*.

En general \LaTeX hace un gran trabajo equilibrando los espacios con la separación silábica, pero solo cuando el texto es «lo suficientemente ancho». ¿Es posible cambiar manualmente el comportamiento de \LaTeX ? Sí, aunque lo que hay que hacer es más cercano a la magia que a un método de trabajo.

\LaTeX usa un algoritmo complejo para decidir cómo distribuir el texto en las líneas del párrafo. Simplificando un poco el proceso, \LaTeX calcula un parámetro que mide «qué tan mal» se ve el texto y trata de minimizarlo en dos pasadas. En la primera pasada calcula qué tan mal se ve el texto sin la separación silábica y si el resultado es aceptable, pues todo listo. Si el resultado de la primera pasada no es aceptable (y suele no serlo), pasa de nuevo permitiendo esta vez la separación silábica al final de la línea.

Lo que dice si un resultado es aceptable o no es un parámetro llamado *tolerancia*: si lo mal-que-se-ve está dentro de lo tolerado, pues pasa.

Y dado que el algoritmo hace dos pasadas tenemos dos parámetros de tolerancia, uno para la primera pasada y otro para la segunda.

¿Y como controlar estas tolerancias? En el preámbulo \LaTeX se escribe

```
\pretolerance=<valor>  
\tolerance=<valor>
```

Donde <valor> es un número entero entre 0 y 10000.

Y es en estos valores donde viene la magia: el valor correcto para `pre-tolerance` y `tolerance` dependerá del tamaño de la página, de la fuente tipográfica, de los márgenes, del número de columnas... en fin, que encontrarlos es cosa de prueba y error. Y es que cuanto más angosto el texto es, menos estables estos parámetros resultan, lo cual significa que cuando más los necesitemos más difícil será el encontrarlos.

Y sí, la vida es cruel. Mejor deja que \LaTeX haga su trabajo aquí.

7.12 Microtipografía

Antes que nada, una advertencia: el tema de esta sección es puramente sobre *estética*. De lo que te voy a hablar es de cómo hacer que el texto creado en \LaTeX , ya sea con \LaTeX o con \XeTeX , se vea más bello. No tiene utilidad por sí mismo y de hecho la mayor parte de los lectores no notará lo que haremos... ¡lo cual significará que lo hemos hecho bien!

La idea es justamente modificar en forma sutil el texto para que «se vea mejor», pero que quien lo vea no sea capaz, en un primer análisis, de decir qué se modificó. ¿Y esto cómo se logra?

Bien, primero que nada debemos hacer una declaración sumamente fuerte: es terriblemente simple engañar a nuestro cerebro. Para muestra, es más que suficiente revisar el artículo sobre ilusiones ópticas en la wikipedia:

https://en.wikipedia.org/wiki/Optical_illusion

Y es justamente un tipo de ilusión óptica lo que enfrentamos al ver un texto justificado normal: cada tanto tienes la sensación de que el margen derecho es ondulado, con entradas. Justamente donde se tienen los signos de puntuación y los guiones parece que la línea retrocediera, aún cuando esto no sea así (puedes comprobarlo con una regla).

Lo que sucede aquí es que los puntos, comas y guiones tienen *menos peso* que los caracteres que los anteceden, por lo que nuestro cerebro interpreta la imagen como si hubiera huecos allí.

Una forma de *resolver* este problema es hacer que esos caracteres más etéreos atraviesen ligeramente el margen derecho, así el borde de las líneas parecerá más lleno y tu cerebro interpretará que el margen derecho es más uniforme... aún cuando no lo sea. El cambio es realmente mínimo, pero

mejora notablemente la apariencia del texto.

Esta técnica de *microtipografía* se llama *protrusión* y puedes ver cómo funciona en la figura 7.3: la línea gris muestra claramente cómo el guion y la coma se pasan del margen.

Otra técnica de *microtipografía* es deformar los caracteres para que ocupen más o menos espacio horizontal y así evitar el que los espacios entre palabras sean diferentes entre las distintas líneas. También se puede modificar la distancia entre caracteres... en fin, que las técnicas son varias. Pero a diferencia de la protrusión que siempre queda bien, estas otras técnicas son importantes

mayormente cuando se tienen columnas angostas de texto, es decir, cuando la macrotipografía comienza a mostrar sus limitaciones.

No intentes siquiera pensar en tratar de aplicar alguna de estas técnicas en procesadores de texto normales como Writer o Word: simplemente no es posible. Scribus permite trabajar limitadamente con algunas de estas técnicas, pero nunca he utilizado este programa por lo que no puedo comentar al respecto.

Lo único que queda en el mundo FLOSS es \LaTeX y sus derivados con el paquete `microtype`, cuya documentación se instala, en Linux, en `/usr/share/texmf/doc/latex/microtype/microtype.pdf`

El manual tiene más de 250 páginas, pero no te preocupes que en un uso normal las opciones por defecto son simplemente perfectas y no tienes que preocuparte de cómo funciona.

Para habilitar las opciones por defecto de microtipografía es suficiente ir al menú de configuración de fuentes y marcar la correspondiente casilla. Listo, no hay más que hacer. Esto funciona ya sea con \LaTeX normal, $X\TeX$ o \LuaTeX , aunque en estos dos últimos solo en forma limitada: por

s espacio
entre di-
... en fin,
« queda

FIGURA 7.3: *ejemplo de protrusión*

ejemplo, al momento de escribir este libro $\text{X}_{\text{T}}\text{E}_{\text{X}}$ solo ofrece soporte para protrusión.

Si sientes curiosidad y quieres profundizar en estos temas, he aquí algunos enlaces

<http://hstuart.dk/2007/11/27/microtypography/>

<http://wiki.lyx.org/Tips/Typography>

<https://en.wikipedia.org/wiki/Microtypography>

7.13 Sub- y superíndices

Me refiero a sub- y superíndices en el texto, que en modo matemático es fácil: el guion bajo para subíndices, el acento circunflejo para los superíndices y a otra cosa. Ahora bien, tener sub- y superíndices en modo texto también es necesario, ¿cómo lograrlos?

Antes de seguir hay que distinguir entre sub- y superíndices *reales* y *simulados*. Los sub- y superíndices reales son caracteres especiales diseñados explícitamente para actuar como tales mientras que los simulados simplemente provienen de hacer la fuente más pequeña y subirla (o bajarla) un poco, lo cual nunca se ve igual. Por ejemplo,

$$2a^{22}$$

donde el primer 2 después de la a es un superíndice real mientras que el segundo ha sido simulado (el 2 delante de la a sirve de referencia).

Como puedes ver los índices reales se ven mejor que los simulados, con menos contraste y menos detalles, lo que facilita su legibilidad. Y es que al escalar la fuente para simular el índice también se reduce el grosor de los trazos, haciendo que los índices se vean más claros que la letra de base.

Ya te conté cómo aplicar especialidades OpenType sobre una fuente, por lo que activar los índices verdaderos con las etiquetas `+sup`, `+sinf` o `+subs` no debería ser un problema... salvo por el hecho de que no todas las fuentes tipográficas ofrecen estas opciones y las que sí, no nos dan todos los caracteres que podríamos necesitar.

En fin, que generalmente convendrá utilizar índices simulados. Para aliviar este problema resulta conveniente elegir una fuente tipográfica que ofrezca tamaños ópticos, como por ejemplo Latin Modern o Antykwa Pol-

tawskiego, ya que los índices serán generados a partir de la fuente correcta para ese tamaño y se verán bien.

L^AT_EX permite trabajar con índices simulados gracias a las instrucciones `\textsuperscript{}` y `\textsubscript{}`. Para ambas LyX ofrece soporte completo desde la interfaz gráfica.

Puedes insertar estos índices en modo texto en LyX desde los menús Insertar >> Formato >> Superíndice/Subíndice.

Por defecto LyX no ofrece atajos de teclado para estas opciones, pero ya vimos cómo solucionar este problema (sección 4.2), por lo que lo único que necesitas son los códigos correspondientes, que para un superíndice es:

```
script-insert superscript
```

mientras que para el subíndice es:

```
script-insert subscript
```

NOTA: Si la fuente usada ofrece soporte para los sub- y superíndices reales existe un paquete (siempre existe un paquete) que al encontrarse con las instrucciones `\textsuperscript` y `\textsubscript` las reemplaza automáticamente por la opción OpenType correspondiente.

Este paquete se llama `realscripts`, pero no hablaré de él aquí ya que su documentación es muy simple (seis páginas). Ten en cuenta que al utilizar este paquete, si se te escapa un carácter para el cual la fuente no tiene un índice (muchas fuentes tienen solo los números y casi ninguna ofrece acentos o la ñ) tendrás resultados extraños.

Ahora bien, ¿cómo hacer para modificar cuánto se reduce la fuente del índice simulado y cuánto es este subido o bajado? Pues bien, para lograrlo hay que recurrir a código L^AT_EX puro y duro a través de las famosas cajas T_EX.

La instrucción a utilizar es `\raisebox` y funciona del siguiente modo

```
\raisebox{<altura>}{<texto>}
```

donde `<altura>` indica el desplazamiento vertical y `<texto>` aquello a convertir en índice.

El valor de `<altura>` puede ser positivo (desplaza hacia arriba) o negativo (desplaza hacia abajo) y puede ser expresado en cualquier unidad de medida. Por ejemplo

```
\raisebox{6pt}{superíndice}
\raisebox{-6pt}{subíndice}
```

nos dará (la primera) un superíndice a 6 puntos por encima de la línea de base y un subíndice (la segunda) a 6 puntos por debajo de la línea de base.

Para el tipógrafo detallista, también puedes utilizar como unidad el `ex`. Muy básicamente, un `ex` es un valor igual al de la altura de una letra minúscula sin trazos ascendentes o descendentes: por ejemplo, la altura de una `x`. Podríamos por lo tanto utilizar

```
\raisebox{1ex}{superíndice}
```

para hacer que el superíndice suba a la altura del «techo» de las letras bajas.

Ahora bien, la instrucción `\raisebox` no cambia por si misma el tamaño del texto por lo que tendrás que utilizar una instrucción extra. Por ejemplo

```
\raisebox{1ex}{\scriptsize{superíndice}}
```

dará un superíndice de tamaño *scriptsize* (ya te conté sobre los tamaños de fuente para \LaTeX en la sección 6.3) a una altura de `1ex` sobre la línea de base. En principio, para utilizar este código se necesitan dos cajas \TeX , una antes y la otra después del texto a poner como índice, siguiendo el modelo siguiente

```
base {\raisebox{1ex}{\scriptsize{superíndice}}
obtendrás el siguiente resultado
basesuperíndice
```

Sé lo que estás pensando: escribir todo esto todas las veces que necesitas un sub- o superíndice... ¿no existe una forma de abreviar? Ciertamente, pero tendrás que esperar hasta el capítulo 9.

Capítulo 8

Notas al pie y notas al final

Aquí te mostraré cómo modificar la presentación y el funcionamiento de las notas al pie de página en un documento LyX y de cómo utilizar notas al final.

Para todo esto será necesario el paquete `texlive-yafont`. Y código en el preámbulo, por supuesto.

Varios de los comandos que indicaré en este capítulo aceptan parámetros de configuración, pero no siempre entraré en el detalle de lo que pueden dar. En la mayor parte de los casos hacer lo comentado aquí será más que suficiente por lo que queda a tu curiosidad el consultar la documentación de estos paquetes para comprender cómo ir más allá.

8.1 Formato de las notas al pie y sus «marcas»

Por defecto, en $\text{L}\text{A}\text{T}\text{E}\text{X}$ y derivados no solo el número que identifica la nota al pie en el texto está como superíndice, sino también el que numera la nota en sí. Una forma de cambiar esto último es agregar lo siguiente en el preámbulo $\text{L}\text{A}\text{T}\text{E}\text{X}$:

```
\makeatletter
\renewcommand\@makefnmark[1]{%
  \noindent\makebox[0pt][r]{
    \@thefnmark.\space}#1}
\makeatother
```

Aquí se redefine la marca de las notas al pie, haciendo que no tengan sangría (`noindent`), que el número esté a la derecha (la `r`) de una caja (`makebox`) alineada a 0 pt del margen (y por lo tanto el número entrará en el margen izquierdo) mientras que al número en sí (`@thefnmark`) no se le aplica formato alguno, por lo que aparecerá normal y no como superíndice. La instrucción `\@thefnmark` es la que indica la marca de la nota al pie, por lo que si quieres darle un formato especial la rodeas de llaves y de las instrucciones necesarias.

Pero esto no es lo único que puedes hacer, también es posible cambiar la alineación de las notas al pie y la fuente utilizada por las mismas. Mira el siguiente código:

```
\renewcommand{\@makefnmark}[1]{%
  \setlength{\parindent}{0pt}%
  \begin{list}{}{\setlength{\labelwidth}{6mm}%
    \setlength{\leftmargin}{\labelwidth}%
    \setlength{\labelsep}{3pt}%
    \setlength{\itemsep}{0pt}%
    \setlength{\parsep}{0pt}%
    \setlength{\topsep}{0pt}%
    % \setlength{\rightmargin}{0.2\textwidth}%
    \footnotesize}%
    \item[\@thefnmark\hfil]#1%
  \end{list}%
}
```

Te dejo explorarlo. Si quieres una pista, la línea donde dice `\footnotesize` es donde se configura la fuente de la nota al pie, prueba a escribir allí algo como `\sffamily\footnotesize` y luego me cuentas. A ver, no es que lo recomiende, es más, *no* lo recomiendo, pero puedes probar. Venga, vale, una segunda pista: la línea comentada te permite cambiar el margen derecho del párrafo de las notas. Ahora sí, te queda como ejercicio el entender el resto.

8.2 Notas al pie en columnas con texto principal sin columnas

Imagina que tienes muchas notas al pie, pero todas ellas sumamente breves del tipo «ver página tal», «referencia cual, en página #», etcétera. En esta situación podría resultar interesante el tener las notas al pie de página en dos columnas aún cuando el cuerpo del texto se presente sin columnas.

Para esto escribe en el preámbulo \LaTeX lo siguiente

```
\usepackage{dblfnote}
```

y todo listo. La única cosa a tener presente es que si la página está medio vacía las notas al pie se acomodarán a sus anchas llenando la primera columna en lugar de distribuirse uniformemente.

8.3 Notas al pie sin quiebres de párrafo

Si agrupar las notas al pie en dos columnas no es suficiente y quieres obtener notas al pie en la misma línea sin quiebres de párrafo, el paquete `footmisc` es tu amigo. En el preámbulo \LaTeX es suficiente escribir:

```
\usepackage[para]{footmisc}
```

A alguien podría gustarle.

8.4 Reiniciar la numeración de las notas al pie en cada página

Es suficiente agregar, en el preámbulo \LaTeX , lo siguiente

```
\usepackage{perpage}  
\MakePerPage{footnote}
```

La primera línea declara el uso de la instrucción `perpage`, una instrucción que reinicia la numeración de distintos contadores en cada página, mientras que la segunda indica cuál contador queremos reiniciar (el de las notas al pie).

8.5 «Numerar» las notas con símbolos

Imagina que para los números de las notas al pie quieres la siguiente sustitución:

- 1 → * (comando `\ast`)
- 2 → † (comando `\dagger`)
- 3 → ‡ (comando `\ddagger`)
- 4 → § (comando `\S`)
- 5 → **

etcétera. Es importante tener en cuenta que numerar con símbolos solo tiene sentido si reinicias la numeración de las notas en cada página y si se tienes pocas notas en una misma página, de lo contrario el documento se volvería caótico.

Esto tendría que poder hacerse con el paquete `footmisc`, pero aquí el paquete `Spanish` de `Babel` se mete en el camino ya que la tradición ortotipográfica española pide que la primera nota tenga un asterisco, la segunda dos, la tercera tres... y así siguiendo.

Ningún problema, que ya te dije que el `Spanish` de `Babel` es muy flexible. En el preámbulo escribe lo siguiente:

```
\usepackage[perpage,symbol*]{footmisc}
\usepackage[spanish,es-nolayout]{babel}
```

Con esto obtendrás la distribución de símbolos por defecto de `TeX`, que es como la lista de más arriba, con la diferencia de mostrar un ¶ en el quinto nivel y un || en el sexto. Si tienes más notas, a partir de la séptima solo se duplican los símbolos anteriores en el mismo orden.

`footmisc` ofrece otras combinaciones de caracteres, que puedes elegir agregando una línea más al código anterior

```
\setfnsymbol{<opción>}
```

donde `<opción>` puede ser `bringhurst`, `chicago` o `wiley`. Eso sí, he tenido problemas con esas opciones: si quieres poner más de seis notas en la misma página, te darán error.

Puedes encontrar más información en las secciones 1.6 y 1.7 del manual de `footmisc`, el cual se instala en

`/usr/share/texmf/doc/latex/footmisc/footmisc.pdf`

A continuación te presento una alternativa a lo anterior que, si bien resulta un poco más complicada, es más flexible ya que te permite elegir la secuencia de símbolos que quieras sin tener que llamar Babel y en forma completamente independiente del idioma.

Siguiendo con el ejemplo del principio de esta sección, en el preámbulo \LaTeX escribe lo siguiente:

```
\def\mysymbols#1{\@mysymbols{\@nameuse{c#1}}}  
\def\@mysymbols#1{\ifcase#1\or $\ast$\or  
$\dagger$\or $\ddagger$\or $\S$\or  
$\ast\ast$\or $\dagger\dagger$\or  
$\ddagger\ddagger$\or $\S\S$\or  
\else\@ctrerr\fi}  
\def\thefootnote{\mysymbols{footnote}}  
\usepackage{perpage}  
\MakePerPage{footnote}
```

La primera línea define una nueva instrucción llamada `\mysymbols`, la segunda configura esa instrucción estableciendo las sustituciones comentadas más arriba, finalmente se redefine la instrucción `\thefootnote` para hacer que a la numeración venga aplicada la sustitución anterior. Las últimas dos líneas *deben estar al final* y aplican el paquete `perpage` a las notas al pie.

Es importante notar que si tratas de utilizar en una página más notas del número de símbolos definidos, \LaTeX dará error, por lo que mejor pecar de exagerado: si quieres definir más símbolos, simplemente agregas otro `\or $\$$ símbolo $\$$` en la lista de la segunda línea de código.

Fuente de inspiración para esta sección:

https://groups.google.com/d/msg/comp.text.tex/QZrTF7tzo5k/TvT_RsCyBfsj

8.6 Volver desde una nota al pie

Con `hyperref` obtendrás archivos PDF que te permiten ir desde el ancla de la nota al pie a la nota misma, ¿pero cómo hacemos para *volver*?

Pues con un pequeño, viejo, poco conocido, pero indiscutiblemente útil paquete: `footnotebackref`.

Luego de instalar el paquete (no suele venir por defecto), reconfigurar \LaTeX y configurar todas opciones que quieras para `hyperref`, en el preámbulo debes escribir

```
\usepackage{footnotebackref}
```

Esto hará que las marcas de las notas al pie se vuelvan enlaces que te permitirán regresar al ancla de la nota, *pero solo si no las has redefinido con lo que te comenté en la primera sección de este capítulo.*

En efecto, si cambias manualmente el formato de las marcas de las notas al pie terminarás sobrescribiendo, y por lo tanto anulando, lo que te da `footnotebackref`. Si quieres que las marcas se muestren en texto normal y que al mismo tiempo sean enlaces para volver, tendrás que utilizar este código:

```
\makeatletter
\renewcommand\@makefnmark[1]{%
  \noindent\makebox[0pt][r]
    {\hyperref[\BackrefFootnoteTag]{
  \@thefnmark.\space}}#1}
\makeatother
\usepackage{footnotebackref}
```

Si utilizas `hyperref` para definir colorines en los enlaces y cosas así, **este código tiene que estar al principio del preámbulo!**

Como puedes ver estamos aquí rodeando a `\@thefnmark` con código que indica que tienen que convertirse en enlaces. Para poner aún más formato... ya sabes qué tienes que hacer.

8.7 Notas del editor

Lo habrás visto más de una vez, especialmente en libros traducidos: a las normales notas al pie del autor se suman las notas al pie del editor, con su propia numeración y más de una vez claramente separadas de las otras.

Para poder definir un grupo independiente de notas al pie necesitas utilizar los paquetes `manyfoot` y `perpage`. Ahora, en el preámbulo del documento escribe lo siguiente:

```
\usepackage{manyfoot}
\newcommand{\Afootnoterule}{}
\SelectFootnoteRule{A}[\noindent\footnotesize
    Notas del editor:\vspace{2mm}]
\DeclareNewFootnote{A}[alph]
\usepackage{perpage}
\MakePerPage{footnoteA}
```

La primera línea declara el primer paquete. La segunda y la tercera líneas declaran el separador que verás entre las notas al pie normales y las nuevas. La cuarta línea crea un nuevo tipo de nota al pie que estará numerada con letras y que estará asociada a la instrucción \LaTeX `\footnoteA`. La quinta línea declara el segundo paquete para poder, en la sexta línea, decir que estas nuevas notas al pie estén numeradas por página.

Ahora, como anticipo de lo que te contaré en mayor detalle el próximo capítulo (y para no liarnos con infinitas cajas \TeX), debes ir a [Documento](#) [Configuración](#) [Formato local](#) para crear un nuevo «recuadro personalizado» donde escribir estas nuevas notas al pie:

```
Format 95

Counter ednote
    PrettyFormat "Ednote ##"
End

InsetLayout Flex:Ednote
    LyXType      custom
    LatexName    footnoteA
    LatexType    command
    Counter      ednote
    Decoration   classic
    Font
```

```

        Size Small
    EndFont
    MultiPar    true
    LabelString ednote
    Requires    ednote
End

```

validas, aceptas y ya está: tendrás ahora un nuevo «recuadro personalizado» al cual acceder desde [Insertar](#) » [Recuadro personalizado](#) » [Ednote](#) (o el nombre que elegiste) con el que puedes insertar estas nuevas notas al pie. Te dejo probar el resultado.

8.8 Notas al final

No solo de notas al pie vive el escritor, en ciertos documentos es necesario el utilizar notas al final.

LyX ofrece soporte desde la interfaz gráfica del paquete endnotes. Como *otra* introducción de un tema que será discutido en el capítulo 9, debes ahora dirigirte a [Documento](#) » [Configuración](#) » [Módulos](#) para habilitar el módulo NOTAS FINALES. A partir de 2.4 tienes también el módulo NOTAS FINALES (EXTENDIDO) que utiliza el paquete enotez, que es mucho más potente que endnotes.

También es posible forzar a todas las notas al pie a convertirse en notas al final con los paquetes NOTAS AL PIE COMO NOTAS AL FINAL y NOTAS AL PIE COMO NOTAS AL FINAL (EXTENDIDO).

Para insertar una nota final en el documento simplemente te diriges a [Insertar](#) » [Recuadro personalizado](#) » [Notas finales](#) y escribes el texto de la nota dentro de la caja que se te presenta.

Finalmente, para *mostrar* las notas en el documento, desde 2.4 simplemente debes ir a [Insertar](#) » [Listas/contenidos/referencias](#) » [Notas al final](#). De ser necesario, puedes renombrar la sección de las notas al final con

```
\renewcommand{\notesname}{Notas al final}
```

Algo interesante de este sistema es que puedes insertar tantos índices como quieras, por lo que podrías tener las notas al final acumuladas por

capítulos, por ejemplo, y no todas al final. Eso sí, si quieres separar las notas al final de esta forma resultará conveniente reiniciar la numeración de las mismas cada vez, agregando la instrucción

```
\setcounter{endnote}{0}
```

en una caja \TeX al comienzo del nuevo capítulo.

8.9 Una *yapa*: notas al margen

\LaTeX ofrece soporte para insertar notas al margen: simplemente te diriges a Insertar Nota al margen y ya tienes una cajita llamada «margen» esperando a ser completada.

¡Hola!

No hay mucho que pueda hacerse para modificar el formato de estas notas al margen desde la interfaz gráfica del programa, por lo que tendrás que recurrir a las cajas \TeX si quieres personalizarlas.

Por ejemplo las notas al margen se colocan en el margen externo del documento, pero esto puede cambiarse rápidamente insertando una caja \TeX con el código

```
\reversemarginpar{}
```

mientras que para volver a la normalidad solo tienes que escribir, en otra caja \TeX , lo siguiente:

```
\normalmarginpar{}
```

\LaTeX utiliza la implementación por defecto de \LaTeX de las notas al margen, la cual tiene algunas limitaciones (no puede usarse en flotantes, notas al pie, etcétera). Para utilizar el paquete `marginnote`, que es mucho más completo, puedes escribir en el preámbulo \LaTeX lo siguiente:

```
\usepackage{marginnote}
\let\marginpar\marginnote
```

La documentación de `marginnote` se instala en:

```
/usr/share/texmf/doc/latex/marginnote/marginnote.pdf
```

Es muy breve, por lo que estoy seguro de que no tendrás problemas para utilizarla.

A este punto me veo obligado a confesar algo: no soy particularmente entusiasta de las notas al margen, hasta podría decirse que las detesto, por lo que no seguiré hablando de ellas y te dejaré consultar la documentación de `LyX` sobre el tema. Todo lo que se puede saber sobre las notas al margen en `LyX` se encuentra magníficamente explicado en [Ayuda](#) » [Objetos insertados](#) » [sección 4.3](#).

Capítulo 9

Módulos, formato local y macros matemáticas

Otra de las notables características de $\text{L}\text{\AA}\text{X}$ es que puede *extenderse*. En este capítulo te mostraré cómo agregar funciones, el modo correcto de definir formatos no presentes en la *documentclass* utilizada y cómo preparar algunos trucos para escribir fórmulas matemáticas con mayor facilidad.

9.1 Módulos

En el capítulo 8 mencioné, al pasar y como quien no quiere la cosa, los módulos: bajo `Documento` \gg `Configuración` \gg `Módulos` (figura 9.1) tienes una larga lista de nuevas funciones que solo esperan a ser activadas, todas ellas agrupadas por categorías.

Estos módulos simplifican el uso de determinados paquetes \LaTeX . Hablar de todos ellos es simplemente imposible, pero en próximos capítulos comentaré algunos, por lo que por el momento te dejo jugar con las opciones. Ahora bien, ¿es posible agregar *otros* módulos no ofrecidos en la instalación por defecto?

¡Ciertamente! (Si bien generalmente no será necesario...)

En esta página de la wiki de $\text{L}\text{\AA}\text{X}$:

<http://wiki.lyx.org/Layouts/Modules>

es posible encontrar varios módulos listos para ser utilizados y más adelante veremos otros más. Tienes módulos que agregan nueva funcionalidad

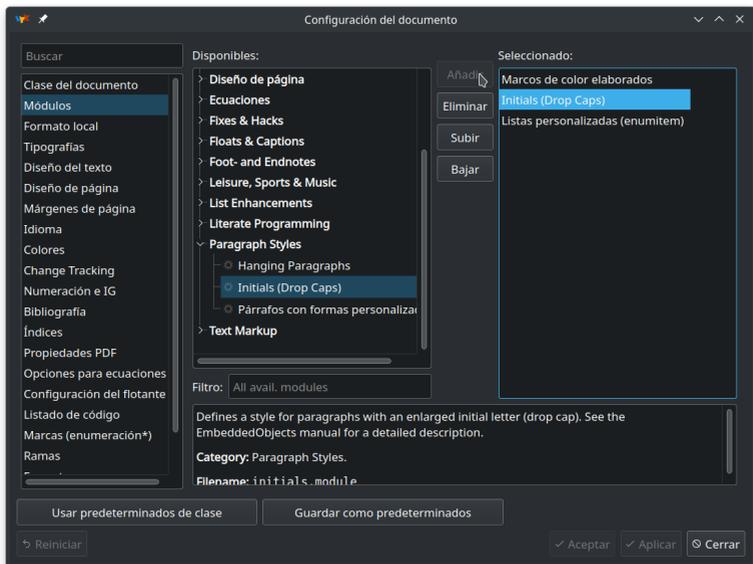


FIGURA 9.1: *activando módulos*

(comentarios, índices de nombres, formato), integración con el programa Sage, modificaciones del formato de distintas *documentclass*, etcétera.

La forma de instalar estos módulos es, luego de descargar los archivos *.module, copiarlos en la carpeta *layouts* del perfil de usuario de LyX que en Linux se encuentra en

`~/.lyx/layouts/`

¡Recuerda siempre el menú reconfigurar!

9.2 Crear nuevas instrucciones

Casi sin darnos cuenta, ya vimos cómo crear instrucciones al definir nuevas familias de fuentes en la sección 7.3. Pero por supuesto eso era solo un aperitivo: verás ahora cómo hacer mucho más, creando instrucciones que te permitan, por ejemplo, aplicar formato, ya sea de párrafo o de carácter.

La forma más simple de lograr nuevas instrucciones en \LaTeX /LyX es a

través del comando `\newcommand{ } [] { }`. Esta instrucción se utiliza en el preámbulo \LaTeX de la siguiente forma:

```
\newcommand{\micomando}[<número de
  parámetros>]{<texto e instrucciones LaTeX>}
```

Retomando el ejemplo al final del capítulo 7, imagina que necesitas crear una nueva instrucción para insertar un superíndice en modo texto utilizando el comando `raisebox`. Aquí `\micomando` es el nombre que le darás a la instrucción, por ejemplo `\subir`, `<número de parámetros>` indica cuántos parámetros necesitará el nuevo comando, en este caso 1, mientras que en `<texto e instrucciones LaTeX>` hay que escribir lo que sirva para crear el formato buscado, llamando a cada parámetro con `#<número>` (`#1` para el primer parámetro, `#2` para el segundo, etcétera).

El código a insertar en el preámbulo \LaTeX será entonces el siguiente

```
\newcommand{\subir}[1]{\raisebox{1ex}{
  \footnotesize{#1}}}
```

En principio, para utilizar esta instrucción dentro de \LaTeX en modalidad «fuerza bruta / vamos a escribir código», abres una primera caja \TeX con **Ctrl** **L**, escribes `\subir{`, sales de la caja para escribir el texto del superíndice y finalmente creas otra caja \TeX donde escribes un `}` de cierre. De esta forma, teniendo lo siguiente en la ventana de edición de \LaTeX

base `\subir{` superíndice `}`
 obtendrás
 base^{superíndice}

Es posible utilizar más parámetros, o incluso ninguno. En este último caso la instrucción tendría la forma

```
\newcommand{\micomando}{<texto e
  instrucciones LaTeX>}
```

¡Todo listo! Bueno, no todo, que como es fácil ver esto sigue siendo poco práctico. ¡Pero ya hemos llegado! Finalmente te mostraré cómo hacer esto y mucho más al mejor estilo \LaTeX .

9.3 Crear estilos de carácter en LyX

En LyX puedes aplicar rápidamente algunos estilos de carácter predefinidos como énfasis (con **Ctrl** **E**), por defecto da *cursiva*), **negrita** (**Ctrl** **B**) o **VERSALITAS** (**Alt** **C**, luego **C**), pero no mucho más. Activando el módulo «marcación lógica» tendrás algunos más que aparecerán listados en el menú al cual accedes desde el **clic derecho** » **Estilo de texto personalizado** o bien en **Editar** » **Estilo de texto personalizado**. Ahora bien, ¿si quieres crear otro? Podrías necesitar un modo más simple de activar una característica OpenType determinada —por ejemplo, ligaduras discrecionales— en lugar de usar tanta caja T_EX o escribir menos a la hora de utilizar el paquete `hologo` del que te hablé antes.

NOTA: ¡Recuerda que para este segundo ejemplo hay que escribir `\usepackage{hologo}` en el preámbulo!

En **Documento** » **Configuración** » **Formato local** hay que escribir algo como lo que se muestra a continuación:

```
Format 95
  InsetLayout Flex:DiscrLig
  LyxType          charstyle
  LabelString      discrlig
  LatexType        command
  LatexName        discrlig
  Preamble
    \newcommand*{\discrlig}[1]{%
      \addfontfeature{RawFeature={+dlig}} #1}}
  EndPreamble
  ResetsFont      true
End

InsetLayout Flex:LogoTipo
  LyxType          charstyle
  LabelString      HoLogo
  LatexType        command
  LatexName        loguito
```

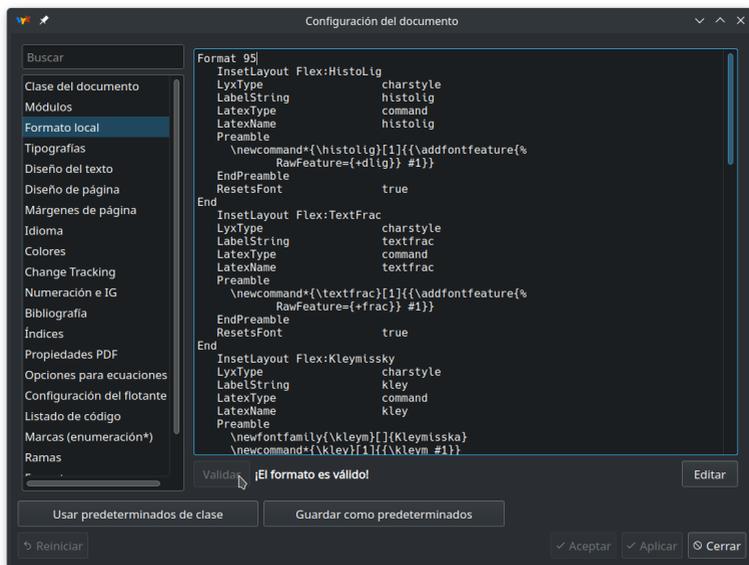


FIGURA 9.2: crear un formato local

```

Preamble
  \newcommand*{\loguito}[1]{\hologo{#1}}
EndPreamble
ResetsFont           true
End

```

NOTAS:

- El Format 95 del inicio indica la versión del sistema utilizado por LyX 2.4 para crear los estilos (en versiones posteriores podría tener otros números). Esta primera línea es obligatoria, *pero solo necesitas escribirla una vez* si defines varios formatos locales.
- Luego de los dos puntos en la segunda línea NO debe quedar espacio. Esta línea define el nombre del nuevo formato.
- LyxType define el tipo de formato que se está creando: en este ejemplo un estilo de carácter (charstyle) mientras que en el ejemplo

que dejé caer en el capítulo anterior era un «recuadro personalizado» (custom).

- `LabelString` es la etiqueta que verás en la ventana de edición de LyX marcando el formato aplicado.
- `LatexType` define que se está aplicando un comando, sea este de base o, como en el ejemplo, definido por el usuario.
- `LatexName` es el nombre (sin la barra inicial) del comando \LaTeX a ser aplicado.
- Si quieres que *en la ventana de edición* tu estilo se presente en forma diferente, puedes variar el tipo de fuente haciendo algo como lo siguiente:

```
Font
```

```
    Family      Serif
    Shape       Italic
```

```
EndFont
```

- La parte entre `Preamble` y `EndPreamble` sirve para definir código que tendría que estar en el preámbulo \LaTeX . Evidentemente, si el código en cuestión ya está en el preámbulo no es necesario definirlo aquí.
- `ResetsFont` sirve para que el formato aplicado no se extienda más allá del texto seleccionado.
- Antes de poder aceptar, tienes que utilizar el botón *validar*: solo si LyX confirma que el código está bien escrito podrás seguir adelante.
- También es posible crear *entornos*, es decir, estilos de párrafo. Eso está ya explicado con lujo de detalles en los manuales de LyX ([Ayuda](#) [Personalización](#) [sección 5.3](#)), pero si quieres un ejemplo, mira la sección siguiente.

Ahora solo queda seleccionar el texto, hacer sobre él un [clic derecho](#) [Estilo de texto personalizado](#) [DiscrLig](#) y aquello que en la ventana de edición de LyX se ve más o menos así

texto normal esto es estándar histórico actual esto es texto normal
disrllig

en el resultado se verá así:

texto normal esto es estándar histórico actual esto es texto normal

Mientras que

XeTeX AmS
HoLogo HoLogo

se verá así:

X_fT_EX A_MS

Para *quitar* un estilo de texto ya aplicado es suficiente hacer clic derecho

Estilo de texto personalizado Disolver estilo de texto

En la subsección 16.1.2 te daré otro ejemplo de formato local mientras que en el apéndice B encontrarás la definición de todos los estilos utilizados para crear este libro.

9.4 ¿Entorno de párrafo?

Solo un ejemplo rápido y sin explicación para crear un entorno de párrafo con un color de fondo y un borde. Utiliza los paquetes `mdframed` y `xcolor` que, por supuesto, tienen que estar instalados:

```
Format 95
Style MyParagraph
Category MainText
Margin Static
LatexType Environment
LatexName myparagraph
NextNoIndent 1
LeftMargin MMM
RightMargin MMM
ParIndent MM
ParSkip 0
ItemSep 0
TopSep 0.5
```

```

BottomSep      0.5
ParSep         0
Align          Block
AlignPossible  Block, Left, Right, Center
Preamble
\usepackage{mdframed}
\usepackage{xcolor}
\newenvironment{myparagraph}
  {\begin{mdframed}[
    backgroundcolor=lightgray,
    linecolor=lightgray
  ]\quotation}
  {\endquotation\end{mdframed}}
EndPreamble
End

```

¡Trata de comprender el código! Solo juega un poco con los parámetros, ¡es divertido!

9.5 Macros en modo matemático para LyX

Considera las siguientes situaciones matemáticas:

SITUACIÓN 1 Podrías tener una expresión matemática compleja que se repita, con pequeñas variantes, varias veces en el mismo documento. Por ejemplo, podrías estar escribiendo un ensayo sobre transformadas de Fourier, lo que te obligaría a introducir muchas veces una expresión donde solo cambiaría la función a transformar, no el resto de la integral.

SITUACIÓN 2 LyX no puede dar soporte a cada comando L^AT_EX existente por lo que a veces tendrás que recurrir a escribir código directamente, código que, por si fuera poco, podría verse extraño en la ventana de edición.

Afrontar cualquiera de estas situaciones significaría un enorme trabajo manual de ajustar detalles y compilar para ver cómo va quedando, trabajo que se repetiría hasta el cansancio... o no, o al menos no siempre.

Te doy la bienvenida a una de las características más interesantes, pero menos conocidas, de LyX: *las macros matemáticas*.

La idea es crear una macro con las instrucciones \LaTeX necesarias (la parte fija de la expresión), los argumentos que te sirvan (la parte que cambiará cada vez) y darle a todo esto un breve nombre. Una vez creada esta macro solo será cuestión de llamarla desde cualquier lugar del documento utilizando su nombre, que la única tarea pendiente será la de llenar los huecos dejados por los argumentos.

Con cada nueva versión de LyX la SITUACIÓN 2 resulta cada vez más rara, ya que los comandos \LaTeX que quedan sin implementar son justamente los más exóticos, por lo que te comentaré cómo resolver la SITUACIÓN 1 que en el proceso resultará claro cómo funciona todo.

Para esto utilizaré el comando `mathop` para el cual LyX tiene soporte gráfico. Este comando sirve para convertir cualquier signo o texto en un operador que acepte límites.

Primero debes ir a    que aparecerá algo como lo mostrado en la imagen superior de la figura 9.3.

Como puedes ver, en el documento se abrirá una caja que será utilizada para crear tu macro mientras que debajo tendrás, además de las barras de herramientas tradicionales del editor de ecuaciones, una nueva para crear o quitar argumentos, hacerlos opcionales, etcétera.

NOTA: Esta caja no aparecerá en el documento compilado por lo que no tiene que preocuparte su presencia. De hecho podrías colocarla en cualquier lugar, pero es mejor agrupar todas las macros al inicio del documento para así tenerlas más a mano.

Ahora reemplaza `newmacroname` por algo que te sea cómodo. Dado que en este ejemplo queremos simplificar el uso del comando `\mathop` podemos llamarlo `mop`.

IMPORTANTE: ¡Cuidado con el nombre elegido, que no puedes utilizar un comando \LaTeX preexistente!

En la primera de las dos pequeñas subcajas, la marcada como TeX, tienes que escribir el código \LaTeX mientras que en la segunda, marcada como

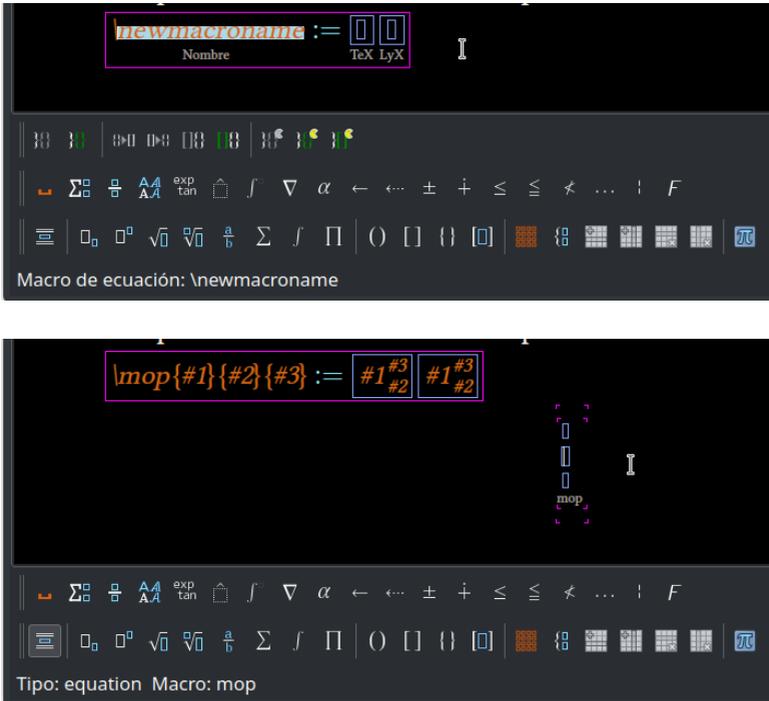


FIGURA 9.3: crear una macro matemática. Arriba, iniciar el proceso; abajo, cómo se ve todo en la ventana de LyX, con la expresión esperando a que los tres parámetros sean completados.

LyX, aquello que quieres que LyX muestre durante el uso de la macro. Dado que `\mathop` tiene soporte gráfico, simplemente escribes lo mismo en ambas cajas.

Y aquí es donde puedes ver cómo resolver la SITUACIÓN 2: en la segunda caja *no* escribes lo mismo que en la primera, sino algo que te permita entender qué obtendrás en la salida.

Recuerda que en el editor de ecuaciones de LyX la forma más simple de insertar los delimitadores `{}` para agrupar expresiones es escribiendo `\{` (el de cierre aparecerá automáticamente).

A continuación utiliza la nueva barra de herramientas para agregar el número de argumentos necesarios, que en este caso son tres, y aplícales las instrucciones necesarias a cada uno (`\mathop` al primer parámetro, subíndice al segundo y superíndice al tercero), lo que te dará lo que se muestra en la imagen inferior de la figura 9.3.

Ya la tienes. Para utilizar la flamante macro solo es necesario escribir dentro de una ecuación

```
\mop
```

para luego presionar la barra de espacio una vez. Ahora tendrás tres cajas esperando contenido.

Finalmente, poniendo un igual en el primer parámetro, dejando vacío el segundo y escribiendo `def` en el tercero, o bien escribiendo la suma para i y j iguales a uno hasta N en el primer parámetro, $i > j$ en el segundo y dejando vacío el tercero, en el documento compilado (recuerda que la caja que define la macro no aparecerá) obtendrás

$$f(x) \stackrel{def}{=} \ln(x)$$

$$\sum_{\substack{i,j=1 \\ i>j}}^N A_{ij}$$

Puedes anidar tus expresiones o, como ves, dejar algunas de las cajas en blanco: el único límite lo pone tu creatividad a la hora de resolver estos problemas.

¡Todo listo! Como puedes ver, las macros matemáticas en LyX resultan una herramienta increíblemente flexible y simple de utilizar, ¡no dejes de aprovecharlas!

Si quieres *ver* un ejemplo de macro matemática en acción, tienes este breve vídeo de YouTube:

<https://www.youtube.com/watch?v=VG9PHl3ToVk>

Allí te muestro cómo resolver la SITUACIÓN 2, definiendo una instrucción para la cual no hay (aún) soporte gráfico en LyX.

Capítulo 10

Listas numeradas

Como ya comenté en la introducción, modificar el formato de las listas numeradas tiene sus vueltas. En este capítulo te contaré cómo hacer que estén numeradas con números romanos, o con letras. Te mostraré también cómo hacer esto para todas las listas de tu documento o para una particular.

Eso sí, asegúrate de tener el paquete `enumitem` instalado en tu sistema, que de lo contrario nada funcionará.

10.1 Modificar todas las listas del documento

La primera parada es `Documentos` » `Configuración` » `Módulos`. Allí, en el bloque con las mejoras para las listas, selecciona `LISTAS PERSONALIZADAS (ENUMITEM)` y presiona `AÑADIR`.

Ahora tienes que ir al preámbulo, pero antes piensa qué quieres lograr para tus listas. Podrías hacer que el primer nivel fuera numerado con letras mayúsculas, que el segundo esté marcado con números, el tercero con numerales romanos y el cuarto con letras en minúscula y en cursiva, y que cada subnivel muestre también los números anteriores. Pues bien, este sería el código:

```
\setlist[enumerate,1]{label = \Alph*}
\setlist[enumerate,2]{label =
    \theenumi.\arabic*}
\setlist[enumerate,3]{label =
```

```

\theenumii.\Roman*}
\setlist[enumerate,4]{label =
\theenumiii.\textit{\alph*}}

```

Cómo dar formato a los números y elegir los estilos es algo que ya hemos visto en capítulos anteriores. Los códigos `\theenumi`, `\theenumii` y `\theenumiii` se refieren a la numeración de los niveles 1, 2 y 3, respectivamente y vienen de *the enumeration <#>*, donde el número está escrito en... ¡*numerales romanos en minúscula!* Pues sí, el cuarto nivel sería `\theenumiv`, por suerte no lo necesitamos.

En lugar de los puntos luego de cada `\theenum#` puedes utilizar lo que quieras, por ejemplo un paréntesis o incluso una instrucción matemática como `\dagger` para obtener un †. También puedes agregar un separador luego del último número. Eso sí, trata de no exagerar.

Si no quieres mostrar los niveles anteriores, puedes utilizar algo así:

```

\setlist[enumerate,1]{label = \Alph*}
\setlist[enumerate,2]{label = \arabic*}
\setlist[enumerate,3]{label = \Roman*}
\setlist[enumerate,4]{label = \textit{\alph*}}

```

10.2 Modificar algunas listas numeradas

En la sección anterior te mostré cómo modificar simultáneamente todas las listas numeradas del documento, ahora verás cómo crear un formato local para modificar listas seleccionadas.

Pues bien, en Documento » Configuración » Formato Local

```

Format 95
Counter legali
LabelString "\arabic{legali}."
End

Counter legalii
Within legali
LabelString "\thelegali
\arabic{legalii}."

```

```

End
Counter legaliii
      Within      legalii
      LabelString "\thelegalii
                  \arabic{legaliii}."
End
Counter legaliv
      Within      legaliii
      LabelString "\thelegaliii
                  \arabic{legaliv}."
End
Style   Legal
      CopyStyle Enumerate
      LatexName legal
      LabelCounter "legal"
      RefPrefix leg
      Requires enumitem
      Preamble
          \newlist{legal}{enumerate}{4}
          \setlist[legal]{label*=\arabic*}
      EndPreamble
End

```

Validas, aceptas y ya puedes trabajar: este formato local agregará un estilo de lista llamado «Legal» al menú desplegable de los estilos de párrafo que se encuentra bajo el menú Archivo. Te dejo como ejercicio interpretar el código y modificarlo para obtener otros formatos.

Al final no fue tan difícil.

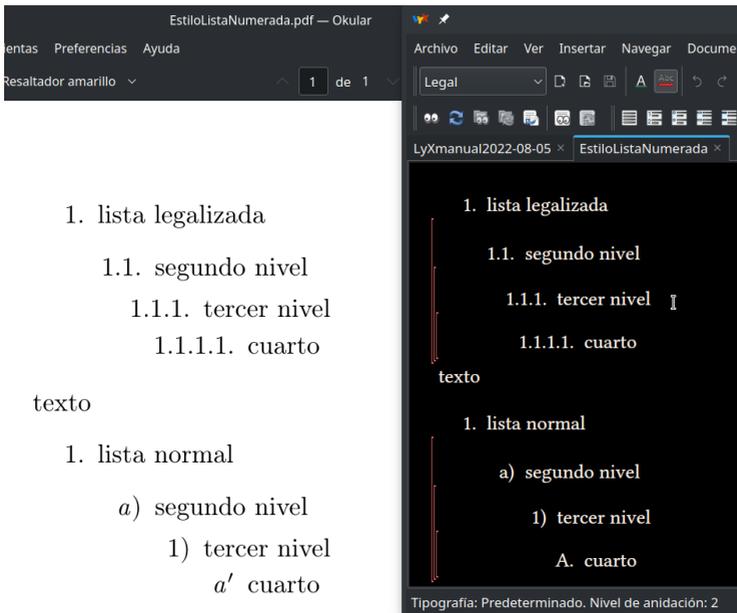


FIGURA IO.I: *ejemplo de listas modificadas. La primera lista ha sido creada con el estilo de lista introducido en el texto, mientras que la segunda usa el formato global comentado en la primera sección de este capítulo (si bien con otro formato).*

Capítulo 11

Sobre los índices

Configurar índices es quizás uno de los puntos más oscuros de \LaTeX . En este capítulo te daré una breve introducción a algunas herramientas para resolver cuestiones específicas relacionadas con los índices.

11.1 Títulos cortos

Comencemos con algo simple: ¿qué tal si el título de un capítulo o sección te ha quedado demasiado largo y quieres que tanto el índice general como la cabecera de la página muestren una versión más corta del mismo?

Con el cursor en el título en cuestión: , o bien , o quizás, para los más memoriosos, , luego . La caja que se presenta te permitirá escribir una versión más corta del título para ser utilizada en el índice y en las referencias cruzadas, sobrescribiendo el título real.

Ahora algo más complicado: ¿cómo tener distinto texto en el título, en el índice y *en la cabecera de la página*?

Luego de insertar el título corto puedes abrir una caja \TeX con el siguiente código para los capítulos

```
\chaptermark{Título de capítulo para cabecera}
```

o lo siguiente para secciones

```
\sectionmark{Título de sección para cabecera}
```

Ten en cuenta que estas instrucciones pueden ponerse *a continuación* de los títulos, no necesariamente en la misma línea. De hecho, puedes volver a introducirlas más tarde para volver a cambiar el texto mostrado en la cabecera para identificar capítulos y secciones. ¡Úsalas todas las veces que las necesites!

11.2 Notas al pie en títulos

Imagina que por algún motivo quieres colgar una nota al pie en un título de sección. Por ejemplo, tienes un artículo con varios autores, cada uno encargándose de una sección, y el editor decide indicar quién escribió cada sección con una nota al pie de página anclada en el título. ¿El problema? Que si insertas las notas al pie normalmente y luego creas un índice general, las notas ancladas a los títulos se repetirán en el índice, ¡todas esas notas duplicadas!

El desafío entonces es lograr que la nota al pie esté solo en el título que le corresponde, no en el índice. La solución es utilizar lo que vimos en la sección anterior: títulos cortos.

Cuando insertas un título corto, por defecto LX copia el título ya ingresado en el párrafo, por lo que el truco es la siguiente:

1. Insertar el título original.
2. Insertar el título corto.
3. Volver con el cursor al final del título original.
4. Insertar la nota al pie.

Listo. Dado que el título corto es lo que va al índice, la nota al pie no será copiada.

11.3 Cambiar el nombre del índice

Digamos que no te gusta tener un «índice general» y que quieres en cambio tener la lista de los «contenidos». Pues para esto puedes utilizar Babel.

No necesitas aquí modificar las opciones por defecto de Babel, por lo que sin tocar nada en las opciones de idioma de la configuración del documento, ve al preámbulo para agregar el siguiente código:

```
\AtBeginDocument{
  \addto\captionsspanish{
    \renewcommand{\contentsname}{Contenidos}
  }}

```

Eso es todo. Si el documento que escribes es en inglés, simplemente tienes que cambiar a `\captionsenglish` en la segunda línea. Lo mismo para otros idiomas.

11.4 Índices parciales para cada capítulo

En el preámbulo \LaTeX debes escribir lo siguiente:

```
\usepackage{minitoc}

```

(obviamente, el paquete `texlive-minitoc` debe estar instalado). En el documento, justo antes del índice general inserta una caja \TeX con el comando

```
\dominitoc

```

Finalmente, por cada capítulo, justo debajo del nombre del capítulo y antes de cualquier sección inserta otra caja \TeX con el comando

```
\minitoc

```

11.5 Corregir espaciado en listas de figuras o cuadros

Cuando se usa una de las *documentclass* de artículo las figuras se numeran en secuencia sin interrupción, mientras que si se usa el formato libro la numeración por defecto de las figuras es <capítulo>.<número>.

En \LaTeX es muy simple el activar una numeración por secciones, que para los artículos será <sección>.<número> mientras que en los libros te

dará <capítulo>.<sección>.<número>: es suficiente dirigirse a Documento Configuración Módulos (ya hablamos de los módulos en el capítulo 9) y activar `NUMERAR FIGURAS POR SECCIÓN`.

El problema con esto surge cuando tienes una *documentclass* de libro y creas el índice de figuras: el número extra ocupa más espacio que el previsto en la configuración por defecto del índice por lo que la numeración choca con el texto que le sigue.

Para conseguir más espacio entre número y texto hay que dirigirse al preámbulo \LaTeX y escribir lo siguiente:

```
\makeatletter
\renewcommand*\l@figure{\@dottedtocline{1}{
  1.5em}{3em}}
\makeatother
```

Para variar la distancia solo necesitas jugar con el valor de `{3em}` que se encuentra a la derecha de la línea central del código indicado.

Si quieres hacer lo mismo con el índice de cuadros, agrega

```
\renewcommand*\l@table{\@dottedtocline{1}{
  1.5em}{3em}}
```

entre `\makeatletter` y `\makeatother` y ya está todos listo.

Para profundizar un poco en este tema puedes consultar los siguientes enlaces:

<http://tex.stackexchange.com/questions/133747/reduce-space-between-table-number-and-title-in-list-of-tables>
<http://tex.stackexchange.com/questions/87019/list-of-figures-numbers-overlap-image-caption>

11.6 Evitar las líneas de puntos en los índices

Por defecto, muchas *documentclass* usan en los índices líneas punteadas como guías que unen el texto del título con el número de página correspondiente.

Imagina que quieres eliminar estas líneas punteadas. En el preámbulo \LaTeX agrega lo siguiente:

```
\renewcommand{\@dotsep}{10000}
```

Listo. Este comando actuará sobre *todos* los índices del documento.

Lo que sucede aquí es que el comando `\@dotsep` controla la separación entre los puntos (*dot separation*, en inglés), por lo que si en lugar de 10000 usas valores como 20 o 30 simplemente obtendrás una separación notablemente mayor que en la configuración por defecto. El valor 10000 es especial en \LaTeX y se usa para decir «infinito» en cualquier instrucción que necesite un parámetro numérico, por lo que en este caso termina deshabilitando la línea punteada.

11.7 Agregar una línea con texto al índice

Imagina que quieres agregar la palabra **Página** en negrita sobre los números de página del índice. Pues bien, justo debajo del punto donde se ha insertado el índice general es suficiente insertar, con `Ctrl` `L`, una caja \TeX para agregar el siguiente código:

```
\addtocontents{toc}{~\hfill\textbf{Página}\par}
```

Aquí se está insertando al inicio del índice un párrafo que solo contenga un «relleno horizontal» y la palabra **Página** en negrita. Ya está. Y sí, también puede utilizarse esto para los otros tipos de índices, solo tienes que sustituir `toc` (*table of content*) con `lof` (*list of figures*) para la lista de figuras o con `lot` (*list of tables*) para la lista de cuadros (¡que no son tablas!).

Puedes usar este truco para agregar cualquier texto en cualquier punto del índice, solo tienes que agregar una de estas cajas \TeX en el lugar apropiado. Como ejercicio, te dejo pensar cómo hacer para agregar el texto «Aquí viene lo bueno» justo antes de un capítulo determinado.

11.8 Quitar el número de página de las entradas para las Partes

Si usas una clase de libro, podrías querer agrupar los capítulos en Partes. Y también podrías querer que el índice no muestre el número de página de las partes, que después de todo solo los capítulos importan.

Para esto, luego de asegurarte de que el paquete `tocloft` esté instalado (y luego de reconfigurar $\text{L}\text{\AA}\text{X}$), escribe en el preámbulo \LaTeX lo siguiente:

```
\usepackage{tocloft}
\cftpagenumbersoff{part}
```

Aquí se activa el mencionado paquete y se utiliza una de sus opciones para eliminar el número de página de las partes.

11.9 Unas (pocas) palabras sobre los índices alfabéticos

En \LaTeX existen muchas formas de generar índices alfabéticos y $\text{L}\text{\AA}\text{X}$ ciertamente permite utilizarlas todas.

Podemos elegir entre `makeindex`, `texindy` o `xindy`. Si no sabes de lo que te estoy hablando, a no preocuparse que el problema no es tuyo: es este un tema complejo que requiere estudio para poder aprovecharlo. Esto significa que el hablar en detalle sobre los índices alfabéticos se escapa del propósito general de este libro por lo que solo diré lo mínimo indispensable. Queda a tu curiosidad el profundizar tanto en la guía del usuario de $\text{L}\text{\AA}\text{X}$ (sección 6.6) como en los distintos manuales \LaTeX disponibles en línea.

Para marcar una palabra o texto a ser incluido en el índice alfabético solo tienes que seleccionarla y dirigirte a [Insertar » Entrada de índice](#). Esto te dará una caja con la marca justo después del texto marcado. Por defecto el contenido de la marca será igual al texto seleccionado, pero puedes cambiarlo sin problemas.

El índice propiamente dicho se introduce en cualquier parte del documento con [Insertar » Listas/Contenidos/Referencias » Alfabético](#). A diferencia del índice general, haciendo clic sobre el botón resultante no sucede nada, pero es posible configurar el procesador para generar el índice alfabético en [Documento » Configuración » Índices](#). Allí además de elegir entre `makeindex`, `texindy` o `xindy` puedes pasar opciones a estos procesadores, elegir de crear más de un índice (podrías tener uno de palabras, otro de nombres, etcétera) y administrar los índices existentes.

Capítulo 12

Títulos

En este capítulo te mostraré cómo modificar la presentación y la numeración de los títulos de un documento, sean estos de «parte», «capítulo», «sección», etcétera.

12.1 Renombrar los entornos Parte y Capítulo

Imagina que quieres escribir un libro con $\text{L}\text{X}\text{E}\text{X}$ y seleccionas, por ejemplo, la *documentclass* Book.

De esta forma tendrás a disposición varios entornos, entre ellos Capítulo (comando $\text{L}\text{A}\text{T}\text{E}\text{X}$ `\chapter{}`) para separar las distintas partes de la obra y Parte (comando $\text{L}\text{A}\text{T}\text{E}\text{X}$ `\part{}`) para agrupar capítulos.

Pero supón que en lugar de separar en Partes quieres separar en Libros y que en lugar de que aparezca «Capítulo 1: Título elegido» salga «Jornada 1: Título elegido». ¿Cómo se pueden renombrar estos entornos?

La inspiración viene del capítulo anterior, donde te mostré cómo cambiar el nombre del índice. Como siempre, en el preámbulo $\text{L}\text{A}\text{T}\text{E}\text{X}$ tienes que escribir el siguiente código:

```
\AtBeginDocument{
  \addto\captionsspanish{
    \renewcommand{\partname}{Libro}
    \renewcommand{\chaptername}{Jornada}
  }}
}
```

Básicamente se está engañando a \LaTeX diciéndole que en lugar de utilizar las traducciones predefinidas para `Part` y `Chapter` (Parte y Capítulo, respectivamente) utilice las que le indicas.

De hecho, si no quieres nada allí simplemente dejas las llaves vacías: `\renewcommand{\partname}{}{}` mostrará solo el número y el título que escribas, sin etiquetas.

En la ventana de edición de LyX nada cambiará, pero en la salida PDF todo se mostrará como quieres.

12.2 Numerar capítulos con números romanos

Para hacer que los capítulos estén numerados con números romanos hay que modificar el contador de los mismos aplicándole un formato, y para esto se utiliza el comando `renewcommand`:

```
\renewcommand{\thechapter}{\Roman{chapter}}
```

Para utilizar letras en mayúscula es suficiente cambiar `Roman` por `Alph`, tal y como te mostré en el capítulo 6.

12.3 Iniciar los capítulos en cualquier página

Por defecto en las *documentclass* de libros los capítulos comienzan siempre en una página nueva que es además una página derecha (impar). Si te perdiste el fugaz comentario en el capítulo 5, lograr que los capítulos comiencen en cualquier página, sea esta izquierda o derecha, es realmente simple, en `Documento` `Configuración` `Clase del documento` `Opciones de clase` `Personalizado` es suficiente escribir:

```
openany
```

12.4 Evitar los quiebres de página al comenzar un capítulo

No es que me guste, pero ya sabes, por completitud.

Imagina que por algún motivo quieres evitar completamente que se produzca un quiebre de página al comienzo de cada capítulo para así lograr que un capítulo comience en la misma página donde ha terminado el anterior. Para lograr esto hay que redefinir el entorno capítulo por lo que en el preámbulo \LaTeX tienes que escribir lo siguiente:

```
\makeatletter
\renewcommand\chapter{%
    \global\@topnum\z@
    \@afterindentfalse
    \secdef\@chapter\@schapter}
\makeatother
```

Este código lo vi aquí:

[http://compgroups.net/comp.text.tex/
latex-report-class-page-break-after-every-chapt/1924905](http://compgroups.net/comp.text.tex/latex-report-class-page-break-after-every-chapt/1924905)

12.5 Cambiar el formato de los títulos en \LaTeX

Como ya vimos, la *documentclass* elegida determina el formato general del documento y esto ciertamente incluye cómo se muestran los títulos de capítulos, secciones, etcétera.

En la mayor parte de las *documentclass* se utiliza para los títulos la misma fuente del cuerpo de texto. Esto generalmente da buenos resultados estéticos, ¿pero qué tal si se quiere cambiar este formato? ¿O cambiar el color del texto? ¿O agregar efectos especiales? Por ejemplo, podrías querer una caja rodeando el título o utilizar un tamaño diferente.

Existen varias formas de lograr cambios en el formato de los títulos y algunas *documentclass* como Koma Script ofrecen herramientas para lograr algunas de estas cosas, pero la opción que te presentaré a continuación es quizás la más general posible (si bien no es completamente general ya que no funciona con la clase de documento Memoir).

12.5.1 Los paquetes a utilizar

Generalmente se instalan por defecto, pero no está de más confirmar que los paquetes `xcolor` y `titlesec` estén instalados en la distribución

LaTeX.

La documentación de `xcolor` suele instalarse, en Linux y con texlive, en:

```
/usr/share/texmf/doc/latex/xcolor/
```

mientras que la documentación de `titlesec` se instala en:

```
/usr/share/texmf/doc/latex/titlesec/
```

Daré aquí indicaciones generales sobre estos paquetes que resultarán más que suficientes en la mayor parte de los casos. Para más información y otros usos extremos, siempre es posible leer el magnífico manual.

12.5.1.1 `xcolor`

El paquete `xcolor` sirve para definir, sí, lo adivinaste, colores. La forma más simple de utilizarlo es declararlo en el preámbulo LaTeX con, por ejemplo

```
\usepackage{xcolor}
\definecolor{AzulClaro}{rgb}{.3,.5,.7}
```

La primera línea declara el paquete mientras que la segunda define un color. Los parámetros entre llaves son, respectivamente, el nombre del color, el método de definición y los valores de cada color básico en forma de tres números entre cero y uno (en este caso, cuánto rojo, cuánto verde y cuánto azul), separados por comas.

12.5.1.2 `titlesec`

Este paquete ofrece varias instrucciones para realizar los cambios en los títulos. Se puede utilizar en dos formas, una simplificada y otra completa.

La forma simplificada solo puede utilizarse en secciones, subsecciones y niveles más bajos, mientras que para los entornos Parte y Capítulo (o para disponer de toda la pirotecnia que ofrece el paquete) debe utilizarse la forma completa.

12.5.2 Un ejemplo completo

En el preámbulo LaTeX prueba a escribir lo siguiente:

```

\usepackage{titlesec}
\usepackage{xcolor}
\definecolor{AzulClaro}{rgb}{.3,.5,.7}
\definecolor{Veeerde}{rgb}{.3,.7,.3}
\newfontfamily{\coral}{TeX Gyre Chorus}
\titleformat{\chapter}[frame]
  {\LARGE\coral\color{AzulClaro}}{
  \filright\chaptertitlename\
  \thechapter}{20pt}{\Huge}
\titleformat*{\section}{
  \Large\bfseries\sffamily
  \color{Veeerde}}

```

Las dos primeras líneas declaran el uso de los paquetes `titlesec` y `xcolor`.

Las dos siguientes definen dos colores, `AzulClaro` y `Veeerde` (no, no fue un error de escritura).

La quinta línea define una nueva familia tipográfica para utilizar en el documento. Ya te hablé de eso en la sección [7.3](#).

En el ejemplo puede verse tanto la forma simple como la completa de la instrucción `\titleformat`, parte del paquete `titlesec`.

La forma con el asterisco es la versión resumida de la instrucción y ha sido utilizada para definir rápidamente el formato de la sección: aplicada al entorno `\section`, usa una fuente en tamaño `\Large`, en negrita, con la fuente palo seco definida para el documento (`\sffamily`) y le asigna el color `Veeerde` que hemos definido.

La forma completa de la instrucción es usada para modificar el entorno `\chapter` (capítulo). Comienza indicando que se debe construir un marco alrededor del título (`[frame]`): si no quieres el marco escribe `[display]`). Luego se tiene el tamaño `\LARGE` para el texto de la etiqueta (la palabra Capítulo seguida del número), la familia de fuente que llamé *coral*, el color *AzulClaro*. Luego vienen una serie de instrucciones de formato que comienzan con *alineado a la derecha*, siguen dando la estructura de la información presentada (incluyendo un espacio de separación entre etiqueta y texto del título) y terminan indicando que el texto del título debe tener el tamaño

Capítulo 1

Escritos de «El pingüino tolkiano»

Texto general con <http://es.lipsum.com/>

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Quisque in massa vel risus facilis finibus. Duis auctor mi quam, sit amet vestibulum tortor iaculis at. Ut malesuada nisi vel molestie commodo. Sed non libero dolor. Aenean ac cursus est. Vestibulum et ultricies neque. Suspendisse ac tincidunt nisi, ac dignissim ligula.

1.1 Título de sección

Nulla sagittis, purus sed suscipit bibendum, lacus purus placerat quam, id egestas libero dolor in ligula. Donec molestie tincidunt neque id fringilla. Suspendisse egestas tempor feugiat. Aliquam placerat mi vel metus dapibus imperdiet. Aliquam finibus sapien felis, non

FIGURA 12.1: *un ejemplo de títulos modificados con la instrucción titlesec*

\Huge, que es decididamente mayor que el de la etiqueta. Puedes ver el resultado de estas instrucciones en la captura de pantalla de la figura 12.1.

Claramente esto también puede hacerse para el entorno Parte, agregando por ejemplo:

```
\titleformat{\part}[display]
  {\huge\cormorant}{\filright\partname\
  \thepart}{20pt}{\Huge}
```

Aquí no he utilizado el marco y elegí un tamaño mayor para la etiqueta de la parte (además de utilizar una familia de fuentes definida en otra parte).

Y bien, los ingredientes básicos ya están por lo que solo queda compilar el documento. Las instrucciones aquí mostradas son aún más potentes, pero no entraré en más detalles.

12.6 Reiniciar la numeración luego de una Parte

Por defecto L^AT_EX no reinicia la numeración de los capítulos luego de un entorno Parte, dando una numeración continua:

- Parte I
 - Capítulo 1
 - Capítulo 2
- Parte II
 - Capítulo 3

Si bien esta opción por defecto me parece lógica y de hecho la uso (en mi libro sobre Writer propongo trucos para conseguirla también allí), es perfectamente posible y por lo tanto completamente válido que alguien necesite la opción contraria, es decir:

- Parte I
 - Capítulo I.1
 - Capítulo I.2
- Parte II
 - Capítulo II.1

La forma más simple de conseguir esto es mediante el paquete `chngcntr`. Desde el 2018 este paquete está integrado en $\LaTeX 2_{\epsilon}$, por lo que si usas una versión reciente de \TeX Live no es necesario llamarlo explícitamente, simplemente puedes utilizar las instrucciones que ofrece (`counterwithin` y `counterwithout`).

La documentación, al menos en Linux, se instala en:

`/usr/share/texmf/doc/latex/chngcntr/chngcntr.pdf`

Son solo dos páginas, por lo que te recomiendo leerla.

Para decir que `chapter` sea contado *dentro* de `part`, es decir, que reinicie la numeración de los capítulos cada vez que `part` cambia, en el preámbulo \LaTeX debes escribir lo siguiente:

```
\counterwithin{chapter}{part}
```

Listo, ¿no? No: al agregar la Parte a la numeración de los capítulos (y de las figuras, y de las tablas) tendrás problemas con el índice, ya que el número se quedará sin espacio y se superpondrá con el texto del título. Ya te mostré cómo resolver esto para las listas de figuras y cuadros en el capítulo 11, por lo que siguiendo aquel ejemplo, luego de las dos líneas anteriores también tendrás que agregar en el preámbulo lo siguiente:

```
\makeatletter
\renewcommand\l@chapter{
  \@dottedtocline{2}{1.5em}{3em}}
\makeatother
```

Para agregar espacio en las líneas del índice correspondientes a las secciones duplica la segunda instrucción mostrada más arriba, pero cambiando `l@chapter` por `l@section`:

```
\makeatletter
\renewcommand\l@chapter{
  \@dottedtocline{2}{1.5em}{3em}}
\renewcommand\l@section{
  \@dottedtocline{2}{1.5em}{3em}}
\makeatother
```

Y así siguiendo con las subsecciones (`subsection`), etcétera. Ahora sí, ¡listo!

Aunque no es algo que verás en este libro.

Capítulo 13

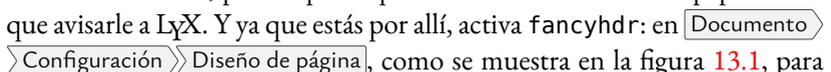
Cabeceras, pie y estilos de página

En \LaTeX y derivados, para modificar los valores por defecto de cabeceras y pie de página se puede utilizar el paquete `fancyhdr`. LyX ofrece soporte parcial para este paquete en su interfaz gráfica, pero de tan parcial resulta poco práctico por lo que tendremos que ir al preámbulo \LaTeX .

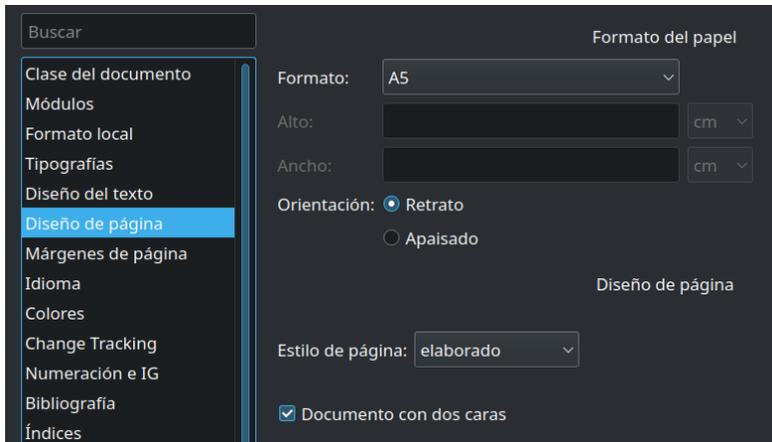
Las cabeceras y pie de página del documento están asociadas a estilos de página, tema que será tratado sobre el final de este capítulo. Antes te contaré cómo cambiar el contenido de las cabeceras y pie de las páginas «normales» para finalmente llegar a la creación y el uso de estilos especiales.

13.1 Preparativos

La documentación de `fancyhdr` se instala, en Linux y con TeXLive , en `/usr/share/texmf/doc/latex/fancyhdr/`

Por defecto los documentos \LaTeX están configurados como «páginas con un solo lado», por lo que si quieres usar las dos caras del papel tienes que avisarle a LyX . Y ya que estás por allí, activa `fancyhdr`: en  Documento > Configuración > Diseño de página], como se muestra en la figura 13.1, para ESTILO DE PÁGINA selecciona ELABORADO y marca la opción DOCUMENTO CON DOS CARAS.

Ahora sí, en este mismo menú puedes abrir el Preámbulo \LaTeX y comenzar a escribir... pero antes, una pequeña introducción a *qué* se puede escribir.

FIGURA 13.1: *activar fancyhdr*

13.2 El paquete **fancyhdr**

El paquete `fancyhdr` permite cambiar el contenido de las cabeceras (*head*) y los pie de página (*foot*). Ambos sectores están divididos en tres partes: izquierda (*left* o *l/L*), central (*center* o *c/C*) y derecha (*right* o *r/R*).

Existen dos modos de utilizar este paquete. La forma más directa, útil para documentos «de un solo lado», es con las instrucciones

```
\lhead{contenido de la cabecera a la izquierda}
\chead{contenido de la cabecera al centro}
\rhead{contenido de la cabecera a la derecha}
\lfoot{contenido del pie de página a la izquier-
da}
\cfoot{contenido del pie de página al centro}
\rfoot{contenido del pie de página a la derecha}
```

Más abajo verás ejemplos del tipo de contenido que puedes utilizar.

La forma más potente de este paquete es con la instrucción `\fancyhead` para las cabeceras y `\fancyfoot` para los pie de página, teniendo en cuenta la siguiente convención:

R representa «derecha» (por *right*)

- L representa «izquierda» (por *left*)
- C representa «centro» (por *center*)
- O representa las páginas impares (por *odd*)
- E representa las páginas pares (por *even*)

La instrucción

```
\fancyhead[R0,LE]{CONTENIDO}
```

colocará por lo tanto CONTENIDO a la derecha en las páginas impares y a la izquierda en las pares. Si escribes

```
\fancyfoot{}
```

tendrás el pie de página vacío.

Ahora bien, ¿qué es lo que se puede poner en CONTENIDO?

Para empezar, el texto puro funciona perfectamente. De hecho, si activas X_YTeX y defines una nueva familia de fuentes (como te conté en la sección 7.3) podrás colocar texto tan extraño como quieras.

También tienes instrucciones para insertar el número de página y el capítulo. Por ejemplo:

```
\thepage
```

inserta el número de página

```
\leftmark
\rightmark
```

se usan (estoy simplificando mucho aquí, pero para lo que tenemos que hacer nos sirve) para información del capítulo y la sección, respectivamente.

NOTA: si por algún motivo falta información para una de estas marcas (por ejemplo, en un capítulo determinado no tienes secciones) tendrás un espacio vacío en las páginas correspondientes.

En general, el nombre del capítulo aparecerá en mayúsculas. Para evitar esto puedes utilizar la instrucción `\nouppercase{}` (la verás en acción en el ejemplo). Se debe tener cuidado con esta instrucción ya que si las páginas se numeran con números romanos, pues estos también podrían aparecer en minúsculas... Sí, \LaTeX puede ser extraño.

Usando estos comandos, una línea de separación se mostrará. Para modificar el espesor de la misma debes utilizar lo siguiente:

```
\renewcommand{\headrulewidth}{0.4pt}
```

donde puedes jugar con la medida entre llaves. Escribiendo `0pt` (cero puntos), la línea desaparece.

13.3 Un ejemplo

A poner las cosas todas juntas:

```
\newfontfamily{\coral}{TeX Gyre Chorus}
\renewcommand{\headrulewidth}{0.4pt}
\fancyhead[R0,LE]{\thepage}
\fancyhead[L0]{\nouppercase{\leftmark}}
\fancyhead[RE]{\coral{El pingüino tolkiano}}
\fancyfoot{}
```

Aquí tienes el número de página a la derecha en las páginas impares y a la izquierda en las pares, el nombre del capítulo a la izquierda en las impares y el texto «El pingüino tolkiano», formateado con la fuente TeX Gyre Chorus, a la derecha de las pares.

Ahora bien, el comando `\leftmark` escribe «Capítulo #. Nombre del capítulo». Si quieres que solo aparezca el nombre del capítulo sin el número, también en el preámbulo tienes que escribir lo siguiente:

```
\renewcommand{\chaptermark}[1]{
\markboth{#1}{}}
```

mientras que si quieres que la sección aparezca sin numerar debes agregar esto:

```
\renewcommand{\sectionmark}[1]{
\markright{#1}}
```

Para que el nombre del capítulo también aparezca en la nueva familia de fuente, solo es necesario rodear `\leftmark` con la correspondiente instrucción. Siguiendo con el ejemplo, sería

```
\fancyhead[L0]{\nouppercase{\coral{\leftmark}}}
```

El manual de `fancyhdr` te dará (¡mucha!) más información, como la forma correcta de incluir una imagen en la cabecera o el pie de la página.

13.4 Crear y utilizar estilos de página

A partir de ahora verás cómo definir y utilizar estilos de página, resolviendo de paso algunos problemas algunas *características no deseadas* de las opciones por defecto en la mayor parte de las clases de documento de libros.

Todo esto será con código \LaTeX puro y duro. Qué se le va a hacer.

Imagina que necesitas crear un documento con cabeceras de página elaboradas y con una estructura como la siguiente:

1. índice con páginas en números romanos (¡en mayúsculas!),
2. un capítulo no numerado que contenga una introducción a la obra,
3. primera parte,
4. primer capítulo numerado,
5. ...

Ya te conté sobre la numeración de las páginas en los índices (capítulo 11), e introduje las herramientas para modificar cabeceras y pies de página más arriba. Para tener los entornos *Parte* y *Capítulo* necesitas una de las clases de documento *Book*. Un capítulo no numerado se crea seleccionando el entorno `Capítulo*` (con el asterisco), por lo que todo parece fácil.

El problema con todo esto, porque si no se presentaran problemas no existiría esta sección, tiene varias partes. Para empezar, la introducción creada con el capítulo no numerado no es incluida en el índice. Para seguir, la

información disponible para la cabecera/pie de página de la introducción mostrará «Índice general» en lugar de «Introducción»: la cabecera de la página no actualiza su información porque al ser el capítulo no numerado (y por lo tanto no *indexado*) no hay nueva información disponible.

Para resolver esto tienes que crear un estilo de página con la información necesaria y decirle a \LaTeX que debe agregar una entrada más al índice.

Para comprender cómo funciona esto, te daré un ejemplo para analizar en detalle. En el preámbulo \LaTeX escribe lo siguiente:

```
% el estilo de página normal, con sus cabeceras
\pagestyle{fancy}
\fancypagestyle{normal}{%
  \fancyhead{} % limpia las cabeceras para dar
                % lugar a lo que sigue
  \fancyhead[RO,LE]{\thepage}
  \fancyhead[LO]{\nouppercase{\leftmark}}
  \fancyhead[RE]{\nouppercase{Nuestro gran
                  libro}}
  \fancyfoot{}
}
% el estilo de página especial
\fancypagestyle{especial}{%
  \fancyhead{} % limpia las cabeceras
  \fancyhead[RO,LE]{\thepage}
  \fancyhead[LO]{\nouppercase{Introducción}}
  \fancyhead[RE]{\nouppercase{Nuestro gran
                  libro}}
  \fancyfoot{}
}
% el estilo de página del índice
\fancypagestyle{index}{%
  \fancyhead{} % limpia las cabeceras
  \fancyhead[RO,LE]{\thepage}
  \fancyhead[LO]{\nouppercase{Índice}}
  \fancyhead[RE]{\nouppercase{Índice}}
  \fancyfoot{}
```

}

Con esto he redefinido el estilo de página normal y he creado dos estilos nuevos llamados «especial» e «index» para utilizar en el capítulo no numerado y en el índice, respectivamente, dándole a todos la información que sirve.

Ahora, al principio del documento, donde va el índice de contenido, inserta una caja \TeX y en ella escribe esto:

```
\frontmatter
\setcounter{page}{1}
\pagestyle{index}
```

(la segunda línea suele no ser necesaria). Con esto le dices a \LaTeX que lo que sigue es el espacio dedicado al índice con el estilo de página que le corresponde y que tiene que numerar las páginas con números romanos.

Si quieres insistir en que estos números romanos sean en mayúsculas podrías agregar lo siguiente:

```
\pagenumbering{Roman}
```

Solo para estar seguros.

Debajo de la caja Insertar Listas/Contenidos/Referencias General y a continuación del mismo inserta otra caja \TeX para en ella escribir lo siguiente:

```
\mainmatter
\setcounter{page}{1}
```

(esta segunda línea tampoco suele ser necesaria). Este código indica que ahora comienza lo bueno, que ya estamos en el argumento principal.

Ahora inserta el entorno «capítulo no numerado» utilizando el menú para elegir entornos y escribe el título, por ejemplo, «Introducción».

¡Utiliza el mismo título definido para la página especial!

En el renglón *inmediatamente debajo de este título* inserta una nueva caja \TeX con el siguiente contenido:

```
\pagestyle{especial}
\addcontentsline{toc}{chapter}{Introducción}
```

La primera línea asigna el estilo de página especial a lo que sigue, mientras que la segunda agrega una línea en el índice de contenido apuntando a esa página, mostrándose en el nivel asignado (`chapter`) y con el texto elegido (Introducción, aquí también hay que mantener la coherencia).

Terminada la escritura de la introducción, es momento de seguir adelante. Luego de aplicar (si sirve) el entorno `Parte` y a continuación el entorno `Capítulo` (*sin* el asterisco), agrega otra caja `TEX` con el siguiente código:

```
\pagestyle{normal}
```

así todo vuelve a la normalidad.

El resto del documento se escribe normalmente, ya no necesitas más cajas `TEX`. Al menos no para estos problemas.

A continuación, algunos enlaces que me han servido para introducirme al tema.

Crear y utilizar estilos de página en \LaTeX :

<http://tex.stackexchange.com/questions/37574/different-headers-for-different-chapters>

Agregar al índice un capítulo no numerado:

<http://wiki.lyx.org/FAQ/TOC#addcontentsline>

13.5 Primera página del capítulo

Por defecto, en las clases de documento de libros la primera página de cada capítulo utiliza el estilo «`plain`». Si quieres aplicar un estilo diferente tienes que agregar, en el preámbulo \LaTeX , algo como lo siguiente:

```
\fancypagestyle{primera}{%
  \fancyhead{}
  \fancyhead[R]{\nouppercase{Nuestro gran li-
bro}}
  \fancyfoot[C]{Página \thepage}
}
```

```
\makeatletter
\let\ps@plain\ps@primera
\makeatother
```

Las primeras líneas definen un estilo de página llamado «primera» mientras que las tres últimas (bueno, solo la central) cambia el estilo de página para la primera página de los capítulos de «plain» al nuevo estilo «primera». Así la primera página de cada capítulo mostrará, a la derecha de la cabecera, el texto «Nuestro gran libro» mientras que en el centro del pie verás «Página» seguido del número de página. De más está decir que te recomiendo no copiar esto tal cual y adaptar el código a tus necesidades.

13.6 Páginas en blanco

En un documento a dos lados con una de las clases *book* seguramente aparecerán páginas en blanco insertadas automáticamente: los capítulos inician en páginas derechas (impares), por lo que si el capítulo anterior terminó también en página impar en medio tendrás una página en blanco.

Por defecto para esas páginas en blanco \LaTeX usa el mismo estilo de todo el capítulo, por lo que la página en blanco no estará *realmente* en blanco.

Para hacer que las páginas en blanco estén realmente en blanco tienes que instalar el paquete `emptypage`. En \TeX Live este paquete se llama:

```
texlive-emptypage
```

Una vez instalado, en el Preámbulo \LaTeX agrega la siguiente línea:

```
\usepackage{emptypage}
```

Ahora bien, ¿qué tal si quieres una página *diferente*, no completamente en blanco, pero tampoco igual a las páginas del cuerpo del capítulo? Podrías necesitar que diga «página dejada intencionalmente en blanco» o algo así.

Si miras la documentación del paquete `emptypage` notarás que lo único que hace es redefinir las páginas en blanco para que utilicen el estilo «empty» por lo que tomando el ejemplo puedes escribir:

```

\fancypagestyle{raro}{%
  \fancyhead{} %
  \fancyhead[LE]{\thepage}
  \fancyhead[RE]{\nouppercase{Página vacía}}
  \fancyfoot[CE]{Página vacía}
}
\def\emptypage@emptypage{%
  \hbox{}%
  \thispagestyle{raro}%
  \newpage%
  \if@twocolumn%
  \hbox{}%
  \newpage%
  \fi%
}
\def\cleardoublepage{%
  \clearpage%
  \if@twoside%
  \ifodd\c@page%
  \else%
  \emptypage@emptypage%
  \fi%
  \fi%
}

```

Te queda como tarea la interpretación del código y su posible modificación para adaptarlo a otros formatos deseados.

13.7 Cambiar el estilo de página del entorno Parte

En un apartado anterior te mostré cómo crear y aplicar un estilo de página para cambiar el contenido de la cabecera o pie de página de un bloque de tu documento. Ahora tenemos un problema similar, pero de solución más compleja: cambiar el estilo de página asociado al entorno Parte.

El entorno Parte, utilizado para agrupar capítulos en las clases de documento libro, tiene asociado un estilo de página particular el cual podría

tener, dependiendo de la clase de documento elegida, un pie de página con el número de página. Digamos que quieres eliminar ese número de página, ¿es posible utilizar lo visto más arriba? Desafortunadamente, no.

El problema es que el entorno `Parte` está también asociado con quiebres de página, por lo que cualquier instrucción del tipo `\thispagestyle` que utilices antes o después del entorno será aplicada a la página anterior o siguiente y no a la que interesa. Tampoco puedes colocar la instrucción `\thispagestyle` dentro del entorno `Parte` ya que esta instrucción no puede estar anidada. ¿Entonces? Pues que hay que parchear el entorno `Parte`.

El siguiente código lo encontré en este tema del foro *L^AT_EX Community*: <http://www.latex-community.org/forum/viewtopic.php?f=47&t=5809> Lo que hace es cambiar la definición del entorno `\part` (`Parte`) asignándole el estilo de página «empty», que no tiene cabeceras ni pie de página.

Para utilizarlo en LyX, en el Preámbulo L^AT_EX escribe:

```
% páginas \part sin pie de página
\makeatletter
\renewcommand\part{%
  \if@openright
    \cleardoublepage
  \else
    \clearpage
  \fi
  \thispagestyle{empty}%
  \if@twocolumn
    \onecolumn
    \@tempwattrue
  \else
    \@tempwafalse
  \fi
  \null\vfil
  \secdef\@part\@spart}
\makeatother
```

Listo. Y sí, L^AT_EX puede ser *muy* extraño.

NOTA: existe una forma más simple de lograrlo, pero que no funciona siempre. Hay que parchear el entorno «part» usando el comando `etoolbox` de la siguiente forma:

```
\usepackage{etoolbox}
\patchcmd{\part}{\thispagestyle{plain}}{
  \thispagestyle{empty}}
  {}{\errmessage{No puedo
                parchear \string\part}}
```

Ahora bien, ¿cómo cambiar el estilo de página que *sigue* al entorno Parte? En este libro tenemos allí diferentes e inspiradas frases...

```
\makeatletter
\def\@endpart{\vfil\newpage
              \if@twoside
              \if@openright
              \null
              \thispagestyle{empty}%
\vspace*{3cm}%
\begin{quote}%
\partnote
\end{quote}%
              \fi
              \fi
              \if@tempwa
              \twocolumn
              \fi}
\newcommand\partnote{}
\makeatother
```

Aquí tienes varias cosas. Por una parte se redefine el «endpart», el final de la parte, por otra se define un entorno de «cita» con todo y espacio inicial y se asigna un estilo de página a todo esto.

Puedes ver cómo creé el estilo de carácter para la frase en los apéndices **B** y **C**. A esta altura no debería resultar difícil el interpretar todos esos códigos.

Capítulo 14

Figuras y otros flotantes

Ya hablamos del concepto de flotantes en la subsección 5.1.7: un objeto dentro del cual se pueden colocar figuras o cuadros y cuya posición será controlada por \LaTeX para colocarlo correctamente cuando el documento venga compilado.

En este capítulo te mostraré algunas de las opciones que existen para modificar el formato tanto de estos flotantes como de su contenido. Si bien por simplicidad de redacción hablaré mayormente de figuras, lo que comentaré aquí sirve en principio para todos los tipos de flotantes, por lo que cada vez que veas «figura» lo que tienes que leer es «figuras/cuadros/etcétera».

14.1 Tamaño de las imágenes insertadas

Cuando insertas una imagen en LyX esta quedará con su tamaño real, por lo que si es grande al compilar el documento te darás cuenta de que ha perforado el margen. De hecho, dependiendo de la resolución de la imagen y de qué tan grande sea más que perforarlo podría *demolerlo*. A continuación te muestro cómo resolver este y otros problemas.

14.1.1 Ajustar el tamaño de la imagen en el documento compilado

Ya sea mientras la insertas o luego al hacer clic sobre la imagen se abrirá el diálogo mostrado en la figura 14.1.

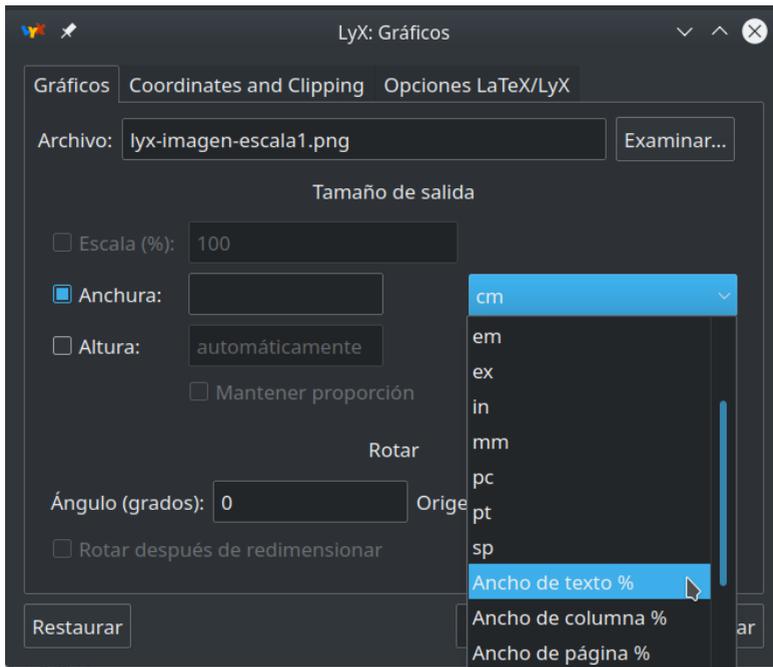


FIGURA I4.1: *modificar el ancho mientras se inserta la figura*

Como puedes ver, en la pestaña GRÁFICOS tienes varias opciones, como dar un porcentaje respecto de la imagen en sí, o fijar la anchura / altura. Para esto último puedes elegir un tamaño absoluto o un porcentaje respecto de la página, de la columna, lo que sirva. La opción seleccionada en la captura permitirá elegir que la imagen cambie su escala al porcentaje seleccionado, por ejemplo, para que ocupe el 80 % del ancho del texto. Podrías elegir también el ancho de la página, o su alto, en fin, que te dejo explorar las opciones.

Eso sí, es importante notar aquí que esto forzará a *cualquier* imagen a ocupar *exactamente* el porcentaje del área de texto elegido: las imágenes más grandes serán automáticamente reducidas, lo cual está muy bien, *pero las más pequeñas serán aumentadas*, lo cual está muy mal, por lo que te recomiendo prestar mucha atención y utilizar esta opción solo cuando sabes que la imagen que estas insertando es demasiado grande para la página de destino.

Desde la versión 2.4, es posible elegir la alineación del *contenido* de los flotantes. Para hacerlo globalmente en el documento es suficiente dirigirse a Documento » Configuración » Configuración del flotante para hacer, por ejemplo, que la imagen quede centrada en el flotante. Es posible modificar esta opción para una imagen particular desde las propiedades del flotante mismo (clic derecho sobre la etiqueta del flotante □ » Configuración).

14.1.2 Tamaño de la imagen en la ventana de edición

Como puedes ver en la figura 14.2, en la pestaña con las opciones L^AT_EX / L_YX puedes elegir cómo se muestra (o si no se debe mostrar) la imagen en la ventana de L_YX durante la edición del documento: mientras trabajas no es necesario ver la imagen en su máxima resolución por lo que podría resultar cómodo reducirla bastante o incluso decidir de no mostrarla.

En la imagen se ve otra cosa, pero eso lo discuto en el siguiente apartado.

14.1.3 Por no repetir: grupos de imágenes

Ahora bien, ¿hay que hacer lo contado en los apartados anteriores *cada vez*?

¡No!

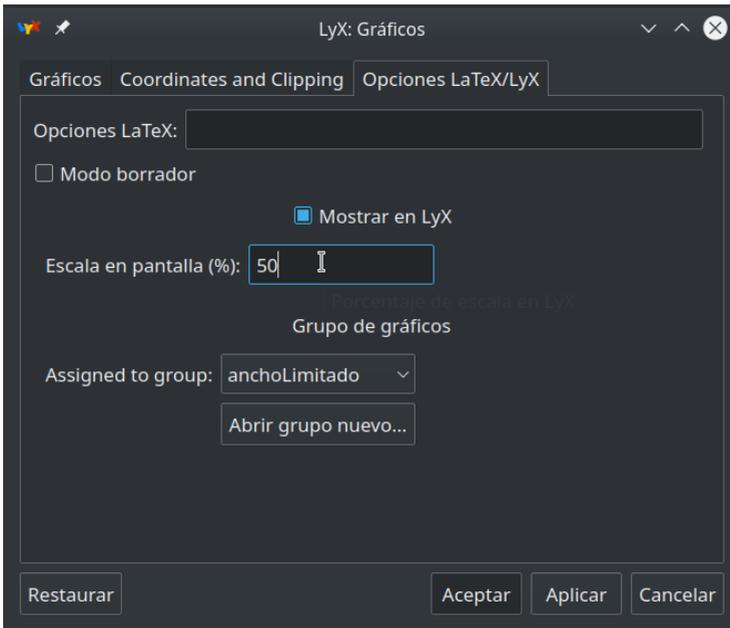


FIGURA 14.2: cómo mostrar las imágenes durante la edición

Como puede verse *también* en la figura 14.2, tienes la opción GRUPO DE GRÁFICOS. Esta opción te permite dar un nombre a una configuración particular para luego llamarla en la próxima figura: con ABRIR GRUPO NUEVO asignas un nombre a la particular configuración que quieres reutilizar mientras que, al insertar la próxima figura, podrás utilizar el menú ASIGNADO AL GRUPO para llamar esa configuración.

Más fácil imposible.

14.2 Figuras envueltas

En esta sección te hablaré de figuras rodeadas de texto, como lo que se muestra aquí mismo.

Me apresuro en advertir que esto es uno de los pocos puntos que \LaTeX no logra manejar del todo bien: el paquete que comento aquí, `wrapfig`, puede resultar un tanto inestable y dar resultados impredecibles. Por ejemplo, con las opciones por defecto si al compilar el documento la figura cae cerca del margen inferior podría perforar el pie de página e incluso dejar un hueco al comienzo de la página siguiente (te dejo hacer unas pruebas, puede ser hasta divertido). Y es que el paquete no verifica si existe lugar en la página para poner la figura allí.

Ya te contaré cómo arreglar eso. Más o menos.

\LaTeX ofrece soporte completo para este paquete, por lo que para utilizarlo solo tienes que ir a `Insertar` \gg `Flotante` \gg `Figura envuelta` (también puedes insertar cuadros envueltos desde allí). Es importante notar que si colocas la figura envuelta entre dos párrafos esta será rodeada por el segundo, mientras que si la colocas entre dos palabras dentro de un párrafo será rodeada por las líneas de texto a partir de aquella en la que se encuentran esas palabras.



FIGURA 14.3: *un bonito pingüino que adopté en internet. Al anónimo autor de este y otros pingüinos, gracias.*

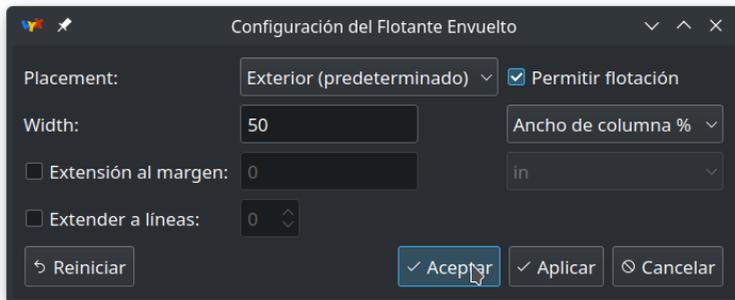


FIGURA 14.4: *la configuración de la figura envuelta*

Esto implica la necesidad de mucho cuidado y control manual de la posición de las figuras envueltas, por lo que surge la pregunta de si es posible hacer que estas floten como las figuras normales. Y además, ¿cómo elegir de qué lado de la página aparecerá la figura o la fracción del ancho de página que esta ocupará?

Haciendo clic derecho sobre la etiqueta del envoltorio puedes acceder al menú de configuración (figura 14.4). Las opciones de ubicación son EXTERIOR (a la derecha en páginas impares y a la izquierda en las pares de los documentos a dos caras), INTERIOR (lo contrario), SIEMPRE IZQUIERDA y SIEMPRE DERECHA. La opción PERMITIR FLOTACIÓN es la que corrige los problemas que comenté anteriormente.

¿Listo? Bueno, no del todo:

- Las figuras envueltas no pueden utilizarse en combinación con listas numeradas o viñetas.
- Si la figura envuelta está en el último párrafo y el texto no es suficiente para envolverla, dependiendo de si se ha activado la flotación o no podrías tener una figura que se desplaza luego del texto y que no respeta el lado elegido (con flotación) o una figura «envuelta a mitad» (sin flotación).
- Si se tienen dos figuras envueltas cercanas, el resultado puede ser malo (nuevamente, puedes hacer la prueba).

- Si bien la numeración se integra con la de las figuras normales, podría darse que no aparezcan en orden, dándonos, por ejemplo, la figura normal de número 5 en el principio de la página y la envuelta de número 4 más abajo.
- ...

En fin, que es un paquete cuyos resultados pueden traer algunos dolores de cabeza, requiriendo mucho control manual del documento antes de poder considerarlo «listo».

¡Estás advertido!

14.3 Configurar las leyendas de figuras y cuadros

La forma más sencilla de configurar la leyenda de figuras y cuadros en un documento \LaTeX es utilizar el paquete `caption`. En Linux y con `TeXLive` (los paquetes son `texlive-caption` y `texlive-caption-doc`) la documentación se instala en

```
/usr/share/texmf/doc/latex/caption/caption-eng.pdf
```

La documentación es muy sencilla por lo que en este breve apartado solo daré un ejemplo cotidiano para ver cómo se utilizan los diferentes parámetros de este paquete.

La idea de este ejemplo es configurar las leyendas de figuras y cuadros para que sean:

- justificadas pero con la última línea centrada,
- la etiqueta se muestre en `VERSALITAS`,
- el texto de la leyenda en *cursiva*.

Para lograr esto, en el preámbulo \LaTeX debes escribir lo siguiente:

```
\usepackage{caption}
\captionsetup{justification=centerlast,%
              labelfont=sc,textfont=it}
```

La primera línea activa el paquete `caption` mientras que la segunda lo configura con las tres opciones comentadas más arriba.

Otras opciones ofrecidas por `caption`: cambiar márgenes, el tamaño o el tipo de la fuente, interlineado, color del texto, modificar o incluso eliminar el nombre de la etiqueta (para que se muestre solo el número), insertar un quiebre entre la etiqueta y el texto de la leyenda, la distancia entre la imagen/cuadro y la leyenda (la opción `skip`), etcétera.

Es interesante notar que la instrucción `\captionsetup` puede utilizarse no solo en el preámbulo: si la colocas en un flotante particular te permitirá configurar *esa* leyenda.

Otro ejemplo. Podrías querer que debajo de cada leyenda se muestre una línea de separación. Para esto puedes utilizar la instrucción `\hrulefill` de la siguiente forma:

```
\usepackage{caption}
\DeclareCaptionFormat{mi formato}{#1#2#3\hrulefill}
\captionsetup[figure]{format=mi formato}
```

Imagina ahora que quieres modificar esta línea para que sea de un color particular y más gruesa:

```
\usepackage{xcolor}
\definecolor{rojito}{rgb}{.8,.2,.1}
\usepackage{caption}
\DeclareCaptionFormat{mi formato}{#1#2#3\color{rojito}
\def\hrulefill{\leavevmode\leaders\hrule
height 2pt\hfill\kern\z@\hrulefill}
\captionsetup[figure]{format=mi formato}
```

Te dejo con los detalles para ver cómo combinar esto con lo anterior.

Existe un último truco para configurar las leyendas de figuras y cuadros: hacer que estas aparezcan *a los lados* en lugar de arriba o abajo. Pero esto requerirá el trabajar con módulos y formato local en gran estilo, por lo que lo dejo para la sección [16.3](#).

14.4 Elegir estilo de página para un flotante particular

Todos apreciamos L^AT_EX por su enorme capacidad a la hora de colocar flotantes en nuestros documentos. Ya sean de figuras o cuadros, los dejas caer en el editor y al compilar el documento irán al mejor lugar posible: si no hay ya otro flotante allí, al principio de la página en el que fueron insertados, si el principio está ocupado al final o a la página siguiente, y si el flotante es muy grande en una página propia, pero siempre cerca del punto de inserción.

Ahora bien, imagina uno de esos flotantes grandes que van a parar en una página propia, ¿existe alguna forma de hacer que la página de *ese* flotante tenga un estilo diferente? Por ejemplo, podrías querer que esa página no muestre la cabecera o el pie de página, solo el flotante. Es decir, lograr que solo la página del flotante en cuestión tenga el estilo de página `empty`.

Pues resulta que es muy fácil. Primero tienes que asegurarte de tener instalado el conjunto de paquetes `sttools`, que provee el paquete `floatpag`. Instalado, debes recordarte de reconfigurar L^AT_EX. Finalmente, en el preámbulo del documento escribe lo siguiente:

```
\usepackage{floatpag}
```

mientras que en el flotante en sí mismo agrega una caja T_EX con esta instrucción:

```
\thisfloatpagestyle{empty}
```

Si quieres otro estilo de página, en lugar de `empty` usas el que quieras y todo listo, que esas cosas ya te las expliqué en el capítulo 13.

14.5 Renombrar flotantes

L^AT_EX ofrece por defecto dos clases de flotantes: `FIGURE` y `TABLE`. L^AT_EX agrega un tercero: `ALGORITHM`.

Ahora bien, por defecto la traducción de *Figure* es Figura, la de *Table* es Cuadro y la de *Algorithm* es Algoritmo. ¿Se pueden cambiar esos nombres? No estaría mal el poder utilizar «Diagrama» en lugar de «Figura» o «Guion» en lugar de «Algoritmo».

Pero atención, que cambiar el nombre a la etiqueta no alcanza: los flotantes pueden tener índices propios que deben ser cambiados también para evitar la confusión de los lectores.

Para esto puedes utilizar instrucciones de Babel para simular una traducción de las etiquetas. Dado que Babel es ya llamado por \LaTeX no tienes que llamarlo otra vez, solo hay que configurarlo de forma apropiada, por lo que al preámbulo vamos:

```
\AtBeginDocument{
  \addto\captionsspanish{
    \def\figurename{Diagrama}
    \def\listfigurename{Índice de
      diagramas}
  }}

```

cambiará Figura por Diagrama. Para el nombre de los cuadros hay que usar $\text{\table}\{\text{nuevo-nombre}\}$ para la etiqueta y $\text{\listtable}\{\text{Índice de nuevo-nombre(s)}\}$ para el índice, etcétera.

Para una lista de los nombres que se pueden modificar, consulta:

<https://texfaq.org/FAQ-fixnam>

14.6 Cambiar el tipo de numeración de figuras y cuadros

Esta es fácil, de hecho ya te mostré algo parecido en el capítulo 12. Para numerar las figuras con números romanos (¡en mayúscula!), en el preámbulo \LaTeX escribe

```
\renewcommand{\thefigure}{\Roman{figure}}
```

Para los cuadros simplemente cambia ambas instancias de `figure` por `table`. Y para cambiar números romanos por letras, bueno, ya sabes: capítulo 6.

Capítulo 15

Letras capitales y afines

 EGURAMENTE habrás encontrado documentos donde sucede lo que se ve en este párrafo particular: la primera letra de la primera palabra en la primera línea se extiende por varios renglones, mostrando quizás (como aquí) una fuente especial y adornada, con el resto de esa primera palabra escrito en versalitas.

Este estilo es lo que se conoce como «letra capitular», «letra capital» o simplemente «inicial». Para hacer esto en \LaTeX /LyX es necesario utilizar el paquete `lettrine`.

15.1 Algo tradicional

 N LyX se tiene soporte para este paquete desde la interfaz gráfica y en la sección 6.3 del manual que encuentras en el menú `Ayuda >> Objetos insertados` se explica claramente una de las formas de utilizarlo. Porque como te mostraré a continuación existe *otra* forma, la cual resulta mucho más simple. Y es que lo que se cuenta en el manual te ofrece toda la flexibilidad que quieras a la hora de elegir la forma de la letra capital cada vez que la necesites, ¿pero qué tal si quieres utilizar letras capitales varias veces (por ejemplo, en el primer párrafo de cada capítulo) y que siempre se vean igual? Hacerlo de la forma indicada en el manual de LyX podría volverse un tanto incómodo por lo que resulta conveniente encontrar un modo de automatizar el proceso.

Para empezar y como siempre, debes estar seguro de que el paquete a

utilizar esté correctamente instalado en \LaTeX y de no ser así instalarlo y luego reconfigurar \LyX para que lo reconozca. La documentación del paquete `Lettrine` en Linux se instala en

```
/usr/share/texmf/doc/latex/lettrine/lettrine.pdf
/usr/share/texmf/doc/latex/lettrine/demo.pdf
```

Ahora hay que dirigirse a `Documento` \gg `Configuración` \gg `Módulos` (ya te hablé de los módulos en el capítulo 9) para habilitar el módulo `CAPITALES`.

¡Clic en `APLICAR`, NO `ACEPTAR`, que aún queda trabajo por hacer en este menú! Y es que, como suele suceder en estos casos, hay que ir al Preámbulo \LaTeX para agregar algo de código.

Este documento está configurado para utilizar $X\TeX$ con la fuente `EB Garamond` como fuente de base y define un alias de la fuente `Pali`¹ utilizando las alternativas estilísticas `(+salt)` para que sirva de letra capital.

Esto es lo que he escrito en el preámbulo:

```
\newfontfamily{\Iniciales}[
  RawFeature={+salt}]{Pali}
\renewcommand{\LettrineFontHook}{\Iniciales}
\setcounter{DefaultLines}{4}
```

La primera instrucción define una nueva familia de fuente (ya te hablé sobre esto en la sección 7.3), definición que es utilizada en la segunda instrucción para decir al paquete `Lettrine` cuál fuente debe utilizar para las letras capitales. La tercera instrucción dice cuántas líneas debe ocupar la letra capital.

Es posible controlar la posición vertical del carácter agregando una cuarta línea

```
\renewcommand{\DefaultLraise}{0.35}
```

donde el valor `0.35` puede modificarse a aquello que sirva. Generalmente no es necesario utilizar esta instrucción.

Hecho esto, para insertar una nueva letra capital selecciona el primer carácter del párrafo en cuestión y `clic derecho` \gg `Estilo de texto personalizado` \gg `Capital`, mientras que para tener el resto de la primera palabra en versalitas

1. <http://www.softviews.org/Fonts.html#Pali>

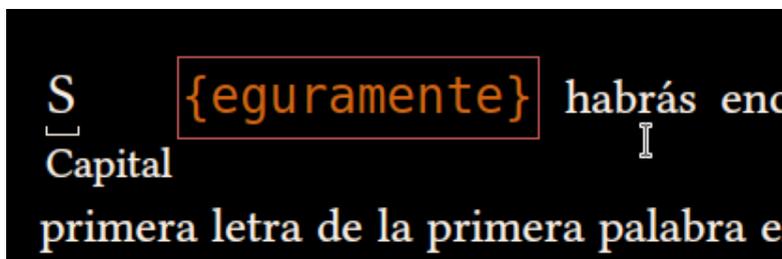


FIGURA 15.1: *letra capital en la interfaz gráfica de LyX*

debes encerrarla entre llaves y meter todo (resto de palabra y llaves) en una caja \TeX (se selecciona el texto y se presiona Ctrl L). En la interfaz gráfica de LyX esto se verá como lo que muestra la figura 15.1.

La documentación del paquete `Lettrine` es breve por lo que no tendrás problemas en encontrar otras opciones para ajustar todo a tus preferencias en forma rápida y simple.

15.2 ¡Atención al número de líneas!

RECISAMENTE, debes estar atento. Y es que algo muy importante a tener en cuenta es que si el párrafo sobre el cual activas la letra capital es breve (por ejemplo, una o dos líneas) y la inicial es grande (por ejemplo, cuatro) *la letra inicial se superpondrá al párrafo siguiente.*

Algo que, ciertamente, no ha sucedido aquí. ¿El truco? Pues unir el párrafo con la inicial al siguiente simulando el quiebre de párrafo con un quiebre de línea.

En \LaTeX los quiebres de línea se introducen con dos barras invertidas consecutivas `<<\\>>` (sin las comillas), pero en LyX es mucho más simple: presionas Ctrl ↵ . En la ventana de edición se verá una flecha quebrada roja.

Al ser todo parte de un mismo párrafo ya no se produce la molesta superposición, *pero el segundo párrafo no muestra sangría.* Algo que aquí sí se ve. ¿Truco? En este punto se presenta un pequeño problema, porque los comandos normales de \LaTeX para agregar espacios horizontales no fun-

cionan al principio de una línea, ¿entonces? Insertar Formato Fantasma Fantasma horizontal y en la caja que se presenta escribes, por ejemplo, dos «m» las cuales se mostrarán en la ventana de edición tachadas y entre paréntesis angulares mientras que en el resultado te darán lo que se muestra al inicio de esta subsección.

15.3 Una variante: letra capital «alzada»



¿QUÉ el truco está en una pequeña variante del método presentado anteriormente.

En el preámbulo \LaTeX (por supuesto, luego de activar el módulo Capitales y todo lo que ya conté antes) tienes que escribir lo siguiente:

```
\newfontfamily{\Iniciales}[
  Scale=2,RawFeature={+salt}]{Pali}
\renewcommand{\LettrineFontHook}{\Iniciales}
\setcounter{DefaultLines}{1}
```

Como puedes ver se está definiendo todo casi de la misma forma que en la sección anterior, pero agregando una instrucción para hacer que la fuente definida en la nueva familia `\Iniciales` se muestre al doble de su tamaño (claramente puedes siempre jugar con el parámetro de la instrucción `Scale`) mientras que el número de líneas lo he cambiado a 1.

Ya con esto es suficiente para lograr que la mayúscula inicial no baje sino que se muestre, más grande, en la misma línea del texto.

15.4 Un desafío



JUSTAMENTE en este punto te dejo con un desafío: combinar las secciones previas para lograr una inicial que tome varias

líneas (digamos, dos), pero que también suba lo suficiente como para lograr lo que se ve en este párrafo.

Vamos, que con lo visto debería ser sencillo.

Y no, no verás la solución en el apéndice con el código del preámbulo \LaTeX de este documento ya que esta letra inicial la he creado en el lugar utilizando el entorno Capital con opciones introducidas a mano *solo para ese párrafo*.

¡No es posible hacer trampas!

15.5 Una alternativa: primera línea de un párrafo en versalitas

UNA ALTERNATIVA A LAS LETRAS CAPITULARES ES QUE LA PRIMERA línea del primer párrafo se encuentre toda ella en versalitas, algo que puede verse principalmente en revistas (además de en este párrafo).

El único modo de obtener este efecto en forma (semi) automática con \LaTeX que he encontrado es con el paquete `magaz`.

La documentación de este paquete se instala, en Linux y con \TeX Live, en

```
/usr/share/texmf/doc/latex/magaz/magaz.pdf
```

Este paquete es un tanto caprichoso ya que no solo se lleva mal con el paquete `lettrine`, tampoco acepta ecuaciones, formatos particulares o tan siquiera eliminar la sangría de primera línea que introduce. Pero en su forma más simple es muy fácil de utilizar.

En el preámbulo \LaTeX escribe

```
\usepackage{magaz}
```

y luego, en el párrafo que quieres modificar, abre dos cajas \TeX , una al comienzo del párrafo con la instrucción

```
\FirstLine{
```

y la otra más adelante (no necesariamente al final del párrafo, solo debe tomarse suficiente texto como para cubrir la primera línea) con un

}

para cerrar la instrucción.

El paquete cambiará automáticamente el formato de la primera línea sin más trabajo por parte del escritor, cuidando incluso (si bien no siempre: ¡hay que controlar el resultado!) los casos en los que la última palabra de la primera línea se quiebre, como sucede aquí.

Es importante notar que la instrucción `FirstLine` debe estar al comienzo del párrafo o de lo contrario te dará error.

Este paquete ofrece más opciones. Por ejemplo es posible seleccionar otro formato para la fuente utilizada en la primera línea con la instrucción

```
\renewcommand\FirstLineFont{\<instrucción de
                             formato>}
```

pero como la breve documentación del paquete es lo suficientemente clara no entraré en esos detalles.

Capítulo 16

Extras

En este capítulo te contaré algunos detalles quizás demasiado particulares. Por una parte, te encontrarás con trucos que he utilizado en este documento, como por ejemplo insertar las direcciones de los menús de una forma que parezcan menús o dar atajos de teclado que parezcan teclas (te doy dos métodos para esto).

Por otra parte también verás algunas cosas que no he hecho para este libro, como agregar más flotantes, usar leyendas laterales para figuras y cuadros, cómo transformar un módulo en formato local o incluso cómo rodear un párrafo de una «caja coloreada».

Por último te daré una breve visión de la enorme variedad de cosas que no he llegado a tratar en este libro.

16.1 Describir teclas y menús: el paquete **menukeys**

A lo largo de este libro he mostrado muchas veces combinaciones de teclas y menús, algo como decir **Ctrl**  o bien  
.

No existe un modo oficial de obtener estos efectos, por lo que te mostraré un método para hacer ambas cosas con el paquete `menukeys`. En la próxima sección se mostraré otro método para realizar la primera.

El manual de `menukeys` generalmente se instala en:

`/usr/share/texmf/doc/latex/menukeys/menukeys.pdf`

Es muy completo, pero supone un mínimo conocimiento por parte del lector por lo que si bien todas las opciones están claramente escritas, no siempre se dice tan claramente cómo utilizarlas. Pero vayamos por partes.

16.1.1 Qué es **menukeys**

Este paquete ofrece tres macros:

- una macro para mostrar atajos de teclado en forma de teclas,
- una macro para mostrar menús en forma destacada,
- una macro para mostrar direcciones de archivos en forma destacada.

La forma más simple de lograr que este paquete funcione es activar en LyX el uso de fuentes no TEX . En realidad podrías prescindir de esto e ir en modo $\text{L}\text{A}\text{T}\text{E}\text{X}$ puro y duro cambiando la codificación del documento, pero no entraré en esos berenjenales.

Luego de comprobar que el paquete está instalado en el sistema y de pedirle a LyX que note que el mismo está disponible (¡siempre hay que reconfigurar!), tienes que ir al preámbulo $\text{L}\text{A}\text{T}\text{E}\text{X}$ para escribir lo siguiente:

```
\usepackage{menukeys}
```

lo cual activará menukeys... con sus valores por defecto, los cuales incluyen utilizar las teclas del teclado Mac en lugar de las de un teclado tipo PC. Es decir, a menos que uses el sistema de la manzanita teclas como Intro, Tab, bloqueo mayúsculas y otras se verán bien extrañas para ti. Para utilizar las teclas tipo PC tienes que pasar un parámetro al paquete para decirle que el sistema operativo que te interesa es (¡ouch!) win, modificando la instrucción mostrada más arriba por:

```
\usepackage[os=win]{menukeys}
```

menukeys ofrece varias formas de teclas y presentaciones de menú entre las cuales puedes elegir. Por ejemplo, si quieres utilizar las teclas sombreadas y que los menús se vean más redondeados tienes que agregar dos líneas en el preámbulo $\text{L}\text{A}\text{T}\text{E}\text{X}$ para dejarlo así:

```
\usepackage[os=win]{menukeys}
\renewmenumacro{\keys}{shadowedroundedkeys}
\renewmenumacro{\menu}{roundedmenus}
```

Los estilos disponibles se encuentran en la sección 4.2 del manual del paquete.

16.1.2 Usar **menukeys** en L^AT_EX

Existen dos formas de utilizar este paquete en L^AT_EX: recurriendo a una caja T_EX o con formato local (sección 9.3).

Antes de pasar a la segunda (y más cómoda) de las dos opciones conven-
drá sin embargo conocer la primera, ya que así sabrás qué código hace falta
escribir.

Pues bien, dentro de una caja T_EX escribe, para un directorio:

```
\directory{/usr/share/texmf/doc/latex/
menukeys/menukeys.pdf}
```

mientras que para un menú tienes dos opciones, utilizar el separador por
defecto:

```
\menu{Documento > Configuración > Preámbulo
LaTeX}
```

o indicar explícitamente *cada vez* cuál separador quieres

```
\menu[,]{Documento , Configuración , Preámbulo
LaTeX}
```

Es importante notar que no se producirá la separación silábica de pala-
bras marcadas por este paquete. Incluso si tienes una frase como uno de los
niveles de un menú esta quedará completa, sin quebrarse, por lo que podría
darse que un menú particularmente largo perfore el margen derecho. Al uti-
lizar este paquete es necesario siempre comprobar la salida de la compilación
para no encontrarse con desagradables sorpresas.

Para representar una tecla tienes que escribir lo siguiente:

```
\keys{\tab}
```

que te dará ... que no es el  visto hasta aquí, pero ya volveré sobre esto en la próxima sección.

En la sección 4.5 del manual de menukeys encontrarás la lista completa de teclas disponibles. Es importante notar que si se escribe una secuencia de letras sin la barra inicial, por ejemplo,

```
\keys{Meta}
```

esta secuencia será tomada como una única tecla . Para más teclas, más instrucciones `\keys` o, si quieres un separador, una sola con el signo «+». Por ejemplo,

```
\keys{\ctrl}\keys{N}
```

te dará  , mientras que

```
\keys{\ctrl+N}
```

te dará  + .

Pues bien, ahora que ya sabes cómo funciona la cosa simplemente debes dirigirte a    y escribir, para obtener un atajo para los menús, lo siguiente:

```
Format 95
  InsetLayout Flex:MenuKeys
  LyxType      charstyle
  LabelString  menukey
  LatexType    command
  LatexName    menu
  Preamble
    \usepackage[os=win]{menukeys}
  EndPreamble
  ResetsFont true
End
```

Ahora en el documento solo tienes que escribir la ruta del menú separando cada paso con un «>», seleccionar todo y con un `⏏` `Estilo de texto personalizado` seleccionar MenuKey. De esta forma aquello que en la ventana de edición saldrá así:

texto Documento > Configuración > Preámbulo LaTeX más texto
menukey

en la salida se verá así:

texto `Documento` `Configuración` `Preámbulo LaTeX` más texto

Te dejo como ejercicio el imaginar cómo se crea un formato local para los atajos de teclado, que después de todo no es lo que he hecho en este libro.

Y hasta aquí llegaremos. El paquete ofrece muchísimas opciones más incluyendo la posibilidad de definir estilos propios, pero no entraré en estos detalles: con lo dicho el resto de la documentación de menukeys debería resultar más accesible... por no decir que las opciones por defecto son de mi agrado y no me interesa cambiarlas.

16.2 Otra forma de representar teclas

Lo comentado en la sección anterior para representar teclas tiene un pequeño problema: no funciona en los títulos. Es decir, si quieres representar un atajo de teclado en un título de sección utilizando menukeys, no puedes, el documento no compila.

La solución que utilizo en este libro para representar teclas es crear un estilo de carácter que aplique la fuente Libertinus Keyboard, la cual ofrece varias ligaduras que cambian el nombre de una tecla a la representación de la misma.

Configurado el documento para utilizar las tipografías no T_EX y teniendo instaladas las fuentes Libertinus¹, en el preámbulo necesitas crear una familia de fuentes con el siguiente código:

```
\newfontfamily{\libertinusKB}{Libertinus
Keyboard}
```

mientras que en el formato local necesitas escribir algo así:

1. <https://github.com/alquerque/libertinus>

```

Format 95
  InsetLayout Flex:LibKB
  LyxType      charstyle
  LabelString  LibKB
  LatexType    command
  LatexName    libkb
  Preamble
    \newcommand*{\libkb}[1]{\{\libertinusKB #1\}}
  EndPreamble
  ResetsFont  true
  Font
    Family     typewriter
  EndFont
End

```

Listo, ya tienes el estilo en el clic derecho Estilo del texto personalizado LibKB. El texto Ctrl te dará Ctrl, Alt te dará Alt, AltGr AltGr, Space , Enter ↵, Esc Esc, Back ←, Del Del, como así también puedes escribir caracteres separados como A, b, →, etcétera.

16.3 Leyendas laterales para figuras/cuadros

Ya vimos en la sección 9.1 cómo agregar módulos a LyX, ahora te mostraré cómo utilizar uno en particular que resulta bastante interesante: `sidecap`.

Este módulo (en realidad, dos módulos) agrega más flotantes que lo que hacen es colocar las leyendas a un costado en lugar de arriba o abajo.

El módulo `sidecap` utiliza el paquete \LaTeX homónimo para realizar esta tarea, pero simplifica el trabajo integrándolo a la interfaz gráfica de LyX.

Desde esta página:

<http://theorie.physik.uni-konstanz.de/jrioux/lyx/>

puedes descargar al menos el primero de los siguientes archivos (¡debes elegir la versión para LyX 2!):

- `sidecap.module`: es la versión principal. Genera flotantes cuya numeración se integra perfectamente a la de las figuras/cuadros norma-

les y cuyo formato respeta lo configurado para estas últimas.

- `sidecap-starred.module`: como el anterior, coloca la leyenda al costado de la imagen, pero elimina la numeración.

Instalado el módulo y reconfigurado LyX puedes dirigirte ahora al menú `Documento` \gg `Configuración` \gg `Módulos` para activar el módulo `Sidecap`. Luego solo tendrás que ir a `Insertar` \gg `Flotante` \gg `...` para insertar las versiones marcadas como *side caption* que te sirvan.

Ahora bien, este módulo utiliza las opciones por defecto del paquete `sidecap`, las cuales consisten en colocar la leyenda en la parte externa de la página (a la derecha en páginas impares y a la izquierda en las pares para documentos a dos lados) con la leyenda alineada a la parte inferior de la figura. Cambiar esto último es muy sencillo. En el preámbulo $\text{L}\text{A}\text{T}\text{E}\text{X}$ puedes escribir lo siguiente:

```
\sidecaptionvpos{figure}{c}
```

para hacer que la leyenda esté verticalmente centrada. Si cambias la `c` por una `t` la leyenda estará alineada a la parte superior de la imagen. Cambiando `figure` por `table` podrás configurar la alineación vertical de las leyendas de los cuadros.

Desafortunadamente cambiar el lado en el que se coloca la leyenda no es tan simple. Si la opción por defecto es apropiada este módulo es muy útil, pero si quieres otra cosa (por ejemplo, colocar la leyenda en la parte interna de la página, o siempre a la derecha/izquierda) lo que puedes hacer es abrir el archivo del módulo en un editor de texto, copiar su contenido desde la instrucción `Format` hasta el final y pegarlo como un nuevo formato local (sección 9.3). Puedes modificar ahora la línea

```
\usepackage{sidecap}
```

para que quede como

```
\usepackage[innercaption]{sidecap}
```

(o bien `leftcaption` o `rightcaption`, lo que quieras). Ahora solo te queda hacer clic en el botón CONVERTIR AL NUEVO FORMATO (el módulo es un tanto viejo), luego en VALIDAR y finalmente en ACEPTAR.

¡Listo! No solo te he mostrado cómo utilizar un módulo externo, también sabes ahora cómo convertirlo en un formato local y cambiarlo según tus necesidades. Incluso no te debería resultar difícil el utilizar todo esto para *crear un módulo nuevo*.

¡Nada mal para una sección *extra*!

16.4 Agregar un nuevo flotante

Un ejemplo sin explicación sobre cómo agregar una característica en LyX: un nuevo flotante utilizando el paquete `photo`. En el formato local, tienes que escribir lo siguiente:

```
Format 95
AddToPreamble
  \usepackage[]{photo}
  \usepackage[spanish]{babel}
  \addto\captionsspanish{
  \def\photoname{Fotografía}
  \def\listphotoname{Índice de fotografías}
  }
EndPreamble
Float
Type photo
GuiName "Photo"
Placement tbp
Extension loscf
NumberWithin none
Style plain
ListCommand listofphotos
ListName "Índice de fotografías"
RefPrefix Photo
IsPredefined 1
UsesFloatPkg 0
```

End

Ahora puedes ir a [Insertar](#) [Flotante](#). También mira los índices. Disfruta.

16.5 Cajas coloreadas

Imagina que quieres obtener el siguiente efecto en tu documento:



Es decir, una caja con bordes coloreados, posiblemente con un título y otras clases de pirotecnias. Para esto y mucho más existe el paquete \LaTeX `tcolorbox`.

\LaTeX ofrece soporte gráfico parcial para este paquete. Para utilizarlo hay que dirigirse a [Documento](#) [Configuración](#) [Módulos](#) para activar el módulo **MARCOS DE COLOR ELABORADOS**. Ahora simplemente te diriges a [Insertar](#) [Recuadro personalizado](#) (sin el módulo del cual estamos hablando aquí, esa entrada del menú estaría vacía) para elegir una de las opciones disponibles.

MARCO COLOR da una caja que toma todo el ancho del área de texto, **MARCO COLOR DINÁMICO** da una caja que se adapta al contenido, etcétera.

Elegida por ejemplo la opción **MARCO COLOR** te encontrarás en la ventana de edición de \LaTeX con una de esas cajas como las de las notas al pie de página: una etiqueta a la izquierda diciendo qué tipo de caja es y un espacio a la derecha donde escribir.

Colocando algo de contenido en la caja y compilando el documento encontrarás en la salida el texto en una caja gris y sin título, digamos, algo aburrido. Y es que debes pasar al paquete los parámetros necesarios para obtener lo que buscas.

Con el cursor dentro de la caja, ya sea mediante el uso del menú [Insertar](#) [Opciones marco color](#) o a través del [clic derecho](#) [Opciones marco color](#) se abrirá una caja dentro de la caja donde pasar los parámetros necesarios. Por ejemplo, para lograr la caja que inicia este apartado he escrito

```
colback=red!5!white,colframe=red!75!black,title=Este
```

es el título de la caja coloreada

¿Que qué significan todos esos códigos? Bueno, la documentación del paquete `tcolorbox`, la cual suele instalarse en Linux en

`/usr/share/texmf/doc/latex/tcolorbox/tcolorbox.pdf`

ocupa 530 páginas, por lo que queda completamente fuera del alcance de esta breve sección el comentar todas las opciones disponibles. El paquete ofrece macros para crear índices, teoremas, cuadros explicativos... para ejemplos mostrando el poder de este paquete es posible consultar el siguiente documento:

`/usr/share/texmf/doc/latex/tcolorbox/tcolorbox-example.pdf`

También es posible consultar el manual de este módulo que acompaña $\text{L}^{\text{X}}\text{Y}$ y que se encuentra en [Ayuda](#) \gg [Manuales específicos](#) \gg [Marcos de color](#).

Ahora que sabes que la herramienta existe, tienes material para entretenerte.

16.6 Cosas de las que no he hablado

$\text{L}^{\text{X}}\text{Y}$ ofrece una lista inmensa de posibilidades para aprovechar la lista inconmensurable de habilidades de $\text{L}^{\text{A}}\text{T}_{\text{E}}\text{X}$ y compañía. En este apartado daré una lista superficial de alguno de los temas que, a pesar de ser interesantes, he decidido no tocar para evitar que el libro crezca desmesuradamente.

16.6.1 Soporte $\text{BIBL}^{\text{A}}\text{T}_{\text{E}}\text{X}$

Como ya te comenté en la sección 5.3, $\text{L}^{\text{X}}\text{Y}$ ofrece soporte para $\text{BIBL}^{\text{A}}\text{T}_{\text{E}}\text{X}$.

Este tema es de interés principalmente para académicos y requiere cierto trabajo para comprender cómo funciona. Ya sabes, internet es tu amigo.

16.6.2 Listado de código con **listings**

$\text{L}^{\text{X}}\text{Y}$ ofrece soporte desde su interfaz gráfica para el paquete `listings` que permite dar formato a código en los documentos.

Con [Insertar](#) \gg [Listado de código](#) obtienes una caja que te permite escribir el código que quieras y con un clic derecho sobre esa caja podrás seleccionar rápidamente el lenguaje de programación, su dialecto, la fuente, la

longitud de las tabulaciones, etcétera. En ese menú también podrás pasar parámetros al paquete para dar color a las instrucciones y todo lo que sirva.

En el capítulo 8 del manual que se abre desde **Ayuda** **Objetos insertados** hay más información al respecto.

16.6.3 Formularios PDF

Pues resulta que también puedes utilizar LyX para crear documentos para la oficina. Bajo **Ayuda** **Manuales específicos** **Formularios PDF** encontrarás una descripción detallada de esta herramienta.

16.6.4 Sweave y knitr

En **Ayuda** **Manuales específicos** **Sweave / knitr** encontrarás dos breves manuales que explican cómo utilizar estos interesantes paquetes \LaTeX que permiten trabajar con el programa estadístico R desde LyX.

16.6.5 Sistemas de álgebra computacional

Si haces un clic derecho dentro del editor de ecuaciones de LyX verás la opción USAR SISTEMA DE ÁLGEBRA COMPUTACIONAL (CAS) y allí varias posibilidades como Octave, Maxima, etcétera, para evaluar rápidamente la expresión insertada. Por ejemplo, seleccionando Maxima en la expresión $\int_a^b x^2 dx$ obtendrás $\int_a^b x^2 dx = \frac{b^3}{3} - \frac{a^3}{3}$.

Por supuesto, Maxima tiene que estar instalado en el sistema.

16.6.6 Y mucho, mucho más

Mejor me detengo aquí, que agregar una lista de las cosas de las que no he hablado *también* implicaría extenderme demasiado.

Crear diagramas de Feynman, escribir partituras musicales con LilyPond, lingüística, presentaciones Beamer, el paquete Xy-pic² para realizar todo tipo de gráficos desde \LaTeX , incluso escritura Braille encuentran soporte directo en la interfaz gráfica de LyX.

Y si crees que lo que te he presentado en esta tercera parte del libro ha sido un mazazo de código \LaTeX , te diré que no te he mostrado más que

2. <https://ctan.org/pkg/xy-pic>

una pequeña fracción de todo lo que es posible hacer: cosas como generar párrafos con formas arbitrarias o incluso utilizar \LaTeX para realizar cálculos matemáticos está al alcance del programador entusiasmado. ¡No hay que olvidar que \TeX es un lenguaje de programación *Turing completo*!

Nadie ha dicho que todas esas cosas sean fáciles y en la gran mayoría de los casos ni siquiera valen la pena el trabajo extra. Por ejemplo, controlar las formas de los párrafos implica muchísimo trabajo manual, pero ciertamente es posible.

Mi intención con este libro era abrir un camino. Está en ti, lector, decidir hasta dónde quieres llegar.

Como dicen por allí, *happy \LaTeX ing!*

Parte IV

Apéndices

Donde se presentan detalles varios sobre el formato de este libro

Las personas grandes son bien extrañas

Apéndice A

Sobre fuentes tipográficas

Elegir la fuente tipográfica correcta para un texto es uno de esos temas en los que se tienen más respuestas que gente respondiendo.

Este no es un libro sobre diseño, sino una explicación sobre cómo utilizar LyX para diseñar. No tengo formación en diseño gráfico por lo que no esperes un consejo. Pero puedo, eso sí, comentar qué utilizo.

A.1 Las fuentes utilizadas en este libro

Como ya comenté antes la fuente del cuerpo de texto es EB Garamond:
<http://www.georgduffner.at/ebgaramond/index.html>

Si bien para esta edición he utilizado la versión de Octavio Pardo, que ofrece varias ventajas (por ejemplo, negrita) junto a alguna desventaja (no tiene tamaños ópticos):

<https://github.com/octaviopardo/EBGaramond12>

Magnífica recreación de los diseños de Claude Garamond de mediados del siglo XVI, esta fuente tiene (al menos a mis ojos) una belleza inigualable. Para tener la *Q de cola larga* he activado la opción OpenType +s06. He superado la falta de tamaños ópticos con lo comentado en la sección 7.6.

Para los títulos también he utilizado otra garamond, Cormorant:

<https://github.com/CatharsisFonts/Cormorant>

Esta fuente es demasiado detallada y con contraste demasiado alto como para poder ser utilizada en el cuerpo del texto, pero para tamaños grandes es simplemente perfecta.

Combinar tipografías en forma armoniosa es un difícil arte. Algunos proyectos nos facilitan las cosas con *familias* (Libertinus, Source pro, PT, Alegreya, etcétera), ¿pero qué fuente *sans* existe que se lleve bien con una garamond? De la gente que produce Cormorant llega una interesante propuesta llamada Ysabeau:

<https://github.com/CatharsisFonts/Ysabeau>

Esta fuente está en plena fase de desarrollo, pero no he tenido problemas utilizándola en este texto.

No he encontrado una fuente monoespaciada pensada especialmente para acompañar una garamond, por lo que al final me he decantado por utilizar un clásico, DejaVu Sans Mono:

<https://dejavu-fonts.github.io/>

Para las fórmulas matemáticas he seleccionado Garamond Math, fuente de la que ya hablé en el capítulo 7.

En los ejemplos durante el desarrollo del libro he nombrado otras fuentes tipográficas, muchas veces indicando dónde encontrarlas.

A.2 Más sobre fuentes tipográficas

Para más información sobre fuentes tipográficas con licencia libre, incluyendo análisis (más o menos) detallados de sus características avanzadas, puedes darte una vuelta por la categoría «tipos gráficos» de mi blog:

<https://elpinguinotolkiano.wordpress.com/category/software/tipos-graficos/>

En particular quiero destacar dos artículos cuyos objetivos resultan demasiado generales como para incluirlos en este libro:

Conceptos tipográficos:

<https://elpinguinotolkiano.wordpress.com/2016/04/28/conceptos-tipograficos/>

Conocer las características OpenType ofrecidas por una fuente:

<https://elpinguinotolkiano.wordpress.com/2017/04/20/conocer-las-caracteristicas-opentype-ofrecidas-por-una-fuente>

Apéndice B

Formato local

Para desarrollar este texto he creado varios estilos de carácter que simplificaran el proceso de edición. A continuación, el contenido de `Documento` `Configuración` `Formato local`.

Pero antes, es importante notar que algunos de los formatos aquí definidos dependen de instrucciones ya existentes en el preámbulo, el cual detallaré en el apéndice C. Y por supuesto, que en un documento normal no es necesario definir *tantos* estilos: las características del presente documento (ser él mismo un ejemplo de los temas que en él se tratan) hacen que sea necesaria una lista semejante. De hecho, el estilo `textfrac` creo haberlo utilizado solo en dos ocasiones... ¿o fue una? ¡Maldición!

¡No exageres con los estilos! El formato te ayuda a expresar mejor tus ideas, pero solo hasta cierto límite: si la presentación del documento es demasiado variada terminará por distraer al lector de aquello que quieres transmitirle. Recuerda: complejidad no es sinónimo de elegancia.

Format 95

```
InsetLayout Flex:DiscrLig
LyxType charstyle
LabelString discrlig
LatexType command
LatexName discrlig
Preamble
\newcom-
```

```
mand*{\discrlig}[1]{\addfontfeature{%
    RawFeature={+dlig}} #1}}
EndPreamble
ResetsFont          true
End
InsetLayout Flex:TextFrac
LyxType             charstyle
LabelString         textfrac
LatexType           command
LatexName           textfrac
Preamble
    \newcommand*{\textfrac}[1]{\addfontfeature{%
        RawFeature={+frac}} #1}}
EndPreamble
ResetsFont          true
End
InsetLayout Flex:Kleymiska
LyxType             charstyle
LabelString         kley
LatexType           command
LatexName           kley
Preamble
    \newfontfamily{\kleym}[] {Kleymiska}
    \newcommand*{\kley}[1]{\kleym #1}}
EndPreamble
ResetsFont          true
End
InsetLayout Flex:MenuKeys
LyxType             charstyle
LabelString         menukey
LatexType           command
LatexName           menu
ResetsFont          true
End
```

```
InsetLayout Flex:KeysKeys
LyxType          charstyle
LabelString      okey
LatexType        command
LatexName        keys
ResetsFont       true
End
InsetLayout Flex:sinserif
LyxType          charstyle
LabelString      sinserif
LatexType        command
LatexName        sinserif
Preamble
  \newcommand*{\sinserif}[1]{\sffamily #1}
EndPreamble
Font
  Family          sans
EndFont
ResetsFont       true
End
InsetLayout Flex:ttmono
LyxType          charstyle
LabelString      ttmono
LatexType        command
LatexName        ttmono
Preamble
  \newcommand*{\ttmono}[1]{\ttfamily #1}
EndPreamble
Font
  Family          Typewriter
EndFont
ResetsFont       true
End
InsetLayout Flex:LibKB
LyxType          charstyle
```

```
LabelString LibKB
LatexType command
LatexName libkb
Preamble
  \newfontfamily{\libertinusKB}{Libertinus Keyboard}
  \newcommand*{\libkb}[1]{\libertinusKB #1}
EndPreamble
ResetsFont true
Font
  Family typewriter
EndFont
ResetsFont true
End
InsetLayout Flex:LogoTipo
LyxType charstyle
LabelString HoLogo
LatexType command
LatexName loguito
Preamble
  \newcommand*{\loguito}[1]{\hologo{#1}}
EndPreamble
ResetsFont true
End
InsetLayout Flex:PartNote
LyxType charstyle
LabelString PartNote
LatexType command
LatexName notitas
Preamble
  \newcommand*{\notitas}[1]{
    \partnote\itshape #1}
EndPreamble
ResetsFont true
End
```

Apéndice C

Preámbulo L^AT_EX

A continuación, el preámbulo de este documento. En las primeras líneas defino la fuente principal del texto. Esto quiere decir que he dejado en *predeterminado* la opción correspondiente en la configuración del documento, ¡a no olvidarse de eso! Tampoco hay que olvidar que he seleccionado TIPOGRAFÍAS NO-TEX PREDETERMINADAS para así poder configurar la fuente matemática y que he activado el uso de la microtipografía en el menú de las tipografías del documento.

```
\makeatletter
\renewcommand\@makefnmark[1]{%
  \noindent\makebox[0pt][r]
    {\hyperref[\BackrefFootnoteTag]{
\@thefnmark.\space}}#1}
\makeatother
\usepackage{footnotebackref}
%------
\setmainfont{EB Garamond}[
  Mapping=tex-text,
  RawFeature={+ss06},
UprightFont = EB Garamond Regular,
BoldFont = EB Garamond SemiBold,
ItalicFont = EB Garamond Italic,
BoldItalicFont = EB Garamond SemiBold Italic,
```

```
UprightFeatures = { SizeFeatures = {
  {Size = {-10}, Font = EB Garamond Medium},
  {Size = {10-}, Font = EB Garamond}}},
BoldFeatures = { SizeFeatures = {
  {Size = {-10}, Font = EB Garamond Bold},
  {Size = {10-}, Font = EB Garamond SemiBold}}},
ItalicFeatures = { SizeFeatures = {
  {Size = {-10}, Font = EB Garamond Medium Italic},
  {Size = {10-}, Font = EB Garamond Italic}}},
BoldItalicFeatures = { SizeFeatures = {
  {Size = {-10}, Font = EB Garamond Bold Italic},
  {Size = {10-}, Font = EB Garamond SemiBold
                    Italic}}},
]
\setmathfont{Garamond-Math}
%-----
\usepackage{hologo} % logo de XeTeX & compañía
\usepackage[os=win]{menukeys} % teclas y menús
\newfontfamily{\voll}[]{\Vollkorn} % ejemplos
\newfontfamily{\genplus}[]{\Gentium Plus}
% para el capítulo sobre letras capitulares --
\newfontfamily{\Iniciales}[RawFeature={+salt}]{Pali}
\renewcommand{\LettrineFontHook}{\Iniciales}
\setcounter{DefaultLines}{4}
\usepackage{magaz}

% Las leyendas -----
\usepackage{caption}
\captionsetup{justification=centerlast,%
  labelfont=sc,textfont=it}
% la sección de estilos de página -----
\renewcommand{\headrulewidth}{0.4pt}

% quitar el número de capítulo
% y sección de la cabecera -----
```

```

\renewcommand{\chaptermark}[1]{
\markboth{#1}{}
\renewcommand{\sectionmark}[1]{
\markright{#1}}

% el estilo de página normal, con sus cabeceras
\pagestyle{fancy}
\fancypagestyle{normal}{%
  \fancyhead{}
  \fancyhead[RO,LE]{\thepage}
  \fancyhead[LO]{\nouppercase{\leftmark}}
  \fancyhead[RE]{\nouppercase{\rightmark}}
  \fancyfoot{}
}
% el estilo del índice
\fancypagestyle{index}{%
  \fancyhead{}
  \fancyhead[RO,LE]{\thepage}
  \fancyhead[LO]{\nouppercase{Índice}}
  \fancyhead[RE]{\nouppercase{Índice}}
  \fancyfoot{}
}
% la sección de los títulos -----
\usepackage{titlesec}
\newfontfamily{\cormorant}{Cormorant}
\titleformat{\part}[display]
  {\huge\cormorant}{\filright%\thepart\
}{20pt}{\Huge}
\titleformat{\chapter}[frame]
  {\LARGE\cormorant}{\filright
  \chaptertitlename\
  \thechapter}{20pt}{\huge}
\titleformat*{\section}{\Large\bfseries\cormorant}
\titleformat*{\subsection}{\large\bfseries\cormorant}

```

```
% esto hace que las páginas \part queden sin pie
\makeatletter
\renewcommand\part{%
  \if@openright
    \cleardoublepage
  \else
    \clearpage
  \fi
  \thispagestyle{empty}%
  \if@twocolumn
    \onecolumn
    \@tempswatru
  \else
    \@tempswafalse
  \fi
  \null\vfil
  \secdef\@part\@spart}
\makeatother
\makeatletter
\def\@endpart{\vfil\newpage
              \if@twoside
              \if@openright
              \null
              \thispagestyle{empty}%
\vspace*{3cm}%
\begin{quote}%
\partnote
\end{quote}%
              \fi
              \fi
              \if@tempswa
                \twocolumn
              \fi}
\newcommand\partnote{}
\makeatother
```

Apéndice D

Otros formatos... y un saludo

Una pequeña lista de las *otras* cosas que he configurado para lograr este formato.

D.1 Otros formatos utilizados en este libro

Para este libro he activado los siguientes módulos:

- Marcos de color elaborados
- Capitales
- Listas personalizadas (enumitem)

El diseño de página usa un formato A5 con estilo elaborado y la opción «documento con dos caras».

Los márgenes son

SUPERIOR 2,5 cm

INFERIOR 1,5 cm

INTERIOR 2,5 cm

EXTERIOR 1,5 cm

He activado el soporte `Hyperref` permitiendo que los enlaces se quiebren en líneas, eliminando los marcos alrededor de los mismos y seleccionando de colorear los enlaces con

```
linkcolor=red,urlcolor=blue
```

En la pestaña Marcadores he seleccionado los MARCADORES NUMERADOS así el índice del PDF también muestra los números de las entradas.

Formato de salida PDF (XeTeX)...

Lo demás ya lo he dicho todo.

D.2 Y sí, también formato directo

¡Pero solo en una página! Me refiero, claro está, a la cubierta: definir estilos que solo se utilicen allí carece de sentido por lo que unos pocos clics derechos para cambiar el formato de los párrafos (llevarlos a alineación derecha) o del texto (el tamaño) he utilizado, junto a un par de saltos de página (`Insertar >> Formato`). Eso sí, el breve texto en Kleymisska utiliza un estilo... si bien a un tamaño modificado manualmente.

Lo que llevó un poco más de trabajo fue colocar la imagen.

La silueta del pingüino que observa las puertas de ~~Moria~~ de L^AT_EX la creé en LibreOffice Draw sobre el perfil de cierta imagen que utilicé como ejemplo en el capítulo 14. Los logos de L^AT_EX y compañía fueron creados con L^AT_EX usando la plantilla PDF (recortado) de la que hablé en su momento, convirtiéndolos en SVG e insertándolos en el dibujo de Draw junto a algunos rectángulos redondeados y gradientes.

Dado que Draw no permite exportar solo una selección al crear un PDF, la diapositiva del documento fue modificada para no tener márgenes y ser de la exacta dimensión necesaria para la imagen.

Es importante al exportar a PDF desde Draw el seleccionar «compresión sin pérdidas» para que la imagen no se arruine... pero bueno, me estoy yendo de tema, que después de todo este libro es sobre L^AT_EX.

A partir de ese documento Draw creé entonces un archivo PDF que inserté en la página dedicada a la cubierta de este documento.

Y aquí viene la parte interesante. Crear un estilo de página con sus márgenes solo para la cubierta no tiene sentido, pero si utilizo solo el ancho del

área de texto para la imagen esta quedaría muy pequeña por lo que indicar explícitamente el tamaño de la imagen como comenté en el capítulo 14 «suena a solución»... salvo por el pequeño problema de hacer que la imagen comience *antes* del margen izquierdo.

¡Todo se resuelve! A continuación te detallo la secuencia de formatos directos utilizada.

Primero una caja T_EX con el siguiente código

```
\begingroup
\leftskip=-3em
```

La segunda instrucción hace que el párrafo a continuación adquiriera una sangría negativa de 3 em... y también el siguiente y todos los párrafos que existan después: por eso fue necesario crear un *grupo* con la primera instrucción para así aislar los efectos de `\leftskip` del resto del documento.

Debajo de esa caja T_EX inserté la imagen PDF que comenté antes, haciendo que tome el 81 % del ancho de la página completa.

Finalmente, a continuación de la imagen otra caja T_EX con un

```
\endgroup
```

para cerrar el grupo y terminar el proceso.

Para la contra cubierta utilicé también algunos espacios verticales desde `\Insertar >> Formato >> Espacio vertical`, pero esos son fáciles de comprender.

Y sí, tuve que compilar el documento varias veces hasta lograr ajustar todo y que quedara bien. Qué se le va a hacer.

D.3 Saludo final

Si has llegado hasta aquí, lector, ¡gracias por tu paciencia!

Espero que hayas disfrutado el libro. Yo ciertamente disfruté escribiéndolo. Tratando de explicar es que he aprendido más de lo que creí que llegaría a saber. El momento en el que decidí escribir este libro se une ahora al momento en el que cumpliste tu propósito de leerlo hasta el final. El viaje ha sido largo, pero creo que valió la pena.

¡Hasta la próxima *Aventura Tipográfica*!