



Universidad de
Deusto

Deustuko
Unibertsitatea

.


**Facultad de
Ingeniería
ESIDE**
Ingeniero en
Informática

**Ingeniaritza
Fakultatea
ESIDE**
Informatikako
Ingeniaria

La memoria del proyecto fin de carrera en OpenOffice.org: how-to & why-to

JosuKa Díaz Labrador

Bilbao, marzo de 2006

	Universidad de Deusto Facultad de Ingeniería	La memoria del proyecto fin de carrera en OpenOffice.org: how-to & why-to	Proyecto Fin de Carrera
---	---	--	----------------------------

Resumen

La memoria del proyecto fin de carrera en OpenOffice.org: *how-to & why-to*

Versión 0.6 de 04/04/2006 por JosuKa Díaz Labrador

Copyright © 2004, 2006 JosuKa Díaz Labrador

Permission is granted to copy, distribute and/or modify this document under the terms of the GNU Free Documentation License, Version 1.2 or any later version published by the Free Software Foundation; with the Invariant Sections being "Acerca de las licencias" and "Conclusiones", no Front-Cover Texts, and no Back-Cover Texts. A copy of the license is included in the section entitled "GNU Free Documentation License".

Contacto:

JosuKa Díaz Labrador [<http://paginaspersonales.deusto.es/josuka/>]


Dpto. de Ingeniería del Software, Facultad de Ingeniería, Universidad de Deusto

Apartado 1 - 48080 BILBAO (SPAIN)

Descriptores


Memoria del proyecto fin de carrera

OpenOffice.org

 <p>Universidad de Deusto</p> <p>Facultad de Ingeniería</p>	<p>La memoria del proyecto fin de carrera en OpenOffice.org: how-to & why-to</p>	<p>Proyecto Fin de Carrera</p>
--	--	------------------------------------

Contenido

Capítulo 1. Introducción.....	1
1.1 Objetivos.....	1
1.1.1 Acerca de la versión 0.6.....	1
1.2 Uso del documento-plantilla.....	2
1.2.1 El escudo de la Universidad de Deusto.....	3
1.2.2 El título de la memoria.....	3
1.2.3 El resto de la portada.....	4
1.2.4 Resto del documento.....	4
1.3 ¿Por qué se ofrecen dos plantillas?.....	4
1.4 Acerca de las licencias.....	5
1.5 Organización del documento.....	6
1.6 Agradecimientos.....	6
Capítulo 2. Máquinas de Turing.....	7
2.1 Nociones preliminares.....	7
2.2 Objetivos.....	7
2.3 Representación gráfica.....	8
2.4 Reconocimiento de cadenas.....	8
2.5 Objetivos.....	9
2.5.1 Objetivos de la cosa que hay que hacer y bla, bla, bla hasta que me canse de poner lo que me dé la gana a mí de una vez.....	9
2.5.2 Otros objetivos.....	9
2.5.3 Otros objetivos más.....	9
2.5.4 Hitos.....	10
2.5.5 Hitos.....	10
2.5.6 Hitos.....	10
2.5.7 Hitos.....	10
2.5.8 Hitos.....	10
2.5.9 Hitos.....	10
2.5.10 Hitos.....	10
2.5.11 Hitos.....	10
2.5.12 Hitos que van y vienen por ahí, y se encargan de transmitir una sensación de trabajo bien hecho.....	10
2.6 Objetivos.....	11
2.7 Objetivos.....	11
2.8 Objetivos.....	11
2.9 Objetivos.....	11
2.10 Objetivos de la cosa que hay que hacer y bla, bla, bla hasta que me canse de poner lo que me dé la gana a mí de una vez.....	11
Capítulo 3. Explicamos el documento-plantilla.....	13
3.1 Tipos de letra.....	13
3.1.1 Tipos propietarios: Postscript y TrueType.....	13

	Universidad de Deusto Facultad de Ingeniería	La memoria del proyecto fin de carrera en OpenOffice.org: how-to & why-to	Proyecto Fin de Carrera
--	---	--	----------------------------


3.1.2	Tipos libres: OpenOffice.org.....	15
3.1.3	Tipos libres: URW++ en CTAN.....	15
3.1.4	Tipos libres: Freefont.....	16
3.1.5	En definitiva.....	18
3.2	Formatos de párrafo.....	19
3.2.1	Cuerpo.....	19
3.2.2	Títulos.....	20
3.2.3	Listas numeradas y no numeradas.....	20
3.3	Elementos flotantes.....	20
3.3.1	Marcos flotantes.....	21
3.3.2	Referencias en el texto.....	21
3.4	Partes del documento (formato de página).....	22
3.4.1	Cuerpo de la memoria.....	22
3.4.2	Portada.....	23
3.4.3	Resumen e índices.....	23
3.4.4	Bibliografía.....	24
3.4.5	Anexos.....	24
	Capítulo 4. Conclusiones.....	25
	Bibliografía.....	27
	Anexo A. GNU Free Documentation License.....	29

Lista de figuras

Figura 2.1.	Los planos de la máquina de Turing.....	8
Figura 2.2.	Otra máquina de Turing más a nuestra manera de verlas.....	9

Lista de tablas

Tabla 2.1.	La primera máquina que Turing describió en su famoso artículo de 1936 y que ha dado pie a diversas interpretaciones.....	7
Tabla 3.1.	Correspondencia de tipos Postscript y TrueType.....	14
Tabla 3.2.	Tipos proporcionados por OpenOffice.org.....	15
Tabla 3.3.	Correspondencia de tipos Postscript y URW.....	16
Tabla 3.4.	Tipos de letra del proyecto Freefont.....	17

	Universidad de Deusto Facultad de Ingeniería	La memoria del proyecto fin de carrera en OpenOffice.org: how-to & why-to	Proyecto Fin de Carrera
---	---	--	----------------------------

Capítulo 1. Introducción

1.1 Objetivos

Este documento puede considerarse como el manual de referencia del *documento-plantilla* construido en OpenOffice.org para la escritura de la memoria del proyecto fin de carrera: explica algunos detalles sobre su implementación, uso y mejora. A su vez, también sirve de ejemplo del mismo, pero en el presente documento se han modificado ciertos detalles (como la hoja de título y otros), por lo que para escribir la memoria, se recomienda encarecidamente empezar con la plantilla original, que está en los archivos:

MPFC_0.6_freefont.sxw (usa tipos de letra libres)
MPFC_0.6_nofreefont.sxw (usa tipos de letra no libres)

Historia (de este documento):


0.4 de 13/03/2004 por JosuKa Díaz Labrador.
0.5 de 25/03/2006 por JosuKa Díaz Labrador.
0.6 de 04/04/2006 por JosuKa Díaz Labrador.

1.1.1 Acerca de la versión 0.6

La versión 0.6 se ha construido actualizando los documentos originales de la versión 0.4 (versión 1.1.0 de OpenOffice.org) para la versión 2.0 de OpenOffice.org.

Aparte de eso, los únicos cambios que se han realizado son, por un lado, la portada de la memoria (para acogerse al nuevo formato establecido en marzo de 2006) y por otro, la forma de escribir el título de la misma. Ambos aspectos se explican someramente en los apartados de la sección siguiente.

El resto del contenido de los documentos no se ha revisado, por lo que en ocasiones pueden aparecer comentarios que tienen sentido para la versión antigua de OpenOffice.org (1.1.0) y no para la actual (2.0).

	Universidad de Deusto Facultad de Ingeniería	1. Introducción	Proyecto Fin de Carrera
--	---	-----------------	----------------------------

1.2 Uso del documento-plantilla

Una plantilla, en el ámbito de la edición de textos, tal como OpenOffice.org o Hasecorp Word la entienden, es una colección de *estilos*. Se supone que el usuario escribe su texto usando los diferentes estilos que componen la plantilla, en lugar de ir aplicando constantemente formatos de una manera desordenada. A este respecto, una de las características fundamentales de un documento para ser correcto es la *uniformidad*, es decir, el seguimiento de un patrón visual a lo largo del mismo.


Por otro lado, a lo largo de nuestra experiencia como usuarios de la herramienta de edición de documentos, hemos podido comprobar que pocas personas sacan partido de las grandes posibilidades que ofrecen estos programas. Aquí, nos referimos a la *automatización*: se trata de que nosotros elaboremos el contenido propiamente dicho, dejando que un gran número de elementos accesorios pero importantes surjan de forma automática. Así, hablamos de los contenidos de los encabezamientos y pies de páginas (incluyendo, pero no solamente, los números de página), los índices, la numeración de títulos de cualquier nivel, la numeración de figuras, tablas y otros elementos no textuales, las referencias a los mismos, etc.

Es importante ver un documento (y más en el caso de un documento extenso como la memoria) como algo cambiante, de hecho, muy cambiante. El mero hecho de querer insertar, por ejemplo, un capítulo nuevo entre otros ya redactados en un documento complejo en elaboración puede suponer a la mayor parte de usuarios de procesadores de texto un esfuerzo de varias horas (dependiendo de la complejidad del documento), porque han de realizar un número elevado de cambios en lo *ya redactado*. Con una plantilla bien construida y automatizada, se puede realizar en segundos, realmente lo que cuesta el propio hecho de insertar nuevo texto, porque todos los cambios del resto del documento se realizan *automáticamente*.

Y por supuesto, no hay que desdeñar otro aspecto: la automatización implica *corrección*, es decir, sabemos que la numeración de los títulos es la correcta, que los números de página que pone en el índice son los correctos, y así sucesivamente. Cualquier manera no automática de obtener estos elementos solo puede llevar a un resultado: errores.

Casi todo lo explicado hasta ahora puede quedar más o menos reflejado en la plantilla. Sin embargo, un documento extenso requiere además una *estructura compleja* de diferentes partes: portada, hojas preliminares, índices, capítulos, anexos, etc. han de estar organizados correctamente. Esto no puede reflejarse en una plantilla que carece de contenido. Por esa razón, creemos más conveniente ofrecer un *documento-plantilla*, que además de servir como plantilla por contener los estilos necesarios, presenta como contenido esa estructura compleja de apartados ya elaborada, o la parte principal de la misma, de forma que el usuario de la misma solo tenga que escribir su texto en los “huecos” preparados.

Después de esta larga introducción podemos explicar la mejor manera, a nuestro entender, de usar el documento-plantilla: primero cambiamos el nombre del archivo por el que más nos guste; después lo abrimos con el OpenOffice.org, y se verá que se trata de sustituir el contenido ya preparado por el que vaya a tener nuestra memoria. Sin embargo, hay tres elementos cruciales para empezar, que tienen que ver básicamente con la portada (aunque no solo): la imagen con el escudo de la UD y el título del documento. Lo explicamos en los siguientes apartados.

 <p>Universidad de Deusto</p> <p>Facultad de Ingeniería</p>	1.2. Uso del documento-plantilla	Proyecto Fin de Carrera
--	----------------------------------	----------------------------

1.2.1 El escudo de la Universidad de Deusto

La portada de la memoria contiene varios elementos que hay que manejar con cuidado, para no desviarse de la portada establecida: el escudo de la UD, el título de la memoria, el autor o autores, el nombre del director y la fecha. Recomendando prestar atención a las siguientes explicaciones.

En primer lugar, estos elementos están colocados en posiciones específicas, mediante cajas (o “marcos”, técnicamente hablando) flotantes. Habría que tener cuidado por lo tanto en no recolocarlos en otro lugar: por esa razón, su tamaño y posición se han protegido (y en algún caso, también el contenido).

El primer marco flotante es el escudo de la UD. Se trata de una imagen específica para cada titulación, lo cual es lógico, pero supone un reto para una plantilla (la cosa no sería problemática si la titulación se hubiera escrito como texto). Sin embargo, es posible ofrecer una solución rápida y relativamente sencilla.


Esa imagen se ha colocado ahí mediante un *enlace automático* a un archivo llamado `escudo.jpg`, que debe estar en el mismo directorio que el documento (puedes comprobarlo pulsando con el botón derecho del ratón encima de la imagen, seleccionando “Imagen...”, y la pestaña “Imagen”: verás que aparece la dirección del archivo). Cada vez que se abre el documento, OpenOffice.org lee el archivo y lo presenta en el lugar previsto.

Por lo tanto, hacer que aparezca la imagen de otra titulación (si eres de informática, solo tienes que preocuparte de que el archivo `escudo.jpg` esté en el mismo directorio que el documento) es bien fácil: sin que esté abierto el documento en OpenOffice.org, haz una copia de la imagen correcta (directorio `escudos`) con el nombre `escudo.jpg`, y después lleva este archivo al directorio en que esté el documento con la memoria. Una vez hecho esto, la imagen correcta aparecerá mágicamente al abrir el documento con OpenOffice.org, y solo has de preocuparte de que el archivo `escudo.jpg` esté siempre ahí.

1.2.2 El título de la memoria

El título de la memoria ha de aparecer, lógicamente, en la portada. Sin embargo, también ha de constar en los encabezamientos de las primeras páginas de cada capítulo. Esto podría haberse resuelto sin más copiando el texto desde la portada a los encabezamientos, pero tendría la desventaja de que entonces el usuario del documento-plantilla tendría que volver a hacer lo mismo. Pero eso no es lo peor. Imagina que por la razón que sea, decides durante la escritura de la memoria un pequeño cambio en el título: o bien te acuerdas de que tienes que volver a copiarlo en los encabezamientos (pero tienes que hacerlo, y son varios) o bien te olvidas, con lo que tu memoria quedará fea. Por lo tanto, aquí tenemos un claro ejemplo de la ayuda que ofrece la automatización.

La solución es fácil y ya está implementada en el documento-plantilla: el título se escribe en un solo lugar, y cada vez que queremos sacarlo, usamos *campos* (*fields* en inglés), uno de los aspectos más relevantes para aprender de OpenOffice.org. En concreto, el título ha de escribirse como una *propiedad del documento*. Elige el menú “Archivo/Propiedades”, y la pestaña “Usuario”. Verás que la primera se llama “Título del proyecto”. Escribe en el cuadro de texto el título de tu proyecto, y de nuevo mágicamente, este aparecerá tanto en la portada como en los encabezamientos. Fácil, rápido y seguro, ¿verdad?

	Universidad de Deusto Facultad de Ingeniería	1. Introducción	Proyecto Fin de Carrera
--	---	-----------------	----------------------------

Si después de hacer esto, no aprecias cambio, es porque a veces los campos no se actualizan automáticamente: la tecla F9 te permitirá forzar esa actualización. Pulsa y verás.

1.2.3 El resto de la portada

En lo demás de la portada, solamente hay que escribir en los huecos presentados. Eso sí, poned vuestros nombres como autores del proyecto (si sois varios) en líneas consecutivas, y por favor, escribid el mes en minúsculas.

1.2.4 Resto del documento

Seguimos: la segunda página contiene el resumen y los descriptores, solo hay que borrar lo escrito y añadir lo correspondiente al proyecto. La parte de índices creo que no habrá que tocarla, saldrán automáticamente según cambie vuestro documento (en cualquier caso, para actualizarlos manualmente, pulsad en cada índice, luego botón derecho, y seleccionad “Actualizar índice”).

A partir de ahí, utilizad los capítulos ya preparados para escribir vuestro texto. Más adelante, se indica la manera de añadir nuevos capítulos, anexos, o lo que sea.

Atención por otro lado, al borrado (tanto de partes pequeñas como grandes). A veces, el comportamiento de OpenOffice.org es un tanto “raro” (o al menos, no he conseguido pillarlo del todo): la operación de deshacer será vuestra principal aliada.


1.3 ¿Por qué se ofrecen dos plantillas?

Más adelante se profundiza en este asunto, pero podemos avanzar el problema. Los *tipos de letra* (por cierto, estamos absolutamente acostumbrados a llamarlos *fuentes*, a pesar de que opino que no es una buena denominación en español) no son programas estrictamente, pero están sujetos como ellos a derechos de uso y copia.

Existen algunas alternativas libres, bajo la GPL incluso, pero no es precisamente un tema en el que la comunidad de *software* libre haya avanzado demasiado. Entre las alternativas que se han encontrado, dos resultan excluidas por problemas graves, y solo una tercera, el proyecto *Freefont*, cumple los requisitos mínimos exigibles para el desarrollo de esta clase de documentos. Aún así, este repertorio libre de tipos de letra (diseñado en el mismo formato Truetype que utilizan los tipos propietarios disponibles en Hasefroch y otros sistemas) adolece de algunos problemas, que aun siendo leves, pueden resultar ciertamente molestos para el desarrollo del documento.

Por esa razón, se ofrece una versión del documento-plantilla (con la palabra *nofreefont* en el nombre) que usa los tipos de letra propietarios *BookAntiqua*, *Arial* y *CourierNew*, que generalmente se encuentran disponibles en cualquier versión de Hasefroch. Y se ha preparado otra versión que utiliza tipos de letra libres (de ahí que incluya *freefont* en el nombre), concretamente, los tipos *FreeSerif*, *FreeSans* y *FreeMono* del proyecto *Freefont*, que pueden instalarse tanto en GNU/Linux como en Hasefroch.

Aparte de esto, la propuesta de tipos de letra elegidos se ha basado en preferencias puramente personales, que, por descontado, puedes modificar a tu criterio. Se explica posteriormente.

 <p>Universidad de Deusto</p> <p>Facultad de Ingeniería</p>	1.4. Acerca de las licencias	Proyecto Fin de Carrera
--	------------------------------	----------------------------

1.4 Acerca de las licencias

Aviso para navegantes: este apartado va a ser un poco *rayante*, pero los *frikis* estarán encantados.

Tanto el documento-plantilla como este manual de referencia tienen licencias *copyleft*, concretamente las de GNU [Free Software Foundation]. Que este documento, como manual de referencia que es, tenga la *GNU Free Documentation License* (FDL) parece algo obvio, es justo el tipo de documento para el que está pensada. Más sorprendente puede resultar que el documento-plantilla tenga la *GNU General Public License* (GPL), por lo que parece necesaria cierta explicación.

Desde luego, a primera vista choca que una licencia como la GPL, pensada para el concepto de *programa*, se esté aplicando a un *documento* textual, algo que no parece que tenga que ver con un programa. Pero ese es justamente el paralelismo que yo planteo.


Afirmo: el documento-plantilla tiene ciertas características que se asemejan a las de un programa (también algunas diferencias, está claro). Primero, un programa puede darse en código fuente o en código ejecutable. Yo entiendo que en el documento-plantilla, el *código fuente* es lo mismo que el *código ejecutable*, es el formato propio de OpenOffice.org, es decir, la extensión SXW. Realmente, que sea código fuente es más verdad de lo que parece, porque (como se explica en el capítulo de conclusiones), un archivo SXW no es más que un archivo comprimido ZIP, en cuyo interior hay varios documentos XML con el contenido del documento, más las imágenes que utilice.

Entiendo que *ejecutar* o usar el programa (el documento-plantilla) es el mero hecho de abrirlo en OpenOffice.org, y usarlo para componer un documento nuevo. El programa (la parte que se entiende como “plantilla”) permite la realización de unos cálculos (la edición de texto con un formato, estructura, etc.) para obtener unos resultados (en este caso, un documento de texto con un contenido dado). Está claro que todo esto es posible porque está OpenOffice.org por detrás, pero piensa en cómo obtener los mismos resultados sin el documento-plantilla.

En la medida que la GPL no impide *ejecutar* el trabajo protegido por la misma a cualquier persona que lo obtenga, tienes plena libertad para ello, sin que esto tenga, *a priori*, ninguna consecuencia sobre el resultado obtenido, es decir, sobre los documentos que desarrolles a partir de la plantilla, los PDF que generes o las impresiones que hagas. Es decir, entiendo, *a priori*, que la naturaleza *vírica* que se suele asociar a la GPL no es aplicable en este caso, no es obligatorio que tu documento lleve ninguna licencia en especial (GPL, FDL u otra, salvo que tú expresamente quieras que la lleve), porque solamente se estaría ejecutando el “programa”, y eso queda fuera de la GPL (artículo 0, segundo párrafo).

Sin embargo, la analogía no es tan simple como parece (yo lo achaco precisamente a esa doble identidad de plantilla y documento, por un lado, y código fuente y código ejecutable, por otro). Veamos, probablemente tendrás que añadir algún estilo, o modificar alguno ya existente. Eso cuenta como *modificación* de la plantilla, y por lo tanto, aquí sí se aplica la GPL. En principio, esto está permitido, y sigue sin afectar a los documentos que obtengas con la misma. Por supuesto, si quisieras distribuir tu alteración de la plantilla, deberías hacerlo bajo la GPL, eso está claro, pero igual que alguien que modifica un programa bajo la GPL.

Todo lo anterior parece correcto, salvo por un pequeño detalle: el documento de texto que obtienes (equivalente a la “salida del programa”) *contiene la plantilla original* (o la alterada). Normalmente, los resultados de un programa no incluyen el propio programa. En unos pocos, pero afortunados para los *frikis*, casos (como el programa *bison* de generación de

	Universidad de Deusto Facultad de Ingeniería	1. Introducción	Proyecto Fin de Carrera
--	---	-----------------	----------------------------

analizadores sintácticos ascendentes [Donnelly y Stallman 1999] o ciertas máquinas de Turing [Díaz Labrador 1994]), la salida del programa contiene el mismo programa. Esto significa que el usuario del programa no puede deshacerse de la GPL ni siquiera por el mero hecho de ejecutar el mismo y querer utilizar sus resultados (lee ahora otra vez el artículo 0, segundo párrafo de la GPL). He estado analizando la solución que se da en *bison* para este problema, creo que la entiendo, pero no acabo de ver cómo se aplicaría a este caso.

Total, que no tengo excesivamente claro qué consecuencias tiene lo explicado. Por supuesto, al poner el documento-plantilla bajo la GPL, deseo que cualquier mejora de ese trabajo *como tal* que vaya a distribuirse permanezca bajo la GPL. Pero también ha de estar claro que *no quiero obligar* a cualquier *usuario* de la plantilla a que sus documentos (hayan sido obtenidos con la plantilla original o con alguna alteración) tengan que quedar necesariamente bajo la GPL. Me parece que esa es una opción personal que cada cual ha de elegir, y que yo, particularmente, no tengo ninguna intención de restringir.

Dicho esto, *alea jacta est*. El documento-plantilla se distribuye bajo la GPL, y solo me gustaría saber qué consecuencias últimas plantea este hecho. Serán bienvenidos los comentarios.

1.5 Organización del documento

En el resto del presente documento hay primero un capítulo de contenido “aleatorio”, de ejemplo, casi totalmente similar al que aparece en el documento-plantilla, pero con la primera tabla y la primera figura sin colocar en su lugar adecuado, para mostrar los pasos a dar para su colocación (se explica luego).

El siguiente capítulo explica ya varios pormenores del documento-plantilla, a veces como justificación o razón de las elecciones realizadas, a veces como indicación de lo que hay que hacer para usarla en todo su potencial, y a veces mostrando las pautas para su posible modificación o enriquecimiento.

En las conclusiones, a continuación, se recogen unas pocas opiniones personales sobre la experiencia que ha supuesto esta primera toma de contacto con el sistema OpenOffice.org, aunque solo haya sido con la herramienta de edición.

El anexo que sigue a la bibliografía recoge la *GNU Free Documentation License*, por la cual se rige, como se ha dicho, este documento.

1.6 Agradecimientos

Primero, al *e-ghost* (<http://eghost.deusto.es/>) y todos los que en él participan, por su labor en pro del *software* libre. Más en particular, Nando Quintana (lovelace) me ha ayudado mucho en algunas pruebas del documento en GNU/Linux; Iker Sagasti (aktor) proporcionó en la lista de correo del *e-ghost* unas recetas estupendas sobre la instalación y uso de fuentes escalables en GNU/Linux; finalmente, Pablo Garaizar (txipí) fue el que animó a iniciar este trabajo, y además, ha aportado algunos puntos de vista “filosóficos” que han resultado muy clarificadores.

Capítulo 2. Máquinas de Turing

En este capítulo inicial, esto no estaba. En este capítulo inicial, esto no estaba. En este capítulo inicial, esto no estaba. En este capítulo inicial, esto no estaba. En este capítulo inicial, esto no estaba. En este capítulo inicial, esto no estaba. En este capítulo inicial, esto no estaba. En este capítulo inicial, esto no estaba. En este capítulo inicial, esto no estaba.

2.1 Nociones preliminares


Los autómatas finitos, expresiones regulares, gramáticas regulares, gramáticas independientes del contexto y autómatas con pila, todos ellos estudiados en la primera parte de la Teoría de la Computación, se han mostrado como modelos matemáticos útiles, pero todavía sin la capacidad suficiente para representar algunos (quizá muchos) lenguajes, como se ve en la *tabla 2.1*.

2.2 Objetivos

De hecho, observamos que los tres primeros modelos citados especifican una primera clase de lenguajes, los lenguajes regulares, y demostramos que existen lenguajes que no son regulares; del mismo modo, los otros dos modelos representan los lenguajes de una segunda clase, los lenguajes independientes del contexto, que contiene propiamente a la anterior, y

	<i>0</i>	<i>1</i>	<i>Ninguno</i>
<i>b</i>			<i>(0, c)</i>
<i>c</i>	<i>(D, e)</i>		
<i>e</i>			<i>(1, f)</i>
<i>f</i>		<i>(D, b)</i>	

Tabla 2.1. La primera máquina que Turing describió en su famoso artículo de 1936 y que ha dado pie a diversas interpretaciones

	Universidad de Deusto Facultad de Ingeniería	2. Máquinas de Turing	Proyecto Fin de Carrera
--	---	-----------------------	----------------------------

también probamos que existen lenguajes con mayor “complejidad”, que están fuera de dicha clase.

En definitiva, se precisan modelos matemáticos más potentes para representar estos lenguajes que no pertenecen a las clases anteriores.

2.3 Representación gráfica

La maquinilla de marcos puede construirse con ayuda de los planos que se adjuntan en la *figura 2.1*. Ahí se ve que hay una cinta infinita, con celdas numeradas correlativamente desde 0, una cabeza que tiene capacidad para leer el contenido de cada celda, pero también para escribir en ella algún otro símbolo, y que además, puede moverse a derecha e izquierda.

Turing practicaba atletismo, concretamente carreras de fondo, e inventó (no me acuerdo muy bien si en colaboración con algún colega) una modalidad para jugar al ajedrez que bien podría ser incluida en las Olimpiadas: solo se podía realizar la siguiente jugada después de haber dado varias vueltas al edificio. De esta manera, pretendían observar la influencia del ejercicio físico en la mente, el razonamiento, o lo que sea que tenemos dentro del cráneo. Desconozco las conclusiones del experimento.

2.4 Reconocimiento de cadenas

La conversión de la máquina de Turing en un dispositivo reconocedor de cadenas se puede realizar muy fácilmente si adoptamos la filosofía general de las máquinas reconocedoras. Por un lado, debemos proveer a la máquina con la cadena a examinar; para ello, colocaremos dicha cadena en la cinta antes de empezar el proceso (en lugar de suponer que está en blanco).

Aún así, seguimos considerando esa cinta como un espacio de trabajo (y de tamaño ilimitado por la derecha), más que como una mera entrada, como ocurría en los autómatas finitos y en los autómatas con pila; por tanto, la máquina sigue pudiendo leer el símbolo

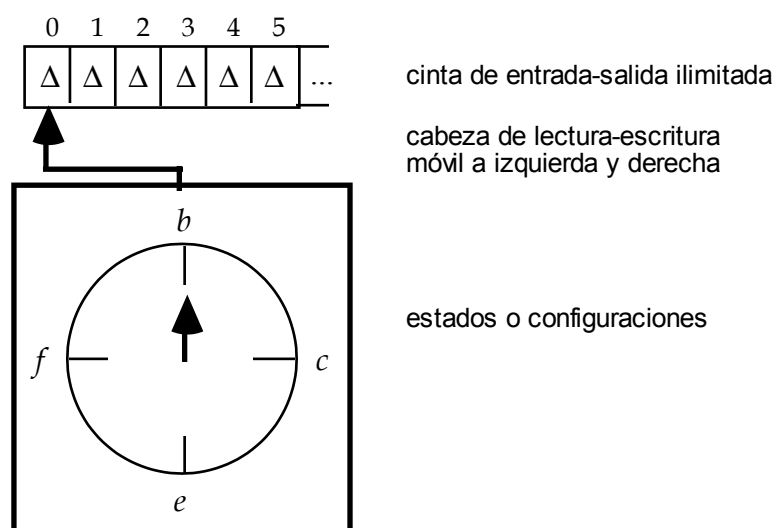



Figura 2.1. Los planos de la máquina de Turing

	Universidad de Deusto Facultad de Ingeniería	2.4. Reconocimiento de cadenas	Proyecto Fin de Carrera
---	---	--------------------------------	-------------------------

almacenado en la celda actual o escribir otro símbolo en su lugar, también puede usar otros símbolos (los que Turing llamaba de la segunda clase), además del Δ que se supone que tienen todas las celdas a la derecha de la cadena de entrada en el momento de iniciar el proceso de reconocimiento.

2.5 Objetivos

La figura 2.2 contiene otro ejemplo en el que se observa la representación más habitual, mediante un diagrama de estados.

2.5.1 Objetivos de la cosa que hay que hacer y bla, bla, bla hasta que me canse de poner lo que me dé la gana a mí de una vez

El ejemplo nos ha mostrado que puede ser fundamental conocer el límite izquierdo de la cinta; si retomamos la idea intuitiva de Turing referente a que la cinta de trabajo es como un “cuaderno” ilimitado, podríamos pensar que es fácil saber que estamos en la primera hoja del mismo; sin embargo, ello obligaría a introducir una condición especial adicional en la definición de transición, que podría complicar la forma de las máquinas.

2.5.2 Otros objetivos

En lugar de esto, podemos idear otra solución: para conocer el límite derecho nos basamos en la aparición del símbolo Δ en la celda actual; podemos hacer que sea la misma condición para conocer el límite izquierdo si antes de poner en funcionamiento la máquina, en lugar de colocar la cadena a partir de la celda 0, la colocamos a partir de la celda 1 y hacemos que la celda 0 tenga el símbolo Δ .

2.5.3 Otros objetivos más

Esta será la configuración inicial de la cinta de trabajo, donde también hemos hecho que la celda de partida sea el número 1, la que tiene el símbolo más a la izquierda de la cadena.

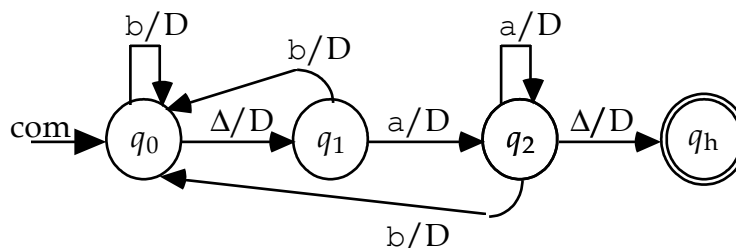



Figura 2.2. Otra máquina de Turing más a nuestra manera de verlas

	Universidad de Deusto Facultad de Ingeniería	2. Máquinas de Turing	Proyecto Fin de Carrera
--	---	-----------------------	----------------------------

2.5.4 Hitos

Esta será la configuración inicial de la cinta de trabajo, donde también hemos hecho que la celda de partida sea el número 1, la que tiene el símbolo más a la izquierda de la cadena.

2.5.5 Hitos

Esta será la configuración inicial de la cinta de trabajo, donde también hemos hecho que la celda de partida sea el número 1, la que tiene el símbolo más a la izquierda de la cadena.

2.5.6 Hitos

Esta será la configuración inicial de la cinta de trabajo, donde también hemos hecho que la celda de partida sea el número 1, la que tiene el símbolo más a la izquierda de la cadena.

2.5.7 Hitos

Esta será la configuración inicial de la cinta de trabajo, donde también hemos hecho que la celda de partida sea el número 1, la que tiene el símbolo más a la izquierda de la cadena.

2.5.8 Hitos

Esta será la configuración inicial de la cinta de trabajo, donde también hemos hecho que la celda de partida sea el número 1, la que tiene el símbolo más a la izquierda de la cadena.

2.5.9 Hitos

Esta será la configuración inicial de la cinta de trabajo, donde también hemos hecho que la celda de partida sea el número 1, la que tiene el símbolo más a la izquierda de la cadena.

2.5.10 Hitos


Esta será la configuración inicial de la cinta de trabajo, donde también hemos hecho que la celda de partida sea el número 1, la que tiene el símbolo más a la izquierda de la cadena.

2.5.11 Hitos

Esta será la configuración inicial de la cinta de trabajo, donde también hemos hecho que la celda de partida sea el número 1, la que tiene el símbolo más a la izquierda de la cadena.

2.5.12 Hitos que van y vienen por ahí, y se encargan de transmitir una sensación de trabajo bien hecho

Esta será la configuración inicial de la cinta de trabajo, donde también hemos hecho que la celda de partida sea el número 1, la que tiene el símbolo más a la izquierda de la cadena.

 <p>Universidad de Deusto</p> <p>Facultad de Ingeniería</p>	2.6. Objetivos	Proyecto Fin de Carrera
--	----------------	----------------------------

2.6 Objetivos

Esta será la configuración inicial de la cinta de trabajo, donde también hemos hecho que la celda de partida sea el número 1, la que tiene el símbolo más a la izquierda de la cadena.

2.7 Objetivos

Esta será la configuración inicial de la cinta de trabajo, donde también hemos hecho que la celda de partida sea el número 1, la que tiene el símbolo más a la izquierda de la cadena.

2.8 Objetivos


Esta será la configuración inicial de la cinta de trabajo, donde también hemos hecho que la celda de partida sea el número 1, la que tiene el símbolo más a la izquierda de la cadena.

2.9 Objetivos

Esta será la configuración inicial de la cinta de trabajo, donde también hemos hecho que la celda de partida sea el número 1, la que tiene el símbolo más a la izquierda de la cadena.

2.10 Objetivos de la cosa que hay que hacer y bla, bla, bla hasta que me canse de poner lo que me dé la gana a mí de una vez

Esta será la configuración inicial de la cinta de trabajo, donde también hemos hecho que la celda de partida sea el número 1, la que tiene el símbolo más a la izquierda de la cadena.

	Universidad de Deusto Facultad de Ingeniería	La memoria del proyecto fin de carrera en OpenOffice.org: how-to & why-to	Proyecto Fin de Carrera
---	---	--	----------------------------

Capítulo 3. Explicamos el documento-plantilla

En este capítulo se explica la hoja de estilos, pero también otras cosas igual o más importantes. Intentaré ser lo más conciso posible (ja).


3.1 Tipos de letra

Primera guerra. Hay que definir tres estilos de letra que pueden resultar de utilidad para la memoria. La letra “serif” o “romana” ha de usarse para el cuerpo del texto, algo que queda fuera de toda discusión, como indica Martínez de Sousa [1999]. Para enfatizar lo que no es cuerpo de texto, mi postura personal (que admite discusión) es usar la letra “sans serif” o “paloseco”: para los títulos, para las etiquetas de los elementos flotantes (tablas, figuras, *etc.*), para sus referencias, para los encabezamientos y pies de página, y me queda la duda de si usarlo para los contenidos de los índices. Finalmente, entiendo también como una obligación usar letra “monoespaciada” o de ancho constante para todo lo que sean programas, entradas o salidas de los mismos, nombres de variables, funciones... ya sean elementos flotantes o estén en el propio cuerpo del texto.

Ahora bien, dado esto, hay que elegir una familia concreta en cada caso. Los tipos de letra, ¡cómo no!, también están sujetos a derechos de copia, uso y demás, por lo que la cosa no es tan fácil como abrir el desplegable de tipos de letra. En cualquier caso, presumo que esta plantilla puede ser usada en básicamente dos sistemas operativos, GNU/Linux y Hasefroch, que tienen diferentes modos de ver este asunto y diferentes familias en cualquier caso.

3.1.1 Tipos propietarios: Postscript y TrueType

Siguiendo un criterio puramente técnico-estilístico, considero que el repertorio de tipos Postscript (lenguaje desarrollado por Adobe) contiene todo lo necesario básicamente para resolver los problemas de edición. Como letras romanas están la archifamosa *Times*, pero también *Bookman*, *NewCenturySchoolbook* y *Palatino*: esta última es mi preferida para cuerpo de texto con diferencia. Como letras paloseco están *AvantGarde*, *Helvetica* y *HelveticaNarrow* (una versión condensada de la anterior). La letra de ancho fijo es la también famosa *Courier*.

	Universidad de Deusto Facultad de Ingeniería	3. Explicamos el documento-plantilla	Proyecto Fin de Carrera
--	---	--------------------------------------	----------------------------

Finalmente, dispone de unos tipos “especiales”, como son la conocida *Symbol* para letras griegas y símbolos matemáticos, *ZapfChancery* como letra “de fantasía” y *ZapfDingbats* como repertorio de símbolos no alfabéticos.

En Hasefroch, muchos de los tipos TrueType que se proporcionan con el sistema operativo o con aplicaciones como Office no hacen más que emular estos tipos Postscript.

<i>Tipo Postscript</i>	<i>Tipo Hasefroch</i>	<i>Características</i>
<i>AvantGarde</i>	<i>Century Gothic</i>	<i>Paloseco, 4 variedades</i>
<i>Bookman</i>	<i>Bookman Old Style</i>	<i>Romana, 4 variedades</i>
<i>Courier</i>	<i>Courier New</i>	<i>Monoespaciada (romana), 4 variedades</i>
<i>Helvetica</i>	<i>Arial</i>	<i>Paloseco, 4 variedades</i>
<i>HelveticaNarrow</i>	<i>Arial Narrow</i>	<i>Paloseco, 4 variedades</i>
<i>NewCenturySchoolbook</i>	<i>Century Schoolbook</i>	<i>Romana, 4 variedades</i>
<i>Palatino</i>	<i>Book Antiqua</i>	<i>Romana, 4 variedades</i>
<i>Symbol</i>	<i>Symbol</i>	<i>Romana (griega), 1 variedad</i>
<i>Times</i>	<i>Times New Roman</i>	<i>Romana, 4 variedades</i>
<i>ZapfChancery</i>	<i>Monotype Corsiva</i>	<i>Fantasía, 1 variedad</i>
<i>ZapfDingbats</i>	<i>Monotype Sorts</i>	<i>Símbolos, 1 variedad</i>

Tabla 3.1. Correspondencia de tipos Postscript y TrueType

Observamos entre otros los míticos nombres de *Arial*, *CourierNew*, *TimesNewRoman* o *Symbol*, que son los tipos básicos de Hasefroch. En la columna de “características”, cuando se dice “4 variedades”, nos referimos a que existen archivos TrueType para las variedades normal, cursiva, negrita y negrita-cursiva (lo que es imprescindible cuando un tipo se quiere usar para cuerpo de texto o incluso para títulos y demás), mientras que si hay una sola variedad, es que su propósito es más limitado, y su uso debería quedar restringido a esa variedad (así, es un error grave poner en cursiva o negrita el tipo *Symbol*, por ejemplo).

En resumen, si tuviera que decidir los tipos a utilizar por cuestiones meramente estilísticas, elegiría *Palatino* (*BookAntiqua*) para el cuerpo de texto, *Helvetica* (*Arial*) para títulos y demás, *Courier* (*CourierNew*) para programas, y usaría esporádicamente *Symbol* para letras griegas y símbolos matemáticos y *ZapfDingbats* (*MonotypeSorts*) para otros símbolos.

3.1.2 Tipos libres: OpenOffice.org

Ahora nos planteamos si existen versiones libres de estos u otros tipos de letra. Sin haber hecho un estudio demasiado exhaustivo, creo haber encontrado tres posibilidades principales, pero solo la última es una opción mínimamente válida.

En primer lugar, OpenOffice.org instala unos cuantos tipos, llamados *Vera* o *Bitstream Vera*, que creo que tienen una licencia libre (aunque no es exactamente la GPL, que yo sepa, por lo que habría que asegurar este tema). La *tabla 3.2* recoge esa lista de tipos.

<i>Tipo</i>	<i>Características</i>
<i>Vera Sans</i>	<i>Paloseco, 4 variedades</i>
<i>Vera Sans Mono</i>	<i>Monoespaciada, 4 variedades</i>
<i>Vera Serif</i>	<i>Romana, solo normal y negrita</i>

Tabla 3.2. Tipos proporcionados por OpenOffice.org

El problema principal es que no hay una letra romana válida (porque el tipo *Vera Serif* no aparece con las 4 variedades necesarias para poder ser usada como cuerpo de texto). Tampoco hay un tipo para letras griegas y símbolos matemáticos, lo cual es otra gran desventaja. Así que esta opción no resulta válida, a mi modo de ver, para desarrollar un proyecto escrito mínimamente decente.

3.1.3 Tipos libres: URW++ en CTAN

El sistema $\text{T}_{\text{E}}\text{X}$ dispone de un repertorio, a priori relativamente numeroso, de tipos libres. Lo que ocurre es que la mayoría usan el mecanismo propio de representación, llamado METAFONT. Como tipos de contorno se encuentran algunos tipos en el formato Type1 de Adobe. Este formato permite (como TrueType) escalar una misma descripción de un tipo a cualquier tamaño.

Hace años, una empresa de tipografía alemana, URW++ Design & Development (<http://www.urwpp.de>) liberó una versión Type1 de los 35 tipos Postscript recogidos en la primera columna de la *tabla 3.1*. La licencia es GPL, y se pueden recoger de cualquier depósito CTAN (*Comprehensive Tex Archive Network*), como por ejemplo:

<http://www.ctan.org/tex-archive/fonts/urw/base35/>

La lista de nombres de tipos aparece en la tabla siguiente:

<i>Tipo Postscript</i>	<i>Tipo URW (type1)</i>
<i>AvantGarde</i>	<i>URWGothicL</i>
<i>Bookman</i>	<i>URWBookmanL</i>
<i>Courier</i>	<i>NimbusMonL</i>
<i>Helvetica</i>	<i>NimbusSanL</i>
<i>HelveticaNarrow</i>	<i>NimbusSanL-Cond</i>
<i>NewCenturySchoolbook</i>	<i>CenturySchL</i>
<i>Palatino</i>	<i>URWPalladioL</i>
<i>Symbol</i>	<i>StandardSymL</i>
<i>Times</i>	<i>NimbusRomNo9L</i>
<i>ZapfChancery</i>	<i>URWChanceryL</i>
<i>ZapfDingbats</i>	<i>Dingbats</i>

Tabla 3.3. Correspondencia de tipos Postscript y URW

Esta podría ser una interesantísima opción, salvo por un pequeño detalle (estoy hablando irónicamente, se trata de un problema realmente grave): he comprobado que la exportación a PDF que incorpora OpenOffice para Hasefroch *no funciona* correctamente con estos tipos en formato Type1. Al exportar, el documento PDF resultante no contiene los tipos de letra que debería contener. Si la exportación se hace a través del Adobe Acrobat original, no hay ningún problema, los tipos salen perfectamente, así que se trata de un problema de la exportación a PDF del OpenOffice para Hasefroch.

Este problema parece no producirse exactamente igual en la versión de OpenOffice para GNU/Linux, pero las pocas pruebas que hemos realizado Nando Quintana y yo tampoco dan un resultado correcto (en uno de los casos, el tipo *StandardSymL* no salía bien en el PDF generado).

En resumen, esta podría ser una opción válida si no fuera por la generación a PDF. No me extrañaría nada que próximamente se pidiera entregar un PDF de la memoria junto con su versión impresa. Igualmente, si vais a imprimir en una imprenta o tienda de fotocopias, deberéis llevar un PDF, con lo que estáis pillados. Yo no usaría esta opción sin antes hacer una prueba con los tipos de letra elegidos.

3.1.4 Tipos libres: Freefont

El proyecto *Freefont* creo que no es oficialmente un proyecto GNU, pero aparece en el directorio de software libre de la FSF y está bajo la GPL. Su página principal es:

<http://www.nongnu.org/freefont/>

aunque el directorio para descargar está vacío (parece que han tenido algún problemilla este invierno); usad esta otra dirección:

<http://ftp.gnu.org/savannah/files/freefont/>

que contiene el archivo `freefont-ttf.zip` a descargar.

El objetivo del proyecto *Freefont* es crear un repertorio de tipos de letras escalables que cubran el estándar ISO 10646/Unicode UCS (Universal Character Set). Hasta hace poco, los tipos de letra escalables (Type1 o TrueType) cubrían como mucho el estándar ISO 8859, por lo que contenían alrededor de 250 signos en total. El advenimiento de Unicode ha obligado a los desarrolladores a crear tipos de letra que cubrieran, no todo el mapa de signos Unicode, pero sí al menos las partes más importantes, y eso mismo es lo que intenta el proyecto *Freefont* desde la perspectiva del *software* libre. En este aspecto, el proyecto ya cuenta con una gran ventaja, incluso sobre los tipos propietarios, dado que ha implementado más de 3.500 signos en alguno de los tipos.

Freefont proporciona tres tipos de letra, con las cuatro variedades necesarias, según se observa en la siguiente tabla.

<i>Tipo Freefont</i>	<i>Tipo similar</i>
<i>Free Monospaced</i>	<i>Courier</i>
<i>Free Sans</i>	<i>Helvetica (Arial)</i>
<i>Free Serif</i>	<i>Times</i>

Tabla 3.4. Tipos de letra del proyecto *Freefont*


En principio, por lo tanto, este puede ser un repertorio mínimo suficiente para las necesidades de un documento como el presente¹. Hay que tener en cuenta que no se necesita un tipo de letra como *Symbol* para letras griegas y signos matemáticos, porque se supone que estos son signos Unicode que están implementados en alguno de los otros tipos presentes.

Ahora bien, una vez más hay dos peros, yo ya no me lo puedo creer, pero es así. El primer problema parece afectar solamente a Hasefroch, pero recomiendo a los usuarios de GNU/Linux que vayan a usar estos tipos de letra que hagan también las pruebas necesarias. La cosa es que en Hasefroch, los tipos *Freefont* se ven de pena en la pantalla, vamos, el texto llega a ser casi ilegible a tamaño real. Las pruebas realizadas en varias máquinas con GNU/Linux parecen indicar que este sistema no presenta este problema, los tipos se ven de maravilla. Lo sorprendente es que solo es un problema de pantalla (en cualquier aplicación, luego, je, je ¿baja calidad del *reproductor* de TrueType en Hasefroch?), porque tanto el PDF generado por OpenOffice como la impresión del documento parecen de buena calidad.

Sin embargo, sí he detectado alguna cosa “rara” en la salida PDF del tipo de letra *FreeSans* en un Hasefroch NT: concretamente la negrita no sale tan “gorda” como se supone que debería (pero tampoco queda mal, estéticamente me refiero). Este problema parece que no se produce en GNU/Linux ni en Hasefroch 2000, al menos en los sistemas probados. Así que, insisto, lo mejor es hacer pruebas de generación de PDF y de impresión antes de dar la opción por válida.

El otro problema sí es más peliagudo (aunque avanzo que creo haber conseguido una solución) y afecta por igual a GNU/Linux y Hasefroch. Por alguna razón que se me escapa, los signos de los tipos *Freefont* ocupan en altura más o menos el doble que los signos de otros

¹ El tipo *Times* como letra base me resulta un poco “pobre”, pero esto es un criterio estético puramente personal, que puede ser menos prioritario que el disponer de un tipo de letra libre.

	Universidad de Deusto Facultad de Ingeniería	3. Explicamos el documento-plantilla	Proyecto Fin de Carrera
--	---	--------------------------------------	----------------------------

tipos similares al mismo tamaño, pero no porque el signo en sí sea más grande, sino porque tienen más espacio blanco por arriba y por abajo (técnicamente, el *ojo* es el mismo que en tipos similares, pero el *cuerpo* es doble). Esto implica que el texto parece tener interlineado doble, lo cual queda evidentemente de pena. Este problema con los tipos *Freefont* (que no ocurre solo con OpenOffice) ya apareció públicamente en mayo de 2003, como se puede ver en el mensaje de la lista de correo del proyecto (se adjunta copia local con la distribución de este documento), en la dirección:

<http://mail.gnu.org/archive/html/freefont-bugs/2003-05/msg00016.html>

pero no parece que se haya solucionado (de hecho, sin querer molestar a nadie, da la impresión de que el proyecto está parado).

La única cosa que se me ha ocurrido para intentar solucionarlo es modificar el interlineado de los estilos para que se vea como debe ser. Con los tipos de letra propietarios, el estilo base y sus derivados tienen interlineado sencillo. Con los tipos de letra *Freefont*, he tenido que colocar *interlineado fijo* a 0,48 cm para esos estilos, y practicar algunas modificaciones menores en otros estilos. La solución parece que funciona bien, pero lo cierto es que puede llegar a dar algún problema: en el interlineado fijo, la altura de la línea es fija; si se coloca entonces algún elemento más alto o más bajo (por ejemplo, un subíndice, un superíndice o un carácter más grande), puede llegar a verse truncado, según el caso.


En resumen, los tres tipos de letra *Freefont* suponen un encomiable esfuerzo para disponer de un repertorio libre de tipos escalables. La inclusión de un amplio conjunto de signos Unicode supera lo implementado en tipos de letra propietarios. Sin embargo, la visualización de los tipos en pantalla en Hasefroch es bastante lamentable, y que el cuerpo tenga un tamaño doble de lo habitual es un aspecto incomprensible. A pesar de ello, por lo visto en apartados anteriores, se trata a mi entender de la única opción libre mínimamente verosímil para OpenOffice.

3.1.5 En definitiva

Por lo explicado hasta ahora, considero necesario ofrecer dos alternativas del documento-plantilla, uno que usa fuentes propietarias de Hasefroch, y otro que usa las fuentes libres de *Freefont*, ambas en el formato TrueType. Las definiciones de estilos son ligeramente diferentes en cada caso, por lo que es importante utilizar el documento-plantilla correcto para cada caso. Elegid la opción que más se acomode a vuestros criterios personales, aunque me gustaría recomendar que, si elegís *Freefont*, hagáis una prueba de impresión y de generación de PDF, para asegurar que se cumplen los requisitos de calidad que plantea la memoria del proyecto fin de carrera.

La versión con tipos propietarios utiliza *Book Antiqua* para cuerpo de texto y derivados, *Arial* para títulos y similares y *Courier New* para programas y cosas así. La versión con los tipos *Freefont* utiliza respectivamente los tipos *Free Serif*, *Free Sans* y *Free Mono* para los mismos cometidos.

En cualquiera de los casos, elegir los tipos concretos es una cosa muy personal, así que no hay ningún problema en usar otros tipos de letra que os parezcan más adecuados. Actuando en “Herramientas/Opciones/Documento de texto/Fuentes básicas” hay posibilidad de cambiar los tipos básicos, pero por no sé qué razón, esto no se traslada a *todos* los estilos definidos, así que tendréis que recorrer algunos de ellos haciendo los cambios uno a uno. Si se decide usar

 <p>Universidad de Deusto</p> <p>Facultad de Ingeniería</p>	3.1. Tipos de letra	Proyecto Fin de Carrera
--	---------------------	----------------------------

otros tipos, es mejor partir de la versión de la plantilla preparada con tipos propietarios, que contiene la definición de estilos más correcta.

Sobre el tema de instalar y usar tipos de letra TrueType o Type1 en GNU/Linux, siguiendo las indicaciones dadas por aktor en la lista del *e-ghost* debería llegarse a un resultado correcto (al menos *lovelace* y yo lo hemos conseguido).

3.2 Formatos de párrafo

La primera cuestión a tener en cuenta para usar OpenOffice es el concepto de *estilo* (también en Hasecorp Word, por cierto). Un estilo agrupa una serie de formatos bajo un nombre, de forma que pueden aplicarse de una vez a cada trozo de nuestro texto. Lo más importante de los estilos es que facilitan enormemente el cambio de presentación de nuestro texto: solo hemos de actuar sobre el estilo concreto para practicar el cambio, en lugar de pasar por todos los sitios que tienen ese formato. Para ello, sin embargo, habría que ser muy estricto, y no aplicar nunca formato a través de los menús (“Formato/carácter”, “Formato/Párrafo”, *etc.*), sino siempre a través de estilos, que podemos ir creando según se necesiten.

OpenOffice proporciona un repertorio de estilos bastante rico. Por otro lado, hay estilos de *carácter* (solo formatos de carácter), de *párrafo*, de *marco*, de *página* y de *numeración*. Una plantilla es justamente un documento que no tiene contenido, pero sí todos los estilos necesarios para crearlo. En los apartados que siguen, se explican las decisiones particulares que se han tomado para los estilos de que consta este documento, y cómo usarlos adecuadamente.

Si este tema de los estilos te suena un poco a nuevo, convendría que empollases algo más sobre el asunto, o hicieras alguna prueba antes de empezar en serio. Por descontado, tener activo “Formato/estilista” es, ya desde ahora, una necesidad ineludible.


3.2.1 Cuerpo

El cuerpo del texto (este mismo párrafo, por ejemplo) va en un estilo que, de forma totalmente sorprendente, se llama “Cuerpo de texto”. Sus principales atributos son, por supuesto, la justificación del texto, el interlineado sencillo y el tipo de letra “serif”, y por supuesto², en esto nos separamos notablemente de la plantilla corporativa.

Hay que observar que cada párrafo empieza con una sangría de 1,25 cm. Este es un tema enormemente complejo, aunque no lo parezca *a priori*. Es importante mencionar en primer lugar que dicha sangría se consigue con el fabuloso método de empezar cada párrafo con un tabulador (por ciervo, quien escribe sin tener activado “Ver/caracteres no imprimibles” es un temerario): esta forma pedalística de obtener la sangría puede ser muy discutible, pero es la mía. En segundo lugar, está el tema de la sangría en sí: hay gente que opina que no es necesaria (también hay gente que lee todos los días el horóscopo). En tercer lugar, existe últimamente la costumbre de *no sangrar el primer párrafo* después de un título: en inglés, no lo discuto (en realidad, en esa lengua puedes cometer todo tipo de atropellos al buen gusto, como es sabido), pero opino que en español hay que sangrarlo igual que los demás.

Relacionado con lo anterior, este estilo aplica una pequeña separación entre párrafos a base de añadir un espacio posterior. Aunque este espacio no es estrictamente necesario (porque

² ¿Dos “por supuesto” en una misma frase? Es intencionado, por supuesto.

	Universidad de Deusto Facultad de Ingeniería	3. Explicamos el documento-plantilla	Proyecto Fin de Carrera
--	---	--------------------------------------	----------------------------

la propia sangría permite visualizar bastante bien la separación), he considerado interesante mantenerlo.

3.2.2 Títulos

Los títulos de las diferentes partes del cuerpo principal de nuestra memoria deberán seguir los estilos predefinidos “Encabezado 1”, “Encabezado 2” y “Encabezado 3”: gracias a ello, muchas otras cosas del documento se resuelven de forma automática (cosa a la que no siempre se le da la importancia debida). He elegido una presentación concreta para estos estilos que, una vez más, se aparta bastante de la plantilla corporativa, pero que me parece mucho más vistosa. En cualquier caso, cambiar estos estilos es bien fácil.

Por supuesto, la numeración de títulos a través de los estilos señalados es automática (pero no veas lo que cuesta hacerlo). Por otro lado, no he preparado más que tres niveles de profundidad para los títulos (es decir, el estilo “Encabezado 4” no está tratado). Si esta profundidad te parece poco, mi más sincera recomendación es que intentes ordenar tus ideas.

Ahora bien, he dicho que estos estilos han de usarse para el cuerpo principal de la memoria: llamo a esto todo lo que va entre el “Capítulo 1” (incluido) y el capítulo de conclusiones (también incluido). Otras partes del documento usarán otros estilos de título diferentes (aunque con una presentación similar): permanece atento a la ventana de estilos, aunque comentaré algunos casos más adelante.

3.2.3 Listas numeradas y no numeradas


A pesar de que son elementos muy habituales, no he podido/querido meterme en este asunto. En Word, esta es precisamente una de las mayores pesadillas para las personas con un mínimo sentido del orden natural. Desconozco si OpenOffice.org ofrece una solución razonable al asunto.

3.3 Elementos flotantes

Son las figuras, tablas, ecuaciones, trozos de código, *etc.* Desde un punto de vista más bien intuitivo, distingo dos casos: los que tienen poca altura (una ecuación de una o dos líneas, por ejemplo) y los que tienen una altura significativa (equivalente a varias líneas de texto). Los primeros pueden resolverse con estilos de párrafo *ad hoc* (he preparado únicamente el estilo “Línea suelta centrada” con este propósito). Los segundos son más peliagudos, porque como se sabe, si los dejáramos tal cual en su lugar, harían que las páginas estuvieran incompletas (con una gran zona blanca al final), y eso es feo como pocas cosas.

Hay otro criterio para distinguir estos elementos: saber si deben llevar una *etiqueta* o no (es decir, algo como “Figura 1. Patatín y patatán”). El caso paradigmático de las ecuaciones se resuelve desde tiempos inmemoriales con números correlativos entre paréntesis colocados a la derecha (aunque también los he visto a la izquierda). Para los demás casos, considero que *todos* los elementos han de llevar su etiqueta (en el ejemplo de Borja de L^AT_EX hay un trozo de programa que no lleva etiqueta: a mí entender debe llevarla como las figuras o las tablas).

Opino que la etiqueta ha de incluir el número de capítulo (o sea, “Figura 2.1. Patatín y patatán”, donde el segundo dígito es correlativo). Opino que debe estar situada debajo del

 <p>Universidad de Deusto</p> <p>Facultad de Ingeniería</p>	3.3. Elementos flotantes	Proyecto Fin de Carrera
--	--------------------------	----------------------------

objeto que nombran (aunque hay cierta antigua tradición de ponerlo encima en las tablas). Por lo tanto, llamaré *caja* al conjunto de objeto (figura, tabla o lo que sea) y su etiqueta.

3.3.1 Marcos flotantes

La solución al problema planteado en primer lugar (que no quede una gran zona blanca en la página) consiste en hacer la caja *flotante*. Aquí, como en muchas otras cosas realmente, L^AT_EX gana por goleada (pero de baloncesto, o sea, 111-0). Tanto Word como OpenOffice.org ofrecen el concepto de *marco*, pero se trata de otra gran pesadilla (en Word seguro, pero con lo poco que he podido experimentar en OpenOffice.org, no se trata de una mejor solución la que ofrece).

Un marco, además de un tamaño y un modo de ajuste, está definido por una posición en la página y por un *anclaje*: puede ser a un párrafo, a la página entera o como carácter (esto significa más bien que no hay anclaje). Total, que a mí me ha resultado totalmente imposible anclar un marco a un párrafo y que esté situado en la parte inferior de la página. Esto último se consigue sin problemas si anclamos el marco a la página, pero tiene el inconveniente de que si se añade texto por delante, el marco no se mueve.

Mi solución para este problema es claramente una chapuza, pero es lo único que se me ocurre. Primero, hacemos un marco anclado a un párrafo (normalmente, el que contiene la referencia a ese objeto o alguno posterior en todo caso). El estilo de marco llamado “Marco” sirve para eso, le doy un tamaño horizontal que llena toda la parte impresa de la página (15 cm), y ajuste “Ninguno”.


Segundo, inserto el objeto y la etiqueta. Si es una tabla, “Insertar/tabla”, si es una figura, pegando desde algún archivo o lo que sea (esto crea a su vez otro marco, pero este lo pongo con estilo de marco “Imagen con figura”, que es sin anclaje). Para insertar la etiqueta, uso “Insertar/etiqueta” y elijo el tipo de etiqueta que le corresponde, creándose esta en una línea nueva debajo del objeto. Si se elige una tabla, el formato de párrafo de la etiqueta se coloca automáticamente como “Tabla”, si es una figura, será “Figura” (y así para otros nombres que se puedan crear).

El resultado es el de la *tabla 2.1* o la *figura 2.1*. Como se ve, realmente esos marcos no están bien situados y no son del todo flotantes: podemos moverlos un poco, pero si los movemos demasiado se “desanclan” del párrafo al que se refieren y se anclan al anterior o posterior. El paso final es colocarlos en la parte inferior de la página (a mí es lo que más me gusta, pero pueden colocarse al principio o en el medio, en esto no hay problema). Para ello, habrá que ir marco a marco, anclándolo a la página y situándolo donde nos parezca bien. De esa forma, quedan tal como se ve realmente en el documento-plantilla.

Como esto hace que los marcos no salten automáticamente de página si añadimos texto por delante, recomiendo hacer esto solo cuando hayamos acabado definitivamente de escribir nuestra memoria. Así queda la . ¡Suerte! (la vais a necesitar).

3.3.2 Referencias en el texto

Las referencias a los elementos flotantes (del tipo “como se ve en la figura 2.1”) se consiguen automáticamente con “Insertar/referencia”, pestaña “Referencias”, eligiendo a la derecha “Categoría y número”. Yo uso un estilo de carácter llamado “Referencia” para formatearla. Recomendación: activa cuanto antes la opción “Ver/Marcas”; de esa forma

 <p>Universidad de Deusto</p> <p>Facultad de Ingeniería</p>	3. Explicamos el documento-plantilla	Proyecto Fin de Carrera
--	--------------------------------------	----------------------------

aparecen sombreados en gris los textos que se obtienen automáticamente (números de títulos, índices, referencias). Pero no te asustes: ¡ese sombreado no sale al imprimir!

Vamos a ver, que no sé si nos estamos enterando. Todo esto que estamos haciendo con títulos, figuras, tablas y demás tiene un solo propósito: facilitar nuestro trabajo. Supuesto: has escrito más o menos la mitad de la memoria, y de repente te das cuenta de que tienes que insertar un capítulo en medio de dos que ya tenías casi terminados. Si has escrito tú explícitamente los números de capítulo, de apartado, de figuras, de tablas, referencias a los mismos, *etc.* imagina el trabajo que te queda. Con este documento, siguiendo las indicaciones ofrecidas, solo tienes que insertar el nuevo capítulo, y nada más, todas las numeraciones, referencias, *etc.* se actualizan automáticamente (a veces, la actualización no es inmediata: adivina para qué sirve la tecla F9).

3.4 Partes del documento (formato de página)

La memoria consta de un único fichero, eso queda claro. Hay diferentes partes (portada, resumen, índices, cuerpo principal, bibliografía y anexos) que se han obtenido aplicando *estilos de página*. Para incluir una nueva parte que alguien necesite solo hay que aplicar “insertar/salto manual/salto de página”, eligiendo normalmente el estilo “[xxx]primera página”, dependiendo de la parte en que estemos (más adelante explicamos los diferentes casos).


Un estilo de página define el tamaño de las páginas (en esto seguimos al pie de la letra la plantilla oficial), su colocación (si la impresión será a dos caras, lo que presupongo), la presencia de encabezamiento y pie (aquí también pretendo seguir la plantilla original, pese a que, lo digo sin ambages, me parece horrible), y la numeración de las páginas.

Empezaré explicando el caso “normal” (el del cuerpo de la memoria), para entender luego los problemas que plantean las demás partes y cómo se han solucionado.

3.4.1 Cuerpo de la memoria

Esta parte es tradicionalmente la que empieza con el “Capítulo 1” de nuestro escrito, y suele llevar el número de página “1” (las páginas precedentes se numeran con números romanos). Se supone que las páginas van a dos caras (izquierda y derecha), con sus encabezamientos y pies correspondientes (realmente esto vale para toda la memoria salvo la hoja de portada).

Entre muchas posibilidades, para seguir aproximadamente la plantilla oficial, se ha escogido que la página izquierda (o de número de página par) contenga el título del capítulo en el encabezamiento y el número de página en el pie (colocado en el extremo, y por tanto, a la izquierda), mientras que la página derecha (o impar) contiene el título del subcapítulo en el encabezamiento y el número de página en el pie (por la misma razón que antes, esta vez a la derecha). Estas características se han recogido en los estilos de página llamados “Página izquierda” y “Página derecha”, y se aplican automáticamente según cambiamos de página (si estamos en una página derecha y pasamos a una nueva, llevará el estilo de la página izquierda y así sucesivamente). El hecho de que aparezca automáticamente el título del capítulo o del subcapítulo se debe a que se han usado los estilos “Encabezado 1” y “Encabezado 2” para ellos, colocándose en el encabezamiento una referencia a los mismos.

 <p>Universidad de Deusto</p> <p>Facultad de Ingeniería</p>	3.4. Partes del documento (formato de página)	Proyecto Fin de Carrera
--	---	----------------------------

Cualquier capítulo ha de empezar en una página derecha o impar. Dado esto, no habría ninguna pega en empezar con una página de estilo “Página derecha”, salvo por el hecho de que el título del capítulo (que será lógicamente lo primero que contendrá la página) aparecerá redundantemente en el encabezamiento (por cierto, Borja, tu plantilla de L^AT_EX ¿qué tal resuelve esto?). Para evitar este pequeño problema, hemos creado un estilo de página llamado “Primera página”, haciéndola en casi todo similar a la derecha, por supuesto, salvo por el detalle de que su encabezamiento contiene ¡el título del proyecto!

Como este título va a aparecer en muchos lugares, lo más adecuado es escribirlo una sola vez. Para ello, se ha recogido en una hoja al final de la memoria (no te olvides de que no hay que imprimirla) ese título, y se ha asociado a lo que se llama una “marca de texto”, de nombre “Título_del_proyecto”. En los encabezamientos que deben llevar el título, se ha colocado un “campo” (¿cómo?, ¿todavía no sabes lo que es un campo?) que simplemente inserta el contenido de la marca descrita. O sea, solo tienes que ir a esa última página, escribir el nombre de tu proyecto entre los corchetes, y *voilà*.

En definitiva, para empezar un nuevo capítulo del cuerpo de la memoria, nos colocamos al final del ya escrito, y aplicamos “Insertar/salto manual/salto de página”, eligiendo el estilo “Primera página”. Las demás páginas, según vayan surgiendo, llevarán alternativamente los estilos “Página izquierda” y “Página derecha”.

3.4.2 Portada

La primera página tiene estilo de página “Portada”. Se supone que es la numerada i (1 en romana), aunque no se muestra esa numeración.


En cuanto al contenido, solo hay que sustituir lo que va en negrita por los datos personales de cada uno. El estilo de párrafo es “Portada base”, mientras que los datos en negrita van en el estilo de carácter llamado “Portada base énfasis”.

3.4.3 Resumen e índices

Las dos partes siguientes son la página de resumen y descriptores (se supone que ocupa una sola página) y la parte de índices. Ambas siguen los estilos “Prólogo primera página”, “Prólogo página izquierda” y “Prólogo página derecha”, que son básicamente los mismos estilos de página del cuerpo de la memoria, salvo por el hecho de que los encabezamientos contienen otro texto (he colocado el título del proyecto no solo en la primera página sino en todas las demás), y la numeración de páginas es en romana. Si se necesita alguna otra parte de prólogo más (con numeración romana), solo hay que “Insertar/salto manual/salto de página”, eligiendo el estilo “Prólogo primera página”.

En cuanto al contenido propiamente dicho, la parte de resumen está bastante clara, creo yo. En la de índices he creado en primer lugar y como es lógico la tabla de contenido principal (solo hay que actualizarla, con el botón derecho, según se vaya añadiendo contenido a la memoria), y luego unas listas de figuras y tablas (estas no quedan muy a mi gusto, pero no he conseguido hacerlo mejor, creo que OpenOffice podría mejorar en este tema).

Para que esto funcione, es estrictamente necesario que los títulos de capítulos, subcapítulos y demás se hayan escrito con los estilos “Encabezado 1”, “Encabezado 2” y “Encabezado 3”, mientras que las etiquetas (o pies) de tablas y figuras han de llevar los estilos “Tabla” y “Figura”, respectivamente.

	Universidad de Deusto Facultad de Ingeniería	3. Explicamos el documento-plantilla	Proyecto Fin de Carrera
--	---	--------------------------------------	----------------------------

3.4.4 Bibliografía

Es una parte fundamental de cualquier trabajo científico-técnico. Suele ponerse después de las conclusiones (a pesar de que la memoria, como veremos, puede no acabar ahí). Supongo que ese capítulo de conclusiones sí va numerado como los demás del cuerpo del documento.

La dificultad que tiene este *pseudocapítulo* para la edición de textos es que, en efecto, se trata de un “pseudocapítulo”: no es muy correcto seguir en él la numeración de los capítulos “de verdad” (es decir, no queremos que ponga “Capítulo X. Bibliografía”). Ello obliga, primero, a disponer un estilo propio de título (que proporciona OpenOffice, llamado “Encabezado de bibliografía”, y que nosotros hemos hecho similar en todo a “Encabezado 1”, salvo por la numeración).

Segundo, obliga a incluir una orden especial en el índice, porque ahí sí tiene que aparecer (para el que esté interesado en aprender, aplicar “Editar índice” en el menú contextual del índice y observar que en la pestaña “Índice” está marcado “Estilos”; pulsar la tecla “...” y ver que el estilo “Encabezado de bibliografía” está en la columna del nivel 1: así se consigue).

Tercero, como los encabezamientos de página llevan los títulos de capítulos y subcapítulos, también hay que crear estilos propios de página (“Bibliografía primera página”, “Bibliografía página izquierda” y “Bibliografía página derecha”). ¿Alguien se imaginaba que hacer una memoria llevara tanto curro? Pues todavía queda un poco.


3.4.5 Anexos

No todas las memorias han de llevarlos, pero si alguien los necesita, aquí van las indicaciones. Los problemas que plantean (hay dos anexos al final para que se vea cómo opino que deben quedar) son similares a los de la parte de bibliografía, pero multiplicando por 10 (estos problemas los gestiona L^AT_EX de forma mucho más simple, una vez más). De todos modos, gracias a un mecanismo que, humildemente y salvo que alguien me informe de lo contrario, creo haber inventado, se resuelven todos ellos.

Los anexos son capítulos que han de estar numerados, pero no como los del cuerpo del documento, sino en la forma “Anexo A”, “Anexo B” y así sucesivamente. OpenOffice, como Hasecorp Word, proporciona un mecanismo uniforme de numeración de los títulos de un documento, pero no permite modificar ese mecanismo en cualquier lugar intermedio del mismo. Mi solución (que llevo años aplicando en Word) consiste en utilizar niveles normalmente “inútiles” del esquema de títulos para “simular” esa nueva numeración, sin que afecte a la anterior. Como OpenOffice proporciona 10 niveles, y opino que a partir del cuarto son innecesarios, simulo la numeración de los anexos con los títulos de nivel 6 en adelante. Perdonad, pero ¿no os parece una idea brillante?

Total, los títulos de los anexos usan los estilos “Encabezado 6”, “Encabezado 7” y “Encabezado 8”, que por lo demás, se han hecho similares en aspecto a “Encabezado 1”, “Encabezado 7” y “Encabezado 8”. Luego, se actúa en “Herramientas/Numeración de capítulos”, y luego en el índice para elaborar la nueva tabla de contenido.

Pero como estos títulos son distintos a los del cuerpo, hay que crear estilos de página nuevos, que se llaman “Anexo primera página”, “Anexo página izquierda” y “Anexo página derecha”. Si se necesita un nuevo anexo, solo hay que ponerse después del último, y aplicar “Insertar/salto manual/salto de página”, eligiendo el estilo “Anexo primera página”.

	Universidad de Deusto Facultad de Ingeniería	La memoria del proyecto fin de carrera en OpenOffice.org: how-to & why-to	Proyecto Fin de Carrera
---	---	--	----------------------------

Capítulo 4. Conclusiones


La impresión general acerca de OpenOffice.org ha sido relativamente buena. Por un lado, se trata de un sistema ciertamente complejo, que habrá supuesto un esfuerzo considerable, y que se encuentre en el dominio del *software* libre es algo que no debe olvidarse.

Solamente he utilizado la herramienta de procesamiento de textos, pero haciendo una extrapolación, puedo decir que el sistema cumple con creces los requisitos mínimos para ser una alternativa creíble a Hasecorp Office, y aunque hay aspectos que podrían mejorarse, indudablemente, son más bien detalles y no elementos críticos de funcionamiento.

En lo que se refiere particularmente a la edición de textos, lo que menos me ha gustado es el seguimiento de la filosofía general de Hasecorp Word en muchos aspectos. Hubiera deseado una revisión más profunda del concepto de edición de textos (orientada a la preparación avanzada de documentos como este, no a la escritura de cartas y avisos), que recogiera aspectos de L^AT_EX, y que facilitara muchos de los temas que, por ejemplo, me he visto obligado a desarrollar en el capítulo anterior. Se han hecho indudables intentos para ello, y algunos problemas se resuelven más sencillamente que en Word (estilos de página y de marco, por ejemplo), pero son detalles bajo una filosofía general que me parece “poco estructurada”, por decir algo.


El aspecto que más me ha sorprendido, y esta vez de forma positiva quiero decir, es el almacenamiento de los documentos. En realidad, un documento de texto se trata de un archivo comprimido ZIP, en cuyo interior existen varios archivos XML, por un lado, y archivos gráficos para las imágenes, por otro. Entiendo que esta es otra forma de entender el concepto de “código abierto”, que tiene unas ventajas extraordinarias, no solo ya frente a formatos absolutamente cerrados como el usado por el propio Word, sino frente a otros formatos supuestamente “más abiertos” (como PDF o RTF).

El almacenamiento en formato XML permite, en primer lugar, un control de bajo nivel sobre los contenidos y formatos que es de agradecer para cualquier persona con inquietudes (desde luego, para una persona como yo, que cree que la *programación*, en un sentido amplio de la palabra, es uno de los grandes inventos de la Humanidad, pues eso una bomba). En segundo lugar, permite plantear una reutilización de tus contenidos (por ejemplo, pasarlos a otros formatos) infinitamente menos costosa e infinitamente más automatizable que con cualquier otro modo de almacenamiento. Finalmente, permite esto mismo, pero a nivel industrial, no ya personal: en nuestro campo de investigación por ejemplo, es importante la recogida automática de cantidades ingentes de textos. Hasta ahora, PDF, Word o RTF eran

	Universidad de Deusto Facultad de Ingeniería	4. Conclusiones	Proyecto Fin de Carrera
--	---	-----------------	----------------------------

formatos excluidos, porque no disponemos de los recursos necesarios para desarrollar un filtro para dichos formatos. Sin embargo, sí es factible construir un filtro para XML, y de ahí a poder recuperar textos de OpenOffice.org hay un paso.

En definitiva, a pesar de los inconvenientes mencionados y de las carencias en algunos detalles, lo cierto es que OpenOffice.org es un sistema al que hay que aplaudir.

	Universidad de Deusto Facultad de Ingeniería	La memoria del proyecto fin de carrera en OpenOffice.org: how-to & why-to	Proyecto Fin de Carrera
---	---	--	----------------------------


Bibliografía

Díaz Labrador, JosuKa [1994] *Teoría de autómatas y lenguajes formales*, 4ª edición, Librería Enseñanza Católica.

Donnelly, Charles y Stallman, Richard [1999] *Bison. The YACC-compatible parser generator*. 14 January 1999, Bison Version 1.28, Free Software Foundation [ISBN: 1-882114-45-0].

Free Software Foundation [online]. URL: <http://www.fsf.org/>.

Martínez de Sousa, José [1999] *Manual de edición y autoedición*, Pirámide, Madrid [ISBN: 84-368-0840-1].

	Universidad de Deusto Facultad de Ingeniería	La memoria del proyecto fin de carrera en OpenOffice.org: how-to & why-to	Proyecto Fin de Carrera
---	---	--	----------------------------

Anexo A. GNU Free Documentation License

Version 1.2, November 2002

Copyright (C) 2000,2001,2002 Free Software Foundation, Inc.

59 Temple Place, Suite 330, Boston, MA 02111-1307 USA

Everyone is permitted to copy and distribute verbatim copies of this license document, but changing it is not allowed.

0. PREAMBLE


The purpose of this License is to make a manual, textbook, or other functional and useful document "free" in the sense of freedom: to assure everyone the effective freedom to copy and redistribute it, with or without modifying it, either commercially or noncommercially. Secondly, this License preserves for the author and publisher a way to get credit for their work, while not being considered responsible for modifications made by others.

This License is a kind of "copyleft", which means that derivative works of the document must themselves be free in the same sense. It complements the GNU General Public License, which is a copyleft license designed for free software.

We have designed this License in order to use it for manuals for free software, because free software needs free documentation: a free program should come with manuals providing the same freedoms that the software does. But this License is not limited to software manuals; it can be used for any textual work, regardless of subject matter or whether it is published as a printed book. We recommend this License principally for works whose purpose is instruction or reference.

1. APPLICABILITY AND DEFINITIONS

This License applies to any manual or other work, in any medium, that contains a notice placed by the copyright holder saying it can be distributed under the terms of this License. Such a notice grants a world-wide, royalty-free license, unlimited in duration, to use that work under the conditions stated herein. The "Document", below, refers to any such manual or work. Any

	Universidad de Deusto Facultad de Ingeniería	A. GNU Free Documentation License	Proyecto Fin de Carrera
--	---	-----------------------------------	----------------------------

member of the public is a licensee, and is addressed as "you". You accept the license if you copy, modify or distribute the work in a way requiring permission under copyright law.

A "Modified Version" of the Document means any work containing the Document or a portion of it, either copied verbatim, or with modifications and/or translated into another language.

A "Secondary Section" is a named appendix or a front-matter section of the Document that deals exclusively with the relationship of the publishers or authors of the Document to the Document's overall subject (or to related matters) and contains nothing that could fall directly within that overall subject. (Thus, if the Document is in part a textbook of mathematics, a Secondary Section may not explain any mathematics.) The relationship could be a matter of historical connection with the subject or with related matters, or of legal, commercial, philosophical, ethical or political position regarding them.

The "Invariant Sections" are certain Secondary Sections whose titles are designated, as being those of Invariant Sections, in the notice that says that the Document is released under this License. If a section does not fit the above definition of Secondary then it is not allowed to be designated as Invariant. The Document may contain zero Invariant Sections. If the Document does not identify any Invariant Sections then there are none.


The "Cover Texts" are certain short passages of text that are listed, as Front-Cover Texts or Back-Cover Texts, in the notice that says that the Document is released under this License. A Front-Cover Text may be at most 5 words, and a Back-Cover Text may be at most 25 words.

A "Transparent" copy of the Document means a machine-readable copy, represented in a format whose specification is available to the general public, that is suitable for revising the document straightforwardly with generic text editors or (for images composed of pixels) generic paint programs or (for drawings) some widely available drawing editor, and that is suitable for input to text formatters or for automatic translation to a variety of formats suitable for input to text formatters. A copy made in an otherwise Transparent file format whose markup, or absence of markup, has been arranged to thwart or discourage subsequent modification by readers is not Transparent. An image format is not Transparent if used for any substantial amount of text. A copy that is not "Transparent" is called "Opaque".

Examples of suitable formats for Transparent copies include plain ASCII without markup, Texinfo input format, LaTeX input format, SGML or XML using a publicly available DTD, and standard-conforming simple HTML, PostScript or PDF designed for human modification. Examples of transparent image formats include PNG, XCF and JPG. Opaque formats include proprietary formats that can be read and edited only by proprietary word processors, SGML or XML for which the DTD and/or processing tools are not generally available, and the machine-generated HTML, PostScript or PDF produced by some word processors for output purposes only.

The "Title Page" means, for a printed book, the title page itself, plus such following pages as are needed to hold, legibly, the material this License requires to appear in the title page. For works in formats which do not have any title page as such, "Title Page" means the text near the most prominent appearance of the work's title, preceding the beginning of the body of the text.

A section "Entitled XYZ" means a named subunit of the Document whose title either is precisely XYZ or contains XYZ in parentheses following text that translates XYZ in another language. (Here XYZ stands for a specific section name mentioned below, such as "Acknowledgements", "Dedications", "Endorsements", or "History".) To "Preserve the Title" of such a section when you modify the Document means that it remains a section "Entitled XYZ" according to this definition.

 <p>Universidad de Deusto</p> <p>Facultad de Ingeniería</p>	<p>A. GNU Free Documentation License</p>	<p>Proyecto Fin de Carrera</p>
--	--	------------------------------------

The Document may include Warranty Disclaimers next to the notice which states that this License applies to the Document. These Warranty Disclaimers are considered to be included by reference in this License, but only as regards disclaiming warranties: any other implication that these Warranty Disclaimers may have is void and has no effect on the meaning of this License.

2. VERBATIM COPYING

You may copy and distribute the Document in any medium, either commercially or noncommercially, provided that this License, the copyright notices, and the license notice saying this License applies to the Document are reproduced in all copies, and that you add no other conditions whatsoever to those of this License. You may not use technical measures to obstruct or control the reading or further copying of the copies you make or distribute. However, you may accept compensation in exchange for copies. If you distribute a large enough number of copies you must also follow the conditions in section 3.

You may also lend copies, under the same conditions stated above, and you may publicly display copies.


3. COPYING IN QUANTITY

If you publish printed copies (or copies in media that commonly have printed covers) of the Document, numbering more than 100, and the Document's license notice requires Cover Texts, you must enclose the copies in covers that carry, clearly and legibly, all these Cover Texts: Front-Cover Texts on the front cover, and Back-Cover Texts on the back cover. Both covers must also clearly and legibly identify you as the publisher of these copies. The front cover must present the full title with all words of the title equally prominent and visible. You may add other material on the covers in addition. Copying with changes limited to the covers, as long as they preserve the title of the Document and satisfy these conditions, can be treated as verbatim copying in other respects.

If the required texts for either cover are too voluminous to fit legibly, you should put the first ones listed (as many as fit reasonably) on the actual cover, and continue the rest onto adjacent pages.

If you publish or distribute Opaque copies of the Document numbering more than 100, you must either include a machine-readable Transparent copy along with each Opaque copy, or state in or with each Opaque copy a computer-network location from which the general network-using public has access to download using public-standard network protocols a complete Transparent copy of the Document, free of added material. If you use the latter option, you must take reasonably prudent steps, when you begin distribution of Opaque copies in quantity, to ensure that this Transparent copy will remain thus accessible at the stated location until at least one year after the last time you distribute an Opaque copy (directly or through your agents or retailers) of that edition to the public.

It is requested, but not required, that you contact the authors of the Document well before redistributing any large number of copies, to give them a chance to provide you with an updated version of the Document.

	Universidad de Deusto Facultad de Ingeniería	A. GNU Free Documentation License	Proyecto Fin de Carrera
--	---	-----------------------------------	----------------------------

4. MODIFICATIONS

You may copy and distribute a Modified Version of the Document under the conditions of sections 2 and 3 above, provided that you release the Modified Version under precisely this License, with the Modified Version filling the role of the Document, thus licensing distribution and modification of the Modified Version to whoever possesses a copy of it. In addition, you must do these things in the Modified Version:

A. Use in the Title Page (and on the covers, if any) a title distinct from that of the Document, and from those of previous versions (which should, if there were any, be listed in the History section of the Document). You may use the same title as a previous version if the original publisher of that version gives permission.

B. List on the Title Page, as authors, one or more persons or entities responsible for authorship of the modifications in the Modified Version, together with at least five of the principal authors of the Document (all of its principal authors, if it has fewer than five), unless they release you from this requirement.

C. State on the Title page the name of the publisher of the Modified Version, as the publisher.

D. Preserve all the copyright notices of the Document.

E. Add an appropriate copyright notice for your modifications adjacent to the other copyright notices.

F. Include, immediately after the copyright notices, a license notice giving the public permission to use the Modified Version under the terms of this License, in the form shown in the Addendum below.

G. Preserve in that license notice the full lists of Invariant Sections and required Cover Texts given in the Document's license notice.

H. Include an unaltered copy of this License.

I. Preserve the section Entitled "History", Preserve its Title, and add to it an item stating at least the title, year, new authors, and publisher of the Modified Version as given on the Title Page. If there is no section Entitled "History" in the Document, create one stating the title, year, authors, and publisher of the Document as given on its Title Page, then add an item describing the Modified Version as stated in the previous sentence.

J. Preserve the network location, if any, given in the Document for public access to a Transparent copy of the Document, and likewise the network locations given in the Document for previous versions it was based on. These may be placed in the "History" section. You may omit a network location for a work that was published at least four years before the Document itself, or if the original publisher of the version it refers to gives permission.


K. For any section Entitled "Acknowledgements" or "Dedications", Preserve the Title of the section, and preserve in the section all the substance and tone of each of the contributor acknowledgements and/or dedications given therein.

L. Preserve all the Invariant Sections of the Document, unaltered in their text and in their titles. Section numbers or the equivalent are not considered part of the section titles.

M. Delete any section Entitled "Endorsements". Such a section may not be included in the Modified Version.

N. Do not retitle any existing section to be Entitled "Endorsements" or to conflict in title with any Invariant Section.

O. Preserve any Warranty Disclaimers.

 <p>Universidad de Deusto</p> <p>Facultad de Ingeniería</p>	<p>A. GNU Free Documentation License</p>	<p>Proyecto Fin de Carrera</p>
--	--	------------------------------------

If the Modified Version includes new front-matter sections or appendices that qualify as Secondary Sections and contain no material copied from the Document, you may at your option designate some or all of these sections as invariant. To do this, add their titles to the list of Invariant Sections in the Modified Version's license notice. These titles must be distinct from any other section titles.

You may add a section Entitled "Endorsements", provided it contains nothing but endorsements of your Modified Version by various parties--for example, statements of peer review or that the text has been approved by an organization as the authoritative definition of a standard.

You may add a passage of up to five words as a Front-Cover Text, and a passage of up to 25 words as a Back-Cover Text, to the end of the list of Cover Texts in the Modified Version. Only one passage of Front-Cover Text and one of Back-Cover Text may be added by (or through arrangements made by) any one entity. If the Document already includes a cover text for the same cover, previously added by you or by arrangement made by the same entity you are acting on behalf of, you may not add another; but you may replace the old one, on explicit permission from the previous publisher that added the old one.

The author(s) and publisher(s) of the Document do not by this License give permission to use their names for publicity for or to assert or imply endorsement of any Modified Version.

5. COMBINING DOCUMENTS

You may combine the Document with other documents released under this License, under the terms defined in section 4 above for modified versions, provided that you include in the combination all of the Invariant Sections of all of the original documents, unmodified, and list them all as Invariant Sections of your combined work in its license notice, and that you preserve all their Warranty Disclaimers.


The combined work need only contain one copy of this License, and multiple identical Invariant Sections may be replaced with a single copy. If there are multiple Invariant Sections with the same name but different contents, make the title of each such section unique by adding at the end of it, in parentheses, the name of the original author or publisher of that section if known, or else a unique number. Make the same adjustment to the section titles in the list of Invariant Sections in the license notice of the combined work.

In the combination, you must combine any sections Entitled "History" in the various original documents, forming one section Entitled "History"; likewise combine any sections Entitled "Acknowledgements", and any sections Entitled "Dedications". You must delete all sections Entitled "Endorsements."

6. COLLECTIONS OF DOCUMENTS

You may make a collection consisting of the Document and other documents released under this License, and replace the individual copies of this License in the various documents with a single copy that is included in the collection, provided that you follow the rules of this License for verbatim copying of each of the documents in all other respects.

You may extract a single document from such a collection, and distribute it individually under this License, provided you insert a copy of this License into the extracted document, and follow this License in all other respects regarding verbatim copying of that document.

	Universidad de Deusto Facultad de Ingeniería	A. GNU Free Documentation License	Proyecto Fin de Carrera
--	---	-----------------------------------	----------------------------

7. AGGREGATION WITH INDEPENDENT WORKS

A compilation of the Document or its derivatives with other separate and independent documents or works, in or on a volume of a storage or distribution medium, is called an "aggregate" if the copyright resulting from the compilation is not used to limit the legal rights of the compilation's users beyond what the individual works permit. When the Document is included in an aggregate, this License does not apply to the other works in the aggregate which are not themselves derivative works of the Document.

If the Cover Text requirement of section 3 is applicable to these copies of the Document, then if the Document is less than one half of the entire aggregate, the Document's Cover Texts may be placed on covers that bracket the Document within the aggregate, or the electronic equivalent of covers if the Document is in electronic form. Otherwise they must appear on printed covers that bracket the whole aggregate.

8. TRANSLATION

Translation is considered a kind of modification, so you may distribute translations of the Document under the terms of section 4. Replacing Invariant Sections with translations requires special permission from their copyright holders, but you may include translations of some or all Invariant Sections in addition to the original versions of these Invariant Sections. You may include a translation of this License, and all the license notices in the Document, and any Warranty Disclaimers, provided that you also include the original English version of this License and the original versions of those notices and disclaimers. In case of a disagreement between the translation and the original version of this License or a notice or disclaimer, the original version will prevail.

If a section in the Document is Entitled "Acknowledgements", "Dedications", or "History", the requirement (section 4) to Preserve its Title (section 1) will typically require changing the actual title.


9. TERMINATION

You may not copy, modify, sublicense, or distribute the Document except as expressly provided for under this License. Any other attempt to copy, modify, sublicense or distribute the Document is void, and will automatically terminate your rights under this License. However, parties who have received copies, or rights, from you under this License will not have their licenses terminated so long as such parties remain in full compliance.

10. FUTURE REVISIONS OF THIS LICENSE

The Free Software Foundation may publish new, revised versions of the GNU Free Documentation License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns. See <http://www.gnu.org/copyleft/>.

Each version of the License is given a distinguishing version number. If the Document specifies that a particular numbered version of this License "or any later version" applies to it, you have the option of following the terms and conditions either of that specified version or of any later version that has been published (not as a draft) by the Free Software Foundation. If

 <p>Universidad de Deusto</p> <p>Facultad de Ingeniería</p>	<p>A. GNU Free Documentation License</p>	<p>Proyecto Fin de Carrera</p>
--	--	------------------------------------

the Document does not specify a version number of this License, you may choose any version ever published (not as a draft) by the Free Software Foundation.

How to use this License for your documents

To use this License in a document you have written, include a copy of the License in the document and put the following copyright and license notices just after the title page:

Copyright (c) YEAR YOUR NAME.

Permission is granted to copy, distribute and/or modify this document under the terms of the GNU Free Documentation License, Version 1.2 or any later version published by the Free Software Foundation; with no Invariant Sections, no Front-Cover Texts, and no Back-Cover Texts. A copy of the license is included in the section entitled "GNU Free Documentation License".

If you have Invariant Sections, Front-Cover Texts and Back-Cover Texts, replace the "with...Texts." line with this:

with the Invariant Sections being LIST THEIR TITLES, with the Front-Cover Texts being LIST, and with the Back-Cover Texts being LIST.

If you have Invariant Sections without Cover Texts, or some other combination of the three, merge those two alternatives to suit the situation.

If your document contains nontrivial examples of program code, we recommend releasing these examples in parallel under your choice of free software license, such as the GNU General Public License, to permit their use in free software.