

# La Guía compacta de Org-mode

---

Versión 8.3beta (release 8.3beta-442-g2b98cd)

**Por Carsten Dominik.**

Traducido por David Arroyo Menéndez con contribuciones usuarios de emacs hispanohablantes. Diseño gráfico por Adolfo Antón Bravo.

---

Copyright © 2010–2014 Free Software Foundation

Permission is granted to copy, distribute and/or modify this document under the terms of the GNU Free Documentation License, Version 1.3 or any later version published by the Free Software Foundation; with no Invariant Sections, with the Front-Cover Texts being “Un Manual GNU,” and with the Back-Cover Texts as in (a) below. A copy of the license is included in the section entitled “GNU Free Documentation License” in the full Org manual, which is distributed together with the compact guide.

(a) El texto de contraportada de la FSF es: “Tu tienes la libertad para copiar y modificar este manual de GNU”.

# Resumen del Contenido

1	Introducción . . . . .	1
2	Estructura del documento . . . . .	3
3	Tablas . . . . .	8
4	Hiperenlaces . . . . .	10
5	Elementos TODO . . . . .	13
6	Etiquetas . . . . .	17
7	Propiedades . . . . .	20
8	Fechas y horas . . . . .	21
9	Capturar - Rellenar - Archivar . . . . .	25
10	Vistas de la Agenda . . . . .	28
11	Marcas para enriquecer la exportación . . . . .	34
12	Exportando . . . . .	38
13	Publicación . . . . .	41
14	Trabajando con código fuente . . . . .	42
15	Miscelánea . . . . .	45
A	GNU Free Documentation License . . . . .	47

# 1 Introducción

## 1.1 Prefacio

Org es un modo Emacs para guardar notas, mantener listas TODO, y hacer planificación de proyectos con un rápido y efectivo sistema de texto plano. Es también un sistema de publicación y autoría, que soporta trabajar con código fuente para programación literal e investigación reproducible.

*Este documento es un resumen derivado del [manual completo de Org-mode](#). Contiene todas las funcionalidades básicas y comandos, junto con importantes detalles de personalización. Se recomienda para principiantes que se asustan con manuales de 200 páginas solo por su tamaño.*

## 1.2 Instalación

**Importante:** Si se usa una versión de Org que es parte de una distribución de Emacs o un paquete de XEmacs, por favor sáltese esta sección y vaya directamente a [Sección 1.3 \[Activación\]](#), página 1.

Si se ha descargado Org de la web, como distribución de ficheros bien .zip, o bien .tar, o como un archivo Git, es preferible ejecutarlo directamente desde el directorio de la distribución. Se pueden añadir los subdirectorios lisp a la ruta que carga Emacs. Para hacer esto, se añade la siguiente línea al archivo .emacs:

```
(setq load-path (cons "~/path/to/orgdir/lisp" load-path))
(setq load-path (cons "~/path/to/orgdir/contrib/lisp" load-path))
```

Si ha estado usando git o un tar ball para obtener Org, debe ejecutar el siguiente comando para generar la información de autocarga. Comando:

```
make autoloads
```

## 1.3 Activación

Añada las siguientes líneas a su archivo .emacs. Las últimas cuatro líneas definen las claves *global* de algunos comandos —por favor, elija claves apropiadas por sí mismo.

```
;; Las siguiente líneas son siempre necesarias. Elige
;; tus propias claves
(global-set-key "\C-cl" 'org-store-link)
(global-set-key "\C-ca" 'org-agenda)
(global-set-key "\C-cc" 'org-capture)
(global-set-key "\C-cb" 'org-iswitchb)
```

Los ficheros con extensión ‘.org’ llevarán el modo Org automáticamente.

## 1.4 Realimentación

Si se encuentra problemas con Org, o si tiene preguntas, comentarios, u otras ideas acerca de él, por favor envíe un correo a la lista de Org [emacs-orgmode@gnu.org](mailto:emacs-orgmode@gnu.org). Para más información de cómo enviar informes de error, lea el manual principal.

## 2 Estructura del documento

Org está basado en el modo Outline y suministra comandos flexibles para editar la estructura de un documento.

### 2.1 Outlines

Org está implementado aprovechando el modo outline. Los ítems de contenido (outlines) permiten que el documento se organice en una estructura jerárquica, lo cual (al menos para mí) es la mejor forma para representar notas y pensamientos. Una visión preliminar de esta estructura se logra al encoger (ocultar) grandes partes del documento y mostrar sólo la estructura general del documento y las partes en las que se está trabajando. Org simplifica enormemente el uso de outlines para comprender la funcionalidad completa de mostrar/ocultar en un simple comando, `org-cycle`, el cual se asigna a la tecla TAB.

### 2.2 Cabeceras

Las cabeceras<sup>1</sup> definen la estructura del árbol de sangrado (ítems de contenido). Los títulos y subtítulos en Org comienzan con uno o más asteriscos, en el margen izquierdo<sup>2</sup>. Por ejemplo:

```
* Cabecera de nivel superior
** Segundo nivel
*** Tercer nivel
    cualquier texto
*** Tercer nivel
    más texto

* Otro título de nivel superior
```

Algunas personas encuentran que tantos asteriscos molestan y preferirían un ítem de contenido ‘outline’ que tenga un asterisco seguido de un espacio en blanco para comenzar cabeceras. Más adelante, en [Sección 15.2 \[Vista limpia\]](#), [página 45](#), se describe una configuración para realizar esto.

### 2.3 Visibilidad cíclica

Los outlines hacen posible ocultar partes del texto en el búffer. Org usa solo dos comandos, asignados a TAB y *S-TAB*, para cambiar la visibilidad en el búffer.

TAB            *Subárbol cíclico*: Rotación en el subárbol entre los estados

<sup>1</sup> Nota del Traductor. Headlines es traducido por cabecera y hace referencia a títulos y subtítulos

<sup>2</sup> Vea la variable `org-special-crtl-a/e` para configurar el comportamiento de *C-a* y *C-e* en los títulos.

```
,-> ENCOGIDO -> HIJO -> SUBÁRBOL --.
'-----'
```

Cuando se llama con el argumento prefijo (*C-u TAB*) o con la tecla de desplazamiento (*Shift*), se invoca el ciclo global.

*S-TAB* y *C-u TAB*

*Ciclo global*: Rotar el búffer entero entre los estados

```
,-> RESUMEN -> CONTENIDO -> MOSTRAR TODO --.
'-----'
```

*C-u C-u C-u TAB*

Mostrar todo, incluidos los calzoncillos

Cuando Emacs abre por primera vez un archivo Org, pone el estado global a RESUMEN (OVERVIEW) p.e. sólo las cabeceras de nivel superior son visibles. Esto puede configurarse a través de la variable `org-startup-folded`, o vía fichero añadiendo una palabra reservada `overview`, `content`, `showall`, como esta:

```
#+STARTUP: content
```

## 2.4 Movimiento

El siguiente comando salta a la siguiente cabecera en el búffer.

*C-c C-n*    Siguiete cabecera.

*C-c C-p*    Cabecera previa.

*C-c C-f*    Siguiete cabecera del mismo nivel.

*C-c C-b*    Cabecera previa del mismo nivel.

*C-c C-u*    Retroceder a la cabecera de nivel superior.

## 2.5 Edición de estructura

*M-RET*    Inserta una nueva cabecera al mismo nivel que la actual. Si el cursor está en un ítem de una lista plana, un nuevo ítem se crea (véase [Sección 2.7 \[Listas planas\]](#), página 5). Cuando este comando se usa en medio de una línea, la línea se parte y el resto de la línea será una nueva cabecera<sup>3</sup>.

*M-S-RET*    Inserta una nueva entrada TODO con el mismo nivel de la cabecera actual.

*TAB*    En una nueva entrada sin texto aún, *TAB* rotará cíclicamente a través de niveles similares.

---

<sup>3</sup> Si no desea que la línea se parta, personalice la variable `org-M-RET-may-split-line`.

*M-left/right*

Promociona/devalúa la cabecera actual en un nivel.

*M-S-left/right*

Promociona/devalúa el subárbol actual en un nivel.

*M-S-up/down*

Mueve el subárbol arriba/abajo (intercambia entre anterior/siguiente subárbol del mismo nivel).

*C-c C-w* Mueve la entrada o región en una localización diferente. Véase [Sección 9.2 \[Rellenar y copiar\]](#), página 26.

*C-x n s/w* Limitar la memoria intermedia al árbol actual / ocultándolo

Cuando se trata de una región activa (Transient Mark mode), promueve y devalúa las cabeceras de la región.

## 2.6 Árboles poco densos

Una característica importante de Org-mode es su posibilidad para construir *árboles poco densos* para la información seleccionada en un árbol de outlines, así se maneja el documento entero, pero la información seleccionada se hace visible con la estructura de cabeceras de encima<sup>4</sup>. Inténtelo ahora y verá inmediatamente como funciona.

Org-mode contiene varios comandos para crear árboles, todos estos comandos pueden ser accedidos a través del disparador:

*C-c /* Esto muestra una consola para introducir una clave especial para seleccionar un comando de creación de un árbol poco denso.

*C-c / r* Se refiere a que aparezca un patrón. La consola ofrece introducir una expresión regular y muestra un árbol poco denso con todas sus coincidencias. Cada coincidencia se ilumina; La iluminación desaparece presionando *C-c C-c*.

Otros comandos de árbol poco denso seleccionan las cabeceras basadas en la palabra reservada TODO, etiquetas o propiedades y serán discutidos posteriormente en este manual.

## 2.7 Listas planas

Sin una entrada en el árbol de outlines, las listas formateadas a mano pueden suministrar una estructura adicional. Ello también proporciona una forma de crear listas de cajas de chequeo (véase [Sección 5.6 \[Cajas de chequeo\]](#), página 16). Org soporta la edición de tales listas, y el conversor HTML (véase [Capítulo 12 \[Exportando\]](#), página 38) los analiza y formatea.

Org permite listas ordenadas, listas desordenadas y listas de descripción.

<sup>4</sup> Véase además las variables `org-show-hierarchy-above`, `org-show-following-heading`, `org-show-siblings` y `org-show-entry-below` para controlar los detalles de cómo el contexto se muestra en cada coincidencia.



- *Desordenada* los ítems de la lista comienzan con ‘-’, ‘+’ o ‘\*’ como marcas.
- *Ordenada* los ítems de la lista comienzan con ‘1.’ o ‘1’.
- *Descripción* la lista usa ‘ : : ’ para separar el *término* de la descripción.

Los ítems subsiguientes de la misma lista deben tener el mismo sangrado en la primera línea. Un ítem que termine antes de la siguiente línea será sangrado como bola/número, o no. Una lista termina cuando todos los ítems están cerrados, o antes de dos líneas en blanco. Un ejemplo:

```
** El Señor de los Anillos
  Mis escenas favoritas son (en este orden)
  1. El ataque de Rohirrim
  2. Combate de Eowyn con el rey
    + Esta es también mi escena favorita en el libro
    + Es realmente como Miranda Otto.
  Actores importantes es esta película:
  - Elijah Wood :: En el papel de Frodo
  - Sean Austin :: En el papel de Sam, amigo de Frodo.
```

Los siguientes comandos actúan en ítems cuando el cursor está en la primera línea de un ítem (la línea con la bola o el número).

**TAB** Los ítems pueden manejarse como cabeceras en diferentes niveles jerárquicos.

**M-RET** Inserta un nuevo ítem al nivel actual. Con el argumento de pre-fijo, se fuerza a una nueva cabecera (véase [Sección 2.5 \[Edición de estructura\]](#), página 4).

**M-S-RET** Inserta un nuevo ítem con una caja de chequeo (véase [Sección 5.6 \[Cajas de chequeo\]](#), página 16).

**M-S-up/down**

Mueve el ítem incluyendo los subítems arriba/abajo (intercambia con el ítem previo/siguiente del mismo sangrado. Si la lista es ordenada, volver a numerar es automático.

**M-left/M-right**

Decrementa/incrementa el sangrado del ítem, dejando los hijos sueltos.

**M-S-left/right**

Decrementa/incrementa el sangrado del ítem, incluyendo los subítems.

**C-c C-c**

Si es una caja de chequeo (véase [Sección 5.6 \[Cajas de chequeo\]](#), página 16) en una línea ítem, cambia el estado de la caja de chequeo. Además verifica las bolas y el sangrado consiste en la lista completa.

**C-c -**

Rota la lista entera a través de diferentes bolas de numeración/marcado (‘-’, ‘+’, ‘\*’, ‘1.’, ‘1’).

## 2.8 Notas al pie

Una Nota al pie se define como un párrafo que comienza con una nota al pie marcada entre corchetes en la columna 0, sin ningún sangrado. La referencia a la nota al pie es simplemente una marca hecha con corchetes, texto incluido. Por ejemplo:

```
La página web de Org[fn:1] ahora parece un poco mejor que cuando la
usaba.
```

```
...
```

```
[fn:1] El enlace es: http://orgmode.org
```

El siguiente comando maneja notas al pié:

**C-c C-x f** El comando de acción de Nota al pie. Cuando el cursor está en una referencia a una nota al pie, salta a su definición. Cuando está en la definición, salta a la (primera) referencia. En otro caso, crea una nueva nota al pie. Cuando este comando se llama con un argumento prefijo, aparecerá un menú adicional incluyendo opciones de volver a numerar.

**C-c C-c** Salta entre definición y referencia.

### Más lecturas

Capítulo 2 del manual de Org-mode

Tutorial de Sacha Chua

## 3 Tablas

Org viene con un rápido e intuitivo editor de tablas. Cálculos similares a los de una hoja de cálculo son soportados en conexión con el paquete Emacs `calc` (vea el manual de Emacs Calculator para más información sobre la calculadora de Emacs).

Org hace fácil formatear tablas en ASCII plano. Cualquier línea con ‘|’ como primer carácter no-espacio es considerado parte de una tabla. ‘|’ es además el separador de columnas. Una tabla puede parecer algo así:

```
| Nombre | Teléfono | Edad |
|-----+-----+-----|
| Pedro  | 12345678 | 17   |
| Ana    | 87654321 | 25   |
```

Una tabla se realinea automáticamente cada vez que se presiona `TAB` o `RET` o `C-c C-c` dentro de la tabla. `TAB` además mueve al siguiente campo (`RET` a la siguiente fila) y crea una nueva tabla de filas al final de la tabla o antes de las líneas horizontales. El sangrado de la tabla se pone en la primera línea. Cualquier línea comenzando con ‘|–’ se considera como un separador horizontal de línea y será expandido en la siguiente realineación para expandirse al ancho completo de la tabla. Así, para crear una tabla dentro, debe sólo introducirse:

```
|Nombre|Teléfono|Edad|
|–
```

y entonces presionar `TAB` para alinear la tabla y empieza rellenando campos. Incluso más rápido sería escribir `|Nombre|Teléfono|Edad` seguido de `C-c RET`.

Cuando se introduce texto en un campo, Org trata `DEL`, `Backspace` y todas las teclas de forma especial, para que la inserción y el borrado eviten desplazarse a otros campos. Por tanto, cuando se introduce *inmediatamente después de que el cursor se haya movido dentro de un nuevo campo* con `TAB`, `S-TAB` o `RET`, el campo es automáticamente puesto en blanco.

### Creación y conversión

`C-c |` Convierte la región activa en tabla. Si cada línea contiene al menos un carácter `TAB`, la función asume que la información está separada por tabuladores. Si cada línea contiene una coma, se asumen valores separados por coma (CSV). Si no, las líneas son partidas en campos por los espacios. Si no hay región activa, este comando crea una tabla Org vacía. Pero es fácil justo ahora comenzar a introducir algo como `|Nombre|Teléfono|Edad` `C-c RET`.

### Realineación y movimiento

`C-c C-c` Realinea la tabla sin mover el cursor.

`TAB` Realinea la tabla, se mueve al siguiente campo. Crea una nueva fila si es necesario.

- S-TAB* Realinea, moverse al campo anterior.
- RET* Realinea la tabla y se mueve abajo a la siguiente fila. Crea una nueva fila si es necesario.

### Edición de filas y columnas

*M-left*

*M-right* Mueve el cursor a la columna izquierda/derecha.

*M-S-left* Elimina la columna actual.

*M-S-right*

Inserta una nueva columna a la izquierda de la posición del cursor.

*M-up*

*M-down* Mueve la actual fila arriba/abajo.

*M-S-up* Elimina la fila o línea horizontal actual.

*M-S-down* Inserta una nueva fila sobre la fila actual. Con prefijo argumento, la línea se crea debajo de la actual.

*C-c -* Inserta una línea horizontal bajo la fila actual. Con prefijo argumento, la línea se crea sobre la línea actual.

*C-c RET* Inserta una línea horizontal bajo la fila actual, y mueve el cursor a la fila bajo la línea.

*C-c ^* Ordena las líneas de una tabla en una región. La posición del punto indica la columna usada para la ordenación, y el rango de líneas es el rango entre el separador de líneas más próximo, o la tabla completa.

### Más lecturas

[Capítulo 3 del manual de Org](#)

[Tutorial de tablas de Bastien](#)

[Tutorial de hojas de cálculo de Bastien](#)

[Tutorial de gráficos de Eric](#)

## 4 Hiperenlaces

Al igual que HTML, Org permite enlaces dentro de archivos, enlaces externos a otros archivos, artículos de Usenet, correos electrónicos y mucho más.

### 4.1 Formato de enlace

Org reconocerá enlaces de texto tipo URL y los activará como enlaces en los que se puede hacer click. El formato de enlace general, sin embargo, se ve de la siguiente manera:

```
[[enlace][descripcion]] o de manera alternativa [[enlace]]
```

Una vez que un enlace en el búffer está completo (con todos los corchetes presentes), Org cambiará la vista de tal manera que la ‘*descripción*’ se mostrará en vez de ‘[[enlace][descripción]]’ y ‘enlace’ será mostrado en vez de ‘[[enlace]]’. Para editar la parte invisible de ‘enlace’, use *C-c C-1* con el cursor en el enlace.

### 4.2 Enlaces internos

Si el enlace no parece una URL, puede ser debido a que es un enlace interno en el fichero actual. El caso más importante es un enlace como ‘[[#mi-id-personal]]’ que enlazará a la entrada con la propiedad CUSTOM\_ID como ‘[[#mi-id-personal]]’.

Enlaces tales como ‘[[Mi Objetivo]]’ o ‘[[Mi Objetivo] [Encuentra mi objetivo]]’ conduce a una búsqueda de texto en el fichero actual para el correspondiente objetivo se parezca a ‘<<Mi objetivo>>’

Los enlaces internos serán usados para referenciar su destino, a través de enlaces o números, cuando sea posible.

### 4.3 Enlaces externos

Org tiene soporte para enlaces a ficheros, sitios web, mensajes de correo electrónico y de News, entradas de bases de datos BBDB y enlaces a conversaciones de IRC y sus logs. Enlaces externos son identificadores tipo URL. Estos empiezan con una breve cadena de identificación seguida por dos puntos. Sin espacio después de los dos puntos. Aquí se presentan algunos ejemplos:

<code>http://www.astro.uva.nl/~dominik</code>	en la web
<code>file:/home/dominik/images/jupiter.jpg</code>	fichero, ruta absoluta
<code>/home/dominik/images/jupiter.jpg</code>	lo mismo que arriba
<code>file:papers/last.pdf</code>	fichero, ruta relativa
<code>file:projects.org</code>	otro fichero org
<code>docview:papers/last.pdf::NNN</code>	fichero abierto en modo doc-view en la página NNN
<code>id:B7423F4D-2E8A-471B-8810-C40F074717E9</code>	enlace a un identificador
<code>news:comp.emacs</code>	enlace a un grupo de news
<code>mailto:adent@galaxy.net</code>	enlace de Correo
<code>vm:folder</code>	enlace a una carpeta de VM

<code>vm:folder#id</code>	enlace a un mensaje de VM
<code>wl:folder#id</code>	enlace a un mensaje de WANDERLUST
<code>mhe:folder#id</code>	enlace a un mensaje de MH-E
<code>rmail:folder#id</code>	enlace a un mensaje de RMAIL
<code>gnus:group#id</code>	enlace a un mensaje de GNUS
<code>bbdb:R.*Stallman</code> (presión regular)	enlace a una entrada de BBDB (con ex-
<code>irc:/irc.com/#emacs/bob</code>	enlace de IRC
<code>info:org:External%20links</code> (código en blanco)	enlace a un nodo Info (con el espa-

Un enlace debe ser encerrado entre corchetes y puede tener un texto descriptivo que será mostrado en vez de la URL (véase [Sección 4.1 \[Formato de enlace\]](#), página 10), por ejemplo:

```
[[http://www.gnu.org/software/emacs/] [GNU Emacs]]
```

Si la descripción es un nombre de fichero o URL que apunta a una imagen, la exportación HTML (véase [Sección 12.4 \[HTML export\]](#), página 39) introducirá la imagen como un botón al que se puede hacer click. Si no hay descripción y el enlace apunta a una imagen, esta imagen se incrustará dentro del fichero HTML exportado.

## 4.4 Manejando enlaces

Org provee métodos para crear un enlace con la sintaxis correcta, para insertarlo en un fichero Org, y poder seguir el enlace.

**C-c l** Almacena un enlace desde la posición actual. Éste es un comando *global* (se debe crear el atajo de teclado por uno mismo), el cual puede ser usado en cualquier búffer para crear un enlace. El enlace será almacenado para posteriores inserciones dentro de un búffer Org (ver más abajo).

**C-c C-l** Inserta un enlace. Esto sugiere un enlace que será insertado dentro del búffer. Se puede escribir un enlace, o usar la teclas del historial *arriba* y *abajo* para acceder a los enlaces almacenados. También se consultará por la parte de descripción del enlace. Cuando se llama al prefijo *C-u*, se usa el autocompletado del nombre del fichero para enlazar a un fichero.

**C-c C-l** (con el cursor en un enlace existente)

Cuando el cursor está en enlace existente, *C-c C-l* permite editar el enlace y las partes de descripción del enlace.

**C-c C-o** o *mouse-1* o *mouse-2*

Abre el enlace en el que está el cursor.

**C-c &** Salta a una posición guardada. Una posición se guarda con los siguientes comandos de enlaces internos, y por *C-c %*. Usando este comando varias veces se mueve a través de un anillo de posiciones previamente grabadas en una sucesión directa.

## 4.5 Destinos enlazados

Los enlaces de ficheros pueden contener información adicional para hacer que Emacs salte a una posición particular en el fichero cuando se sigue un enlace. Esto puede ser un número de línea o una opción de búsqueda después de los dos puntos dobles.

Aquí está la sintaxis de los diferentes caminos para adjuntar una búsqueda a un enlace de fichero, junto con una explicación:

<code>[[file:~/code/main.c::255]]</code>	Encontrar línea 255
<code>[[file:~/xx.org::Mi Objetivo]]</code>	Encontrar '<<Mi Objetivo>>'
<code>[[file:~/xx.org::#mi-id-personal]]</code>	Encontrar entrada con id personal

### Más lecturas

[Capítulo 4 del manual](#)

## 5 Elementos TODO

Org mode no requiere listas TODO que estén en documentos separados. En vez de eso, los ítems TODO son parte de ficheros de notas, porque los ítems TODO normalmente llegan mientras se toman notas. Con Org mode, simplemente marca cualquier entrada en un árbol siendo un ítem TODO. De este modo, la información no está duplicada, y los ítems TODO permanecen en el contexto emergente.

Org mode provee métodos para dar una visión de todas las cosas que se tienen que hacer, a veces llegan desde muchos ficheros.

### 5.1 Usando los estados TODO

Cualquier título o subtítulo puede llegar a ser un ítem TODO cuando empieza con la palabra ‘TODO’, por ejemplo:

```
*** TODO Escribir carta a Santa Fortuna
```

Los comandos más importantes para trabajar con entradas TODO son:

**C-c C-t** Rotar el estado TODO del ítem actual

```
(no marcado) -> TODO -> DONE -> (no marcado)\n
```

La misma rotación puede también ser hecha “de manera remota” desde los búffers de agenda y línea de tiempo con la tecla *t* (véase [Sección 10.4 \[Comandos de la agenda\], página 31](#)).

**S-derecha/izquierda**

Seleccionar el siguiente/precedente estado TODO, similar a rotar.

**C-c / t** Ver ítems TODO en un *árbol poco denso* (véase [Sección 2.6 \[Árboles poco densos\], página 5](#)). Encoge el búffer, pero muestra todos los ítems y la jerarquía de títulos y subtítulos por encima de ellos.

**C-c a t** Muestra la lista global TODO. Esta colección de ítems TODO de todos los archivos de la agenda (véase [Capítulo 10 \[Vistas de la Agenda\], página 28](#)) en un único búffer.

**S-M-RET** Inserta una nueva entrada TODO debajo del ítem actual.

Al cambiar el estado de un ítem TODO se puede también activar cambios de etiqueta. Véase la documentación de la opción `org-todo-state-tags-triggers` para más detalles.

### 5.2 Flujos de trabajo multi-estado

Se pueden usar palabras reservadas TODO para indicar diferentes estados *secuenciales* en el proceso de trabajo:



```
(setq org-todo-keywords
  '((sequence "TODO" "FEEDBACK" "VERIFY" "|" "DONE" "DELEGATED")))
```

La barra vertical separa las palabras reservadas TODO (estados que *necesitan acción*) de los estados DONE (realizados, que *no necesitan más acción*). Si no se proporciona la barra separadora, el último estado es usado como estado DONE. Con esta configuración, el comando `C-c C-t` rotará una entrada desde TODO a FEEDBACK, después a VERIFY y finalmente a DONE y DELEGATED.

```
(setq org-todo-keywords
  '((sequence "TODO(t)" "|" "DONE(d)")
    (sequence "REPORT(r)" "BUG(b)" "KNOWNCAUSE(k)" "|" "FIXED(f)")))
```

Las palabras reservadas son todas diferentes, esto ayuda a Org mode a guardar la traza de la subsecuencia que debería ser usada para una entrada dada. El ejemplo también muestra cómo definir teclas para un rápido acceso a un estado particular, añadiendo una letra entre paréntesis, así al pulsar `C-c C-t` se pregunta por la letra para cada palabra reservada que se quiere escoger.

Para definir palabras reservadas TODO que son válidas únicamente en un solo fichero, use el siguiente texto en cualquier lugar del fichero.

```
#+TODO: TODO(t) | DONE(d)
#+TODO: REPORT(r) BUG(b) KNOWNCAUSE(k) | FIXED(f)
#+TODO: | CANCELED(c)
```

Después de cambiar una de estas líneas, use `C-c C-c` con el cursor todavía en la línea para que Org mode reconozca los cambios.

### 5.3 Proceso de acceso

Org mode puede grabar automáticamente una marca de tiempo *timestamp* y posiblemente una nota cuando se marca un ítem TODO como DONE, ó incluso cada vez que se cambia el estado de un ítem TODO. Este sistema es altamente configurable; las configuraciones pueden seguir una lógica por tecla y pueden afectar a un fichero o incluso a un subárbol. Para más información de cómo fijar fecha y hora de una tarea, lea la [Sección 8.4 \[Estableciendo tiempo de trabajo\]](#), página 23.

### Cerrando ítems

El registro más básico es guardar *cundo* un ítem TODO se finaliza. Esto se logra con<sup>1</sup>.

```
(setq org-log-done 'time)
```

Así, cada vez que se cambia una entrada desde un estado TODO (no DONE) a cualquiera de los estados DONE, una línea `'CLOSED: [timestamp]'` será insertada justo después de la cabecera. Si se quiere grabar una nota con una marca de tiempo *timestamp* use lo siguiente:<sup>2</sup>

<sup>1</sup> La correspondiente configuración en el búffer es `#+STARTUP: logdone`

<sup>2</sup> La correspondiente configuración en el búffer es: `#+STARTUP: lognotedone`

```
(setq org-log-done 'note)
```

Entonces, se preguntará por la nota, y la nota será almacenada debajo de la entrada con una cabecera Cerrar nota *Closing Note*.

## Trazando los estados TODO

Se podría querer guardar la traza de cambios de estado TODO. Para un cambio se puede registrar solo la marca de tiempo (timestamp), o además una nota con una marca de tiempo para un cambio. Estos registros serán insertados después de la cabecera como una lista de ítems. Cuando se toman un montón de notas, se podría querer tener las notas fuera de la vista dentro de un "*cajón*" (drawer)

Para el registro de estados, Org mode confía en una configuración basada en palabras reservadas. Esto se logra añadiendo marcas especiales ‘!’ (para una marca de tiempo) y ‘@’ (para una nota) entre paréntesis después de cada palabra reservada. Por ejemplo:

```
#+TODO: TODO(t) WAIT(w@/!) | DONE(d!) CANCELED(c@)
```

definirá las palabras reservadas TODO y el acceso a los atajos de teclado, y también solicita que se grabe el momento en el que la entrada se estableció como DONE, y que una nota se grabe al cambiar a WAIT o CANCELED. La misma sintaxis funciona también cuando se define `org-todo-keywords`.

## 5.4 Prioridades

Si se usa Org mode intensamente, se puede acabar con suficientes ítems TODO de manera que empiece a tener sentido priorizarlos. Priorizar puede ser hecho poniendo una *marca de prioridad* en la cabecera de un ítem TODO, como esta:

```
*** TODO [#A] Escribir carta a Santa Fortuna
```

Org mode soporta tres prioridades: ‘A’, ‘B’, ‘C’. ‘A’ es la más alta, por defecto será ‘B’ si no se da ninguna. Las prioridades marcan diferencia sólo en la agenda.

**C-c** ,      Pone una prioridad a la actual cabecera. Presionando ‘A’, ‘B’ o ‘C’ se selecciona la prioridad, o SPC para eliminar la marca.

**S-up/down**    Incremente/decrementa la prioridad de la cabecera actual

## 5.5 Partiendo tareas en subtareas

Es a menudo ventajoso dividir grandes tareas en pequeñas y manejables subtareas. Puede realizar esto creando un árbol de ideas (outlines) bajo un ítem TODO, el cual detalla las subtareas en el árbol. Para poner la vista sobre la fracción de subtareas que están ya completadas, inserte un ‘[/]’ o ‘[%]’ en cualquier lugar de la cabecera. Estas marcas serán actualizadas cada vez que el estado TODO de algún hijo cambie, o se presione **C-c C-c** en la marca. Por ejemplo:

```
* Organizar Fiesta [33%]
** TODO Llamar a la gente [1/2]
*** TODO Pedro
*** DONE Sara
** TODO Comprar comida
** DONE Hablar con el vecino
```

## 5.6 Cajas de chequeo

Cada ítem en una lista plana (véase [Sección 2.7 \[Listas planas\]](#), página 5) puede ser un cuadro de chequeo comenzando con la cadena ‘[ ]’. Los cuadros de chequeo no están incluidos en la lista TODO global, así a menudo es preferible dividir la tarea en un número reducido de pasos. Véase el siguiente ejemplo de una lista de chequeo:

```
* TODO Organizar Fiesta [1/3]
- [-] Llamar a la gente [1/2]
  - [ ] Pedro
  - [X] Sara
- [X] Comprar comida
- [ ] pensar qué música escuchar
```

Las cajas de chequeo funcionan jerárquicamente, si un ítem es caja de chequeo y tiene hijos que son cajas de chequeo, marcando las cajas de chequeo hijos se marcará la caja de chequeo del padre para reflejar si ninguno, alguno o todos los hijos están marcados.

El siguiente comando funciona con cajas de chequeo ‘checkboxes’:

*C-c C-c* Cambia el estado de la caja de chequeo o (con prefijo) añade una caja de chequeo.

*M-S-RET* Inserta un nuevo ítem con un cuadro de chequeo. Esto funciona solo si el cursor está en un ítem de la lista plana (véase [Sección 2.7 \[Listas planas\]](#), página 5).

### Más lecturas

Capítulo 5 del manual

Tutorial de introducción de David O’Toole

Configuración de GTD de Charles Cave

## 6 Etiquetas

Una excelente forma de nombrar y contextualizar información interrelacionada es asignar *etiquetas* a las cabeceras. Org mode tiene un amplio soporte para etiquetas.<sup>1</sup>

Cada cabecera puede contener una lista de etiquetas; dicha lista se introduce al final de la cabecera correspondiente. Las etiquetas son palabras normales conteniendo letras, números, ‘\_’ y ‘@’. Las etiquetas deben estar precedidas y seguidas por dos puntos, por ejemplo, ‘:trabajo:’. Es posible asignar varias etiquetas, como en ‘:trabajo:urgente:’. Las etiquetas por defecto estarán en negrilla con el mismo color que la cabecera.

### 6.1 Marca de herencia

Las *etiquetas* hacen uso de la estructura jerárquica de los árboles de org-mode. Si una cabecera tiene una cierta etiqueta, todas las subcabeceras heredarán la etiqueta también. Por ejemplo, en la lista

```
* Encuentro con franceses           :trabajo:
** Resumen para Carolina           :jefa:notas:
*** TODO Preparar presentaciones para ella :actividad:
```

La cabecera final tendrá las etiquetas ‘:trabajo:’, ‘:jefa:’, ‘:notas:’ y ‘:actividad:’ incluso aunque la cabecera final no esté explícitamente marcada con estas etiquetas. Se pueden también asignar etiquetas para que todas las entradas en un fichero hereden si estas etiquetas fueron definidas en un hipotético nivel cero alrededor del fichero entero. Usa una línea como esta<sup>2</sup>:

```
#+FILETAGS: :Pedro:Jefe:Secreto:
```

### 6.2 Poniendo marcas

Las etiquetas pueden simplemente ser escritas en un búffer al final de una cabecera. Después de los dos puntos, *M-TAB* autocompleta etiquetas. Hay también un comando para insertar etiquetas:

**C-c C-q** Introduce nuevas etiquetas para la cabecera actual. Org mode ofrecerá autocompletado o una interfaz especial de una sola tecla para asignar etiquetas, ver más abajo. Después de presionar **RET**, las etiquetas serán insertadas y alineadas para **org-tags-column**. Cuando se llama con el prefijo **C-u**, todas las etiquetas en el búffer actual serán alineadas a esta columna, solo para que las cosas se vean bien.

<sup>1</sup> Nota del Traductor: Tanto label como tag se traduce como etiqueta. En general, label se utiliza más como nombre que se le asigna a algo y tag como palabra clave, si bien ambas suelen traducirse como etiqueta. Así, he decidido traducir *implement label* como nombrar y *tag* como etiqueta.

<sup>2</sup> Como siempre estas configuraciones de búffer se activan presionando **C-c C-c**.

*C-c C-c* Cuando el cursor está en una cabecera, hace lo mismo que *C-c C-q*.

Org soporta inserción de etiquetas basado en una *lista de etiquetas*. Por defecto esta lista es construida dinámicamente, conteniendo todas las etiquetas actualmente usadas en el búffer. También se puede especificar globalmente una lista dura de etiquetas con la variable `org-tag-alist`. Finalmente se puede asignar las etiquetas por defecto para un fichero dado con líneas como

```
#+TAGS: @trabajo(t) @casa(c) @futbol(f) servidor(s) pc(p)
```

Por defecto, Org mode usa las facilidades de completación del minibúffer para introducir etiquetas. Sin embargo, también implementa otro rápido método de selección de etiquetas llamado *fast tag selection*. Esto permite que se seleccionen y se dejen de seleccionar etiquetas con solo presionar una sola tecla. Para que esto funcione bien se debe asignar letras únicas para las etiquetas más usadas. Se puede hacer esto de manera global configurando la variable `org-tag-alist` en el fichero `.emacs`. Por ejemplo, se puede encontrar la necesidad de etiquetar muchos ítems en diferentes ficheros con `:@casa:`. En este caso se puede hacer algo como:

```
(setq org-tag-alist '(("@trabajo" . ?t) ("@casa" . ?c) ("portatil" . ?p)))
```

Si la etiqueta `'tag'` es solo relevante para el fichero en el que se está trabajando, entonces se puede asignar la línea de opción TAG de la siguiente manera:

```
#+TAGS: @trabajo(t) @casa(c) @futbol(f) servidor(s) pc(p)
```

## 6.3 Etiquetar grupos

En un conjunto de etiquetas mutuamente exclusivas, la primera etiqueta puede ser definida como una *etiqueta de grupo*. Cuando se busca una etiqueta de grupo, devolverá las coincidencias para todos los miembros en el grupo. En una vista de agenda, filtrar por una etiqueta de grupo mostrará cabeceras etiquetadas con al menos uno de los miembros del grupo. Esto crea búsquedas y filtros incluso más flexibles.

Se pueden establecer grupos de etiquetas insertando un punto y coma entre la etiqueta grupo y otras etiquetas, como esta:

```
#+TAGS: { @read : @read_book @read_ebook }
```

En este ejemplo, `@read` es una *etiqueta grupo* para un conjunto de tres etiquetas: `@read`, `@read_book` y `@read_ebook`.

Puede también usar las palabras clave `:grouptags` directamente cuando se configure `org-tag-alist`, ver la documentación de esta variable.

Si se quieren ignorar las etiquetas grupo temporalmente, cambia el soporte de etiquetas grupo con `org-toggle-tags-groups`, asociado a *C-c C-x q*. Si se quieren deshabilitar etiquetas grupos completamente, configura `org-group-tags` a nil.

## 6.4 Buscando marcas

Una vez que un sistema de etiquetas ha sido configurado, puede ser usado para recoger información dentro de listas especiales.

- `C-c \`  
`C-c / m` Crear un árbol expandido con todas las cabeceras coincidentes con la etiqueta (tag) buscada. Con el prefijo `C-u`, ignora las cabeceras que no son `TODO`.
- `C-c a m` Crea una lista global de marcas coincidentes de todos los archivos de la agenda. Véase [Sección 10.3.3 \[Emparejar marcas y propiedades\]](#), página 30.
- `C-c a M` Crea una lista global de marcas coincidentes de todos los archivos de la agenda, pero chequea solo ítems `TODO` y fuerza el chequeo de subítems (ver la variable `org-tags-match-list-sublevels`).

Estos comandos buscan coincidencias de cadenas que permiten una lógica básica como `+jefe+urgente-proyecto1`, para encontrar entradas con marcas `'jefe'` y `'urgente'`, pero sin `'proyecto1'`, o `José|Juan` para encontrar ambas entradas, tanto `'José'` como `'Juan'`. La sintáxis completa de las cadenas de búsqueda es rica y permite además coincidencias con todas las palabras clave `TODO`, entradas de nivel y propiedades. Para una completa descripción con muchos ejemplos, vea [Sección 10.3.3 \[Emparejar marcas y propiedades\]](#), página 30.

### Más lecturas

Capítulo 6 del manual

artículo de Sacha Chua acerca de etiquetado en Org-mode

## 7 Propiedades

Las propiedades son pares clave valor asociados con una entrada. Estos se encuentran en un lugar especial con el nombre `PROPERTIES`. Cada propiedad se especifica en una línea simple, con la clave (rodeada por dos puntos) primero, y el valor después de ésta.

```
* Colección de CDs
** Clásica
*** Variaciones de Goldberg
:PROPERTIES:
:Título:      Goldberg Variations
:Compositor:  J.S. Bach
:Discográfica: Deutsche Grammophon
:Discos:      1
:END:
```

Se pueden definir los valores permitidos para una propiedad particular ‘:Xyz:’ asignando un propiedad ‘:Xyz\_ALL:’. Esta propiedad especial es *heredada*, así si se asigna en una entrada de nivel 1, entonces se aplicará al árbol entero. Cuando los valores permitidos están definidos, asignar la propiedad correspondiente llega a ser fácil y es menos propensa a errores. Para el ejemplo de la colección de CDs, se pueden predefinir las discográficas y el número de discos en una caja como esta:

```
* Colección de CDs
:PROPERTIES:
:Discos_ALL: 1 2 3 4
:Discográfica_ALL: "Deutsche Grammophon" Philips EMI
:END:
```

o globalmente usando `org-global-properties`, o un fichero amplio como este:

```
#+PROPERTY: Discos_ALL 1 2 3 4
```

**C-c C-x p** Asigna una propiedad. Se solicitará un nombre y un valor para la propiedad.

**C-c C-c d** Elimina una propiedad de la entrada actual.

Para crear árboles expandidos y listas especiales con selección basada en propiedades, los mismos comandos son usados para buscar marcas (véase [Sección 6.4 \[Buscando marcas\]](#), página 19). La sintaxis completa para la búsqueda de cadenas es descrita en la [Sección 10.3.3 \[Emparejar marcas y propiedades\]](#), página 30.

### Más lecturas

Capítulo 7 del manual

Bastien Guerry’s column view tutorial

## 8 Fechas y horas

Para tener una planificación de un proyecto, los ítems TODO pueden ser etiquetados con una fecha y/o con una hora. La cadena con este formato especial que trae la información de fecha y hora es llamado *timestamp* en Org mode.

### 8.1 Instante en el tiempo

Una marca de tiempo (*timestamp*) es una especificación de una fecha (posiblemente con un tiempo o un rango de tiempos) en un formato especial. Como ‘<2003-06-16 Jue>’ o ‘<2003-06-16 Jue 09:36>’ o ‘<2003-06-16 Jue 12:00-12:30>’. Una marca de tiempo puede aparecer en cualquier lugar, titular o cuerpo de una entrada del árbol Org. Su presencia causa que la entrada sea mostrada en una fecha específica de la agenda la (véase [Sección 10.3.1 \[Agenda semanal/diaria\], página 29](#)). Se distingue:

#### Una marca de tiempo; Evento; Cita

Una marca de tiempo simple asigna una fecha/hora a un solo ítem. Esto es solo como escribir una cita o evento en una agenda en papel.

- \* Ver pelis con Pedro  
<2006-11-01 Wed 19:15>
- \* Discusión acerca del cambio climático  
<2006-11-02 Thu 20:00-22:00>

#### Marca de tiempo con intervalo de repetición

Una marca de tiempo puede contener un *intervalo de repetición*, indicando que se aplica no solo en la fecha dada, sino que aparece una y otra vez después de un cierto intervalo de N días (d), semanas *weeks* (w), meses (m), o años *years* (y). Lo siguiente muestra en la agenda cada Miércoles:

- \* Recoger a Samuel en el colegio  
<2007-05-16 Wed 12:30 +1w>

#### Entradas de estilo diario sexp

Para especificaciones de fecha más complejas, Org mode da soporte usando las entradas diarias sexp implementadas en el paquete Emacs calendar/diary. Por ejemplo:

- \* El encuentro nerd los segundos jueves del mes  
<%(diary-float t 4 2)>

#### Rango fecha/hora

Dos marcas de tiempo conectadas por ‘--’ denotan un rango.

- \*\* Encuentro en Amsterdam  
<2004-08-23 Mon>--<2004-08-26 Thu>

#### Marcas de tiempo inactivas

Como una marca de tiempo *timestamp*, pero con paréntesis cuadrados en vez de angulares. Estas marcas de tiempo son inactivas en el sentido de que *no* aparecen como entrada en la agenda.

- \* Julia llegó tarde por quinta vez  
[2006-11-01 Wed]



## 8.2 Creando instantes de tiempo

Para que Org mode reconozca marcas de tiempo, necesitan estar en el formato específico. Todos los comandos que se listan debajo producen marcas de tiempo en el formato correcto.

**C-c .** Crea un diálogo para una fecha e inserta la correspondiente marca de tiempo. Cuando el cursor está en una marca de tiempo existente en el búffer, el comando se usa para modificar esta marca de tiempo en vez de insertar una nueva. Cuando este comando se usa dos veces seguidas, un rango de tiempo se inserta. Hay un prefijo para añadir también la hora actual.

**C-c !** Como **C-c .**, pero inserta una marca de tiempo inactiva que no causará una entrada en la agenda.

**S-left/right**

Cambia la fecha en el cursor por un día.

**S-up/down**

Cambia el ítem bajo el cursor en una marca de tiempo. El cursor puede ser en un año, mes, día, hora, o minuto. Cuando la marca de tiempo contiene un rango de tiempo como '15:30-16:30', modificando la primera vez dividirá los segundos, dividiendo el bloque del momento con el tamaño constante. Para cambiar el tamaño, modifica una segunda vez.

Cuando Org mode pregunta por una fecha u hora, acepta cualquier cadena conteniendo alguna fecha y/o información, e inteligentemente interpretará la cadena, sacando alguna por defecto, para la información no especificada desde la actual fecha y hora. También se puede seleccionar una fecha en un calendario pop-up. Ver el manual para más información de cómo funciona la fecha/hora.

## 8.3 Fecha límite y planificación

Una marca de tiempo puede ser precedida por palabras clave especiales para facilitar la planificación.

### **DEADLINE**

Significa: la tarea (normalmente un ítem TODO, aunque no necesariamente) se supone que finalizará en tal fecha.

**C-c C-d** Inserta la palabra clave 'DEADLINE' con una marca, en la línea siguiente a la cabecera o titular *headline*.

En la fecha de finalización *deadline*, la tarea será listada en la agenda. Además la agenda para *hoy* traerá un aviso acerca de la fecha de finalización pasada, empezando **org-deadline-warning-days** antes de la fecha y continuando hasta que la entrada es marcada como realizada *DONE*. Un ejemplo:

```
*** TODO escribir artículo acerca de la Tierra para la Guía.
    El editor encargado es [[bbdb:Ford Prefect]]
    DEADLINE: <2004-02-29 Sun>
```

### SCHEDULED

significa: se *planifica empezar trabajando* en esta tarea en la fecha dada<sup>1</sup>

*C-c C-s* Insertar la palabra clave ‘SCHEDULED’ con una marca, en la línea después de la cabecera.

La cabecera será listada bajo la fecha dada<sup>2</sup>. Además, un recordatorio de que la fecha introducida en la agenda que ha pasado estará presente en la compilación para *hoy*, hasta que la entrada es marcada como DONE. Por ejemplo, la tarea automáticamente será reenviada hasta ser completada.

```
*** TODO Llamar a Trillian para una cita en Año Nuevo.
    SCHEDULED: <2004-12-25 Sat>
```

Algunas tareas necesitan ser repetidas de nuevo y de nuevo. Org mode ayuda a organizar tales tareas usando un repetidor en una fecha límite *DEADLINE*, *SCHEDULED*, o una marca de tiempo en texto plano. En el siguiente ejemplo

```
** TODO Pagar el alquiler
    DEADLINE: <2005-10-01 Sat +1m>
```

el *+1m* es un repetidor; la interpretación que se pretende es que la tarea tiene una fecha límite en *<2005-10-01>* y repite por sí mismo cada mes empezando desde este momento.

## 8.4 Estableciendo tiempo de trabajo

Org mode permite contabilizar el tiempo que se gasta en tareas específicas en un proyecto.

*C-c C-x C-i*

Comenzar a medir el tiempo en el ítem actual (*clock-in*). Esto inserta la palabra clave CLOCK junto con una marca de tiempo. Cuando se llama con un argumento prefijo *C-u*, selecciona la tarea desde una lista de las tareas que recientemente se midió el tiempo.

*C-c C-x C-o*

Parar el reloj, finalizar la medición de tiempo (*clock-out*). Esto inserta otra marca de tiempo en la misma localización donde el reloj se inició por última vez. También se computa directamente el tiempo resultante en insertarlo después del rango de tiempo ‘=> HH:MM’.

<sup>1</sup> Esto es bastante diferente, ya que normalmente se comprende la *planificación de un encuentro*, que se hace en Org-mode insertando una marca de tiempo sin palabra clave.

<sup>2</sup> Todavía será listada en esta fecha después de que se marque como DONE. Si no gusta esto, se debe establecer la variable `org-agenda-skip-scheduled-if-done`.

*C-c C-x C-e*

Actualizar el esfuerzo de estimar la tarea para la actual medición de reloj.

*C-c C-x C-q*

Cancela la medición de reloj actual. Esto es útil si el reloj fue iniciado por error, o si se ya se finalizó el trabajo.

*C-c C-x C-j*

Salta a la entrada que contiene el reloj actualmente en ejecución. Con un prefijo *C-u*, selecciona la tarea objetivo desde una lista de tareas recientemente cronometradas.

*C-c C-x C-r*

Inserta un bloque dinámico conteniendo un informe de tiempo de reloj como una tabla de Org-mode dentro del fichero actual. Cuando el cursor es una tabla de reloj existente, lo actualiza.

```
#+BEGIN: clocktable :maxlevel 2 :emphasize nil :scope file
#+END: clocktable
```

Para detalles acerca de cómo personalizar esta vista, ver [el manual](#).

*C-c C-c*

Actualiza el bloque dinámico a punto. El cursor necesita estar en la línea `#+BEGIN` del bloque dinámico.

La tecla `l` puede ser usada en la línea de tiempo (véase [Sección 10.3.4 \[Línea de tiempo\]](#), página 30) y en la agenda (véase [Sección 10.3.1 \[Agenda semanal/diaria\]](#), página 29) para mostrar las tareas en las que se trabajará o cerrarán durante el día.

### Más lecturas

Capítulo 8 del manual

Tutorial de Fechas y Horas de Charles Cave

Flujo de trabajo de cronometrar de Bernt Hansen

## 9 Capturar - Rellenar - Archivar

Una parte importante de cualquier organización de sistemas es la habilidad para capturar nuevas ideas y tareas rápido, y asociar material de referencia con ellas. Org define un proceso de captura para crear tareas. En el sistema, puede surgir la necesidad de mover tareas y proyectos. Mover árboles de proyectos completos a un archivo deja el sistema compacto y rápido.

### 9.1 Capturar

Org permite almacenar notas rápidas con pequeñas interrupciones al quehacer cotidiano. Se pueden definir plantillas para nuevas entradas y asociarlos a diferentes objetivos para almacenar notas.

#### Poniendo una posición de captura

La siguiente personalización configura un fichero objetivo por defecto<sup>1</sup> para notas y define un atajo global para capturar nuevas cosas.

```
(setq org-default-notes-file (concat org-directory "/notes.org"))
(define-key global-map "\C-cc" 'org-capture)
```

#### Usando capturas

- `C-c c` Comienza un proceso de captura, que deja dentro un estrecho búffer indirecto para editar.
- `C-c C-c` Una vez que se ha introducido información dentro del búffer de captura, `C-c C-c` devolverá a la ventana de configuración antes del proceso de captura, así se puede resumir el trabajo sin distracción.
- `C-c C-w` Finaliza moviendo la entrada a una nueva localización (ver sección 9.2).
- `C-c C-k` Abortar el proceso de captura y volver al estado previo.

#### Plantillas para capturar notas

Se pueden usar plantillas para generar diferentes formas de capturar notas y almacenarlas en diferentes lugares. Por ejemplo, si gusta almacenar nuevas tareas bajo una cabecera ‘Tareas’ en el fichero `TODO.org`, y artículos de revistas en un árbol de fechas en `journal.org` se podría usar:

```
(setq org-capture-templates
  '(("t" "Todo" entry (file+headline "~/org/gtd.org" "Tareas")
    "* TODO %?\n %i\n %a")
    ("j" "Revistas" entry (file+datetree "~/org/journal.org")
    "* %?\nIntroducidas en %U\n %i\n %a")))
```

<sup>1</sup> Usando plantillas de captura, se pueden definir localizaciones de grano más fino, ver [Plantillas para capturar notas], página 25.

En estas entradas, la primera cadena es la clave para lograr la plantilla, la segunda es una descripción corta. Entonces sigue el tipo de la entrada y una definición de la localización del objetivo para almacenar la nota. Finalmente, la plantilla en sí misma, es una cadena con %-escapes para rellenar la información basada en tiempo y contexto.

Cuando se llama a *M-x org-capture*, Org pregunta por una clave para seleccionar la plantilla (si se tiene más de una plantilla) y entonces se prepara el búffer así:

```
* TODO
[[file:enlace a dónde se estaba inicializando la captura]]
```

Durante la expansión de la plantilla, %-escapes<sup>2</sup> especiales permiten la inserción dinámica de contenido. Aquí una pequeña selección de las posibilidades, consulte el manual para más.

<code>%a</code>	anotación, normalmente el enlace creado con <code>org-store-link</code>
<code>%i</code>	contenido inicial, la región cuando se captura se llama con C-u.
<code>%t</code> , <code>%T</code>	marca de tiempo, solo fecha, o fecha y tiempo
<code>%u</code> , <code>%U</code>	como el de debajo, pero con marcas de tiempo inactivas

## 9.2 Rellenar y copiar

Cuando se revisa el dato capturado, se puede querer rellenar o copiar algunas de las entradas dentro de una lista diferente, por ejemplo, dentro de un proyecto. Cortando, encuentra la localización correcta y entonces pegar la nota es complejo. Para simplificar este proceso, se pueden usar los siguientes comandos:

*C-c M-x* Copiar la entrada o región. Este comando se comporta como `org-refile`, excepto que el original no será borrado.

*C-c C-w* Reajusta la entrada o región al punto. Este comando ofrece posibles localizaciones para reajustar la entrada y permite seleccionar una con autocompletado. El ítem (o todos los ítems en la región) se rellenan bajo el objetivo estructurado como subítem. Por defecto, todas las cabeceras de nivel 1 en el actual búffer son consideradas para ser objetivos, pero se pueden tener definiciones más complejas a través de un número de ficheros. Ver la variable `org-refile-targets` para detalles.

*C-u C-c C-w* Usa el interfaz de reajuste para saltar a un encabezado.

*C-u C-u C-c C-w* Salta a la localización dónde el último `org-refile` se movió a un árbol.

---

<sup>2</sup> Si se necesita una de estas secuencias literalmente, escapa el % con una barra invertida.

## 9.3 Archivando

Cuando un proyecto se representa por un (sub)árbol y este se finaliza, se puede querer mover el árbol fuera del camino y hacer que no se contribuya a la agenda. Archivar es importante para guardar los ficheros de trabajo de búsquedas globales como la construcción de vistas de agenda rápidas. La acción más común de archivar es mover un árbol de proyecto a otro fichero, el fichero archivo.

`C-c C-x C-a`

Archivar la actual entrada usando el comando especificado en la variable `org-archive-default-command`.

`C-c C-x C-s` o más corto `C-c $`

Archiva el subárbol empezando en la posición del cursor a la localización dada por `org-archive-location`.

La localización del archivo por defecto es un fichero en el mismo directorio como el actual fichero, con el nombre derivado por el sufijo `_archive` al actual nombre de fichero. Para información y ejemplos de cómo cambiar esto, ver la cadena de documentación de la variable `org-archive-location`. Hay también una opción de buffer para la configuración esta variable, por ejemplo

```
#+ARCHIVE: %s_done::
```

### Más lecturas

[Capítulo 9 del manual](#)

[Sebastian Rose's tutorial para capturar desde un servidor web](#)

## 10 Vistas de la Agenda

Debido a que Org funciona, los ítems TODO, los ítems de marca de tiempo y las cabeceras pueden dispersarse a través de un fichero o incluso un número de ficheros. Para tener una visión de ítems de acciones abiertas, o de eventos que son importantes para una fecha particular, esta información debe ser recogida, ordenada y mostrada en un camino organizado. Hay vistas varias vistas diferentes, ver debajo.

La información extraída es mostrada en un *búffer de agenda* especial. Este búffer es de solo lectura, pero proporciona comandos para visitar las localizaciones correspondientes en los ficheros Org originales, e incluso para editar estos ficheros remotamente. La edición remota desde los búffer de agenda significa, por ejemplo, que se pueden cambiar las fechas de entrega y citas desde el búffer de agenda. Los comandos disponibles en el búffer de Agenda son listados en [Sección 10.4 \[Comandos de la agenda\]](#), página 31.

### 10.1 Archivos de agenda

La información que se muestra está normalmente recogida desde todos los *ficheros de agenda*, los ficheros listados en la variable `org-agenda-files`.

- `C-c [`      Añade el fichero actual a la lista de ficheros de agenda. El fichero se añade al principio de la lista. Si ya estaba en la lista, se mueve al frente. Con un argumento de prefijo, el fichero se añade/mueve al final.
- `C-c ]`      Borra el fichero actual desde la lista de ficheros de agenda.
- `C-,`        Cambia a través de la lista de ficheros de agenda, visitando un fichero después del otro.

### 10.2 El disparador de la agenda

Las vistas se crean a través de un disparador, que debería ser emparejado a una tecla global — por ejemplo `C-c a` (véase [Sección 1.2 \[Instalación\]](#), página 1). Después de presionar `C-c a`, se necesita una tecla adicional para ejecutar un comando:

- `a`            La agenda como calendario (véase [Sección 10.3.1 \[Agenda semanal/diaria\]](#), página 29).
- `t / T`        Una lista de todos los ítems TODO (véase [Sección 10.3.2 \[Lista global TODO\]](#), página 29).
- `m / M`        Una lista de titulares coincidentes con la expresión TAGS (véase [Sección 10.3.3 \[Emparejar marcas y propiedades\]](#), página 30).
- `L`            La vista de la línea de tiempo para el búffer actual véase [Sección 10.3.4 \[Línea de tiempo\]](#), página 30).

- s** Una lista de entradas seleccionadas por una expresión booleana de palabras clave y/o expresiones regulares que deben o no deben coincidir con la entrada.

## 10.3 Las vistas internas de la agenda

### 10.3.1 Agenda semanal/diaria

El propósito de la *agenda* semanal/diaria es actuar como una página de una agenda de papel, mostrando todas las tareas para la semana o día actual.

- C-c a a** Compila una agenda para la semana actual desde una lista de fichero Org. La agenda muestra las entradas para cada día.

Emacs contiene el calendario y el diario por Edward M. Reingold. Org-mode comprende la sintaxis del diario y permite usar las entradas sexp directamente en ficheros Org:

```
* Cumpleaños y cosas similares
#+CATEGORY: Vacaciones
%(org-calendar-holiday) ; fuciones especiales para nombres de
fiestas
#+CATEGORY: Aniversario
%(diary-anniversary 5 14 1956)1 Arthur Dent tiene %d
años
%(diary-anniversary 10 2 1869) Mahatma Gandhi tendría
%d años.
```

Org puede interactuar con la funcionalidad de notificaciones de citas de Emacs. Para añadir todas la citas de tus ficheros de agenda, usa el comando `org-agenda-to-appt`. Ver la documentación para detalles.

### 10.3.2 La lista global TODO

La lista global TODO contiene todos los ítems TODO formateados y recogidos en un lugar simple. La edición remota de ítems TODO permite cambiar el estado de una entrada TODO con una simple tecla presionada. Los comandos disponibles en la lista TODO están descritos en la [Sección 10.4 \[Comandos de la agenda\]](#), página 31.

- C-c a t** Muestra la lista global TODO. Esta colección de ítems TODO de todos los archivos de la agenda (véase [Capítulo 10 \[Vistas de la Agenda\]](#), página 28) en un único buffer.
- C-c a T** Como abajo, pero permite la selección de una palabra clave TODO específica.

---

<sup>1</sup> Nótese que el orden de los argumentos (mes, día, año) depende de la configuración de `calendar-date-style`.



### 10.3.3 Emparejar marcas y propiedades

Si las líneas de cabecera en los ficheros de agenda están marcados con *etiquetas* (véase [Capítulo 6 \[Etiquetas\]](#), página 17), o tienen propiedades (véase [Capítulo 7 \[Propiedades\]](#), página 20), se pueden seleccionar líneas de cabecera basadas en estos metadatos y se recogen dentro del búffer de agenda. El emparejamiento de sintaxis descrito aquí también se aplica cuando se crean árboles encogidos con *C-c / m*. Los comandos disponibles en la lista de etiquetas están descritos en la [Sección 10.4 \[Comandos de la agenda\]](#), página 31.

- C-c a m*     Se produce una lista de todas las cabeceras que coinciden con un conjunto dado de etiquetas. El comando pregunta por un criterio de selección, que es una expresión de lógica booleana con etiquetas, como `'+work+urgent-withboss'` o `'work|home'` (véase [Capítulo 6 \[Etiquetas\]](#), página 17). Si con frecuencia se necesita una búsqueda específica, define un comando personalizado para ello (véase [Sección 10.2 \[El disparador de la agenda\]](#), página 28).
- C-c a M*     Como *C-c a m*, pero solo selecciona cabeceras que son también ítems TODO.

### Emparejar sintaxis

Una cadena de búsqueda puede usar operadores booleanos `'&'` para AND y `'|'` para OR. `'&'` es más fuerte que `'|'`. Los paréntesis no están actualmente implementados. Cada elemento en la búsqueda es una etiqueta, o una expresión como PROPIEDAD OPERADOR VALOR con un operador de comparación, que accede al valor de la propiedad. Cada elemento puede ser precedido por `'-'`, para eliminar la selección, y `'+'` que es azúcar sintáctico para la selección positiva. El operador AND `'&'` es opcional cuando `'+'` o `'-'` están presentes. Aquí hay algunos ejemplos, usando solo etiquetas (*tags*).

- `'+work-boss'`  
 Selecciona las líneas de cabecera etiquetadas como `':trabajo:'`, pero descarta estas también etiquetadas como `':jefe:'`.
- `'trabajo|portátil'`  
 Selecciona líneas etiquetadas como `':trabajo:'` o `':portátil'`.
- `'trabajo|escritorio+noche'`  
 Como antes, pero requiere las líneas `':portátil:'` para ser etiquetado también como `':noche:'`.

Se pueden también chequear propiedades al mismo tiempo que se chequea si coinciden etiquetas (*tags*), ver el manual para más información.

### 10.3.4 Línea de tiempo para un simple archivo

La línea de tiempo resume todos los ítems de marca de tiempo desde un simple fichero Org mode en una *vista de tiempo ordenada*. El propósito prin-

cial de este comando es dar una vista general a través de eventos en un proyecto.

*C-c a L* emparejando

### 10.3.5 Vista de búsqueda

Esta vista de agenda es una facilidad de búsqueda general de texto para entradas Org mode. Eso es particularmente útil para encontrar notas.

*C-c a s* Esto es una búsqueda especial que te permite seleccionar entradas emparejando una subcadena o palabras específicas usando lógica booleana.

Por ejemplo, la cadena de búsqueda ‘`equipamiento del ordenador`’ encontrará entradas que contienen ‘`equipamiento del ordenador`’ como una subcadena. La búsqueda también puede encontrar palabras clave específicas en la entrada, usando lógica booleana. La cadena de búsqueda ‘`+ordenador +wifi -ethernet -{8.11[bg]}`’ encontrará entradas que contengan las palabras clave `ordenador` y `wifi`, pero no las palabras clave `ethernet`, y que tampoco concuerdan con la expresión regular `8.11[bg]`, que significa excluir tanto `8.11b` y `8.11g`.

Nótese que además a los ficheros de agenda, este comando también buscará los ficheros listados en `org-agenda-text-search-extra-files`.

## 10.4 Comandos en el búffer de agenda

Las entradas en el búffer de agenda están enlazadas al fichero Org o el fichero de diario dónde ellos lo originan. Los comandos están provistos para mostrar y saltar a la localización de la entrada original, y editar los ficheros Org “remotamente” desde el búffer de la agenda. Esto es solo una selección de los muchos comandos, explora el menú de la *Agenda* y el manual para una lista completa.

### Movimiento

*n* Siguiete línea (igual que *arriba* y *C-p*).

*p* La línea previa (igual que *abajo* y *C-n*).

### Vista/Ir al fichero Org

*mouse-3*

SPC Muestra la localización original del ítem en otra ventana. Con el prefijo *arg*, asegúrese que la entrada entera se hace visible en el outline y, no solo en la cabecera.

TAB Ir a la localización original de los ítems en otra ventana. Bajo Emacs 22 *mouse-1* también funciona para esto.

RET Ir a la localización original del ítem y borra otras ventanas.

### Cambiar la visualización

*o* Borra otras ventanas.

- d / w* Cambiar a la vista de día/semana.
- f y b* Ir hacia delante o hacia atrás en el tiempo para mostrar los siguientes días `org-agenda-current-span`. Por ejemplo, si el display cubre una semana, cambia la siguiente o previa semana.
- .* Ir a hoy.
- j* Pregunta por una fecha y va allí.
- v l* u ordena *l*  
Cambia al modo Logbook. En el modo Logbook, las entradas que se marcan como DONE (dada la variable `org-log-done`), se muestran en la agenda, como entradas que han sido organizadas en este día. Cuando se llama con un prefijo `C-u`, se muestran todas las posibles entradas logbook, incluyendo cambios estado.
- r o g* Recrear el búffer de agenda, para reflejar los cambios.
- s* Salva todos los búffers Org en la sesión Emacs actual, y también las localizaciones de IDs.

### Filtro secundario y buscar la edición

- /* Filtra la vista de agenda actual con respecto a una etiqueta. Se pregunta por una letra para seleccionar una etiqueta. Presiona '-' primero para seleccionar la etiqueta.
- \* Minimiza el filtro de la agenda actual por una condición adicional.

### Edición remota (ver el manual para muchos más comandos)

- 0--9* Argumento dígito.
- t* Cambia el estado TODO del ítem, en la agenda en el fichero org.
- C-k* Borra el ítem de la agenda actual a lo largo con el subárbol perteneciente a ello en el fichero Org original.
- C-c C-w* Rellena la entrada a punto.
- C-c C-x C-a* u ordena *a*  
Archivar el subárbol correspondiente a la entrada a punto usando el comando de archivación establecido en `org-archive-default-command`.
- C-c C-x C-s* u ordena *\$*  
Archiva el subárbol correspondiente a la actual cabecera.
- C-c C-s* Planifica este ítem, con argumento prefijo borra la marca de tiempo planificada.
- C-c C-d* Asigna una fecha límite para este ítem, con el argumento prefijo elimina la fecha límite.

*S-derecha/izquierda*

Cambia la marca de tiempo asociada con la línea actual por un día.

*I* Empieza el cronómetro en el ítem actual.

*O / X* Para/cancela el reloj previamente iniciado.

*J* Salta al reloj en ejecución a otra ventana.

## 10.5 Vistas de agenda personalizadas

La principal aplicación de las búsquedas personalizadas es la definición de atajos de teclado para búsquedas usadas frecuentemente, bien creando un búffer de agenda, o un árbol disperso (el último que cubre de acuerdo solo al búffer actual). Los comandos personalizados son configurados en la variable `org-agenda-custom-commands`. Se puede personalizar esta variable, por ejemplo, presionando `C-c a C`. Se puede también directamente establecerlo con Emacs Lisp en `.emacs`. El siguiente ejemplo contiene todos los tipos de búsqueda válida:

```
(setq org-agenda-custom-commands
      '(("w" todo "ESPERANDO")
        ("u" tags "+jefe-urgente")
        ("v" tags-todo "+jefe-urgente")))
```

La cadena inicial en cada entrada define las claves que se tienen para presionar después el comando disparador `C-c a` en orden para acceder el comando. Usualmente esto será solo un carácter. El segundo parámetro es el tipo de búsqueda, seguida por la cadena o expresión regular para ser usada por el emparejamiento. El ejemplo de arriba además definirá:

`C-c a w` como una búsqueda global para entradas `TODO` con ‘`ESPERANDO`’ como la palabra clave `TODO`

`C-c a u` como una búsqueda global de etiquetas para líneas de cabecera ‘`:jefe:`’ pero no ‘`:urgente:`’

`C-c a v` como la misma búsqueda `C-c a u`, pero limitando la búsqueda para líneas de cabecera están también ítems `TODO`

### Más lecturas

Capítulo 10 del manual

El tutorial de Mat Lundin acerca de comandos personalizados de agenda

Configuración de John Wiegley

# 11 Marcas para enriquecer la exportación

Cuando se exportan documentos en Org-mode, el exportador intenta reflejar la estructura del documento de manera tan precisa como sea posible en el backend. Exportar a formatos como HTML, L<sup>A</sup>T<sub>E</sub>X, o DocBook permite enriquecer el formatos, Org mode tiene reglas de cómo preparar texto para la exportación enriquecida. Esta sección resume las reglas de marcado usados en un búffer Org-mode.

## 11.1 Elementos del marcado estructural

### Título de documento

El título del documento exportado es tomado desde la línea especial

```
#+TITLE: Este es el título del documento
```

### Encabezados y secciones

La estructura de contorno del documento como se describe en el [Capítulo 2 \[Estructura del documento\]](#), [página 3](#), forma las bases para las secciones de definición de los documentos exportados. Sin embargo, desde la estructura de *outline* también se usa para (por ejemplo) listas de tareas, solo los primeros niveles del árbol de *outline* se usan como cabeceras. Los niveles profundos llegarán a ser listas de ítems. Se puede cambiar la localización de este cambio globalmente configurando la variable `org-export-headline-levels`, o en una base por fichero con una línea.

```
#+OPTIONS: H:4
```

### Tabla de contenidos

La tabla de contenidos está normalmente insertados directamente antes de la primera línea de cabecera del fichero.

```
#+OPTIONS: toc:2          (solo a dos niveles en TOC)
#+OPTIONS: toc:nil       (no TOC en todo)
```

### Párrafos, líneas rotas y citación

Los párrafos están separados por al menos una línea vacía. Si se necesita forzar una línea rota con un párrafo, usa ‘\’ al final de una línea.

Para guardar la línea rota en una región, pero usar formateo normal se puede usar este constructo, que puede también ser usado para el formato poesía.

```
#+BEGIN_VERSE
Grandes tormentas agitan los aires
Nubes oscuras nos impiden ver
Hacia el Emacs llama el deber
-- DavidArroyo
#+END_VERSE
```

Cuando se cita un pasaje desde otro documento, es un libro de usos y costumbres para formatear esto como un párrafo que se pretende en ambos la izquierda y el margen derecho. Se puede incluir citas en documentos Org-mode como este:

```
#+BEGIN_QUOTE
Cada cosa debería ser hecha tan simple como sea posible,
pero no simple de cualquier manera -- Albert Einstein
#+END_QUOTE
```

Si se quisiera centrar algún texto, se puede hacer así:

```
#+BEGIN_CENTER
Cada cosa debería hecha tan simple como posible \\
pero no simple cualquier manera
#+END_CENTER
```

## Énfasis y monoespacio

Se pueden crear palabras ***=code=***, */italic/*, underlined, `=code=` and `~verbatim~`, y, se debe, `+strike-through+`. Texto en el código y cadena verbatim no está procesada para una sintaxis específica de Org-mode, ella es una copia literal exportada. Para insertar reglas horizontales, usan un línea que consista de solo dashes, y al menos 5 de ellos.

## Líneas de comentarios

Las líneas empezando con cero o más caracteres de espacios en blanco seguidos por ‘#’ y un espacio en blanco se tratan como comentarios y, como tales, no se exportan.

También, regiones rodeadas por ‘#+BEGIN\_COMMENT’ ... ‘#+END\_COMMENT’ no se exportan.

Finalmente, una palabra clave ‘COMMENT’ al principio de una entrada, pero después de cualquier otra palabra clave o prioridad, los comentarios están fuera del subárbol entero. El comando de debajo ayuda cambiando el estado del comentario de una cabecera.

`C-c ;`      Cambia la palabra clave COMMENT al principio de una entrada.

## 11.2 Imágenes y Tablas

Para tablas Org mode, las líneas antes de la primera línea de separador horizontal llegarán a ser una tabla de líneas cabecera. Se pueden usar las siguientes líneas en algún lugar antes de la tabla para asignar un caption y una etiqueta para referencias cruzadas, y en el texto se puede referir para el objeto con `\\ref{tab:basic-data}`:

```
#+CAPTION: Este es el título para la siguiente tabla (o enlace)
#+LABEL:   tbl:basic-data
| ... | ... |
|-----|-----|
```

Algunos lenguajes de marcado permiten directamente incluir imágenes dentro del documento exportado. Org hace esto, si un enlace a ficheros de imágenes no tiene una parte de descripción, por ejemplo `[[./img/a.jpg]]`. Si se desea definir una captura para la imagen y puede ser una etiqueta para referencias cruzadas internas, se debe asegurar que el enlace está en una línea por sí misma precedida con:

```
#+CAPTION: Este es el título para el siguiente enlace fig (o tabla)
#+NAME:   fig:SED-HR4049
[[./img/a.jpg]]
```

El mismo mecanismo de captura se aplica a otras estructuras en vez de imágenes y tablas (por ej: ecuaciones L<sup>A</sup>T<sub>E</sub>X, bloques de código), provistos por la exportación elegida y el lenguaje de marcado que los soporta.

### 11.3 Ejemplos literales

Se pueden incluir ejemplos literales que no deberían ser asunto de marcado. Tales ejemplos serán tipografiados en monospacio, así está bien situado para el código fuente y ejemplos similares.

```
#+BEGIN_EXAMPLE
Algún ejemplo desde un fichero de texto.
#+END_EXAMPLE
```

Por simplicidad cuando se usan ejemplos pequeños, se puede también empezar líneas con dos puntos seguidos por un espacio. También puede haber espacios en blanco adicionales antes de los dos puntos:

```
Aquí está un ejemplo
  : Algún ejemplo desde un fichero de texto.
```

Para código fuente desde un lenguaje de programación, o cualquier otro texto que puede ser marcado por bloque de fuentes en Emacs, se puede preguntar para ello para mirar como el búffer de Emacs fontificado.

```
#+BEGIN_SRC emacs-lisp
(defun org-xor (a b)
  "or exclusiva."
  (if a (not b) b))
#+END_SRC
```

Para editar el ejemplo en un búffer especial soportando este lenguaje, usa `C-c` ' para ambos entrar y dejar el búffer de edición.

### 11.4 Archivos Include

Durante la exportación, se pueden incluir el contenido de otro fichero. Por ejemplo, para incluir tu fichero `.emacs`, se podría usar:

```
#+INCLUDE: "~/emacs" src emacs-lisp
```

Los parámetros opcionales segundo y tercero son la marca (e.g. 'ejemplo', o 'fuente'), y, si la marca es 'fuente', el lenguaje formatea los contenidos. La marca es opcional, si no es dada, el texto se asume para estar en el formato

de Org mode y será procesado normalmente. Los enlaces de fichero serán interpretados así:

```
#+INCLUDE: "./otherfile.org::#my_custom_id" :only-contents t
C-c ' visitará el fichero incluido.
```

## 11.5 $\LaTeX$ embebido

Para notas científicas que necesitan ser capaces de contener símbolos matemáticos y la fórmula ocasional, Org-mode soporta código  $\LaTeX$  embebido dentro de sus ficheros. Se puede directamente usar sintaxis Tex para símbolos especiales, e introducir fórmulas y entornos  $\LaTeX$  completos.

```
Los ángulos son escritos como letras Griegas \alpha, \beta y
\gamma. La masa si el sol es M_sol = 1.989 x 10^30 kg. El radio del
sol es R_{sol} = 6.96 x 10^8 m. Si $a^2=b$ y $b=2$, entonces la
solución debe ser $a=+\sqrt{2}$ o $a=-\sqrt{2}$.\begin{equation}
x=\sqrt{b}
\end{equation}
```

Con **configuración especial**, trozos de  $\LaTeX$  serán incluidos como imágenes cuando se exporta a HTML.

**Más lecturas**

Capítulo 11 del manual



## 12 Exportando

Los documentos Org-mode pueden exportarse a una variedad de otros formatos: ASCII para su inclusión en correos electrónicos, HTML para su publicación en la web, L<sup>A</sup>T<sub>E</sub>X/PDF para bellos documentos impresos y DocBook para entrar a un mundo de muchos formatos. Además puede exportarse al formato iCalendar, así la información de planificación puede ser incorporada a calendarios de escritorio.

### 12.1 Opciones de exportación

El exportador reconoce líneas especiales en el búffer que proporcionan información adicional. Estas líneas pueden ser puestas en cualquier lugar del fichero. El conjunto completo de líneas que pueden ser insertadas dentro del búffer con *C-c C-e #*.

*C-c C-e #* Inserta plantilla con opciones de exportaciones, ver ejemplo de debajo.

```

#+TITLE:          el título que será mostrado
#+AUTHOR:         el autor (por defecto tomado desde user-full-name)
#+DATE:           una fecha, fija, o una marca de tiempo Org
#+EMAIL:          su dirección de correo (por defecto desde user-mail-address)
#+DESCRIPTION:    la descripción de la página e.g. para la meta etiqueta XHTML
#+KEYWORDS:       la página de palabras clave ('keywords'), e.g. para la meta eti-
queta XHTML
#+LANGUAGE:       lenguaje, por ej. 'en' (org-export-default-language)
#+OPTIONS:        H:2 num:t toc:t \n:nil @:t |:t ^:t f:t teX:t ...

```

### 12.2 El dispensador de exportación

Todos los comandos de exportación se pueden lograr usando el dispensador o disparador de exportación, que es una clave prefija que muestra una clave adicional especificando el comando. Normalmente el fichero entero es exportado, pero si hay una región activa también se exportará.

*C-c C-e* Disparador para exportar y publicar comandos.

### 12.3 Exportación ASCII/Latin-1/UTF-8

La exportación ASCII produce una simple y muy legible versión de un fichero Org-mode, conteniendo solo ASCII plano. Con Latin-1 y UTF-8 se exporta aumentando el fichero con caracteres especiales y símbolos disponibles en estas codificaciones.

*C-c C-e t a* y *C-c C-e t A*  
Exportar como un fichero ASCII o búffer temporal.

*C-c C-e t n* y *C-c C-e t N*  
Como los comandos de arriba, pero usa codificación Latin-1.

*C-c C-e t u* y *C-c C-e t U*

Como los comandos de arriba, pero usa codificación UTF-8.

## 12.4 HTML export

*C-c C-e h h*

Exportar como un fichero HTML `myfile.html`.

*C-c C-e h o*

Exportar como fichero HTML e inmediatamente lo abre con un navegador.

Para insertar el HTML que debe copiarse literalmente para al fichero exportado que usa otro

```

#+HTML: Código Literal HTML para exportar
o
#+BEGIN_HTML
Todas las líneas entre estos marcadores son exportados
literalmente
#+END_HTML

```

## 12.5 Exportar $\LaTeX$ y PDF

*C-c C-e l l*

Exporta como fichero  $\LaTeX$  `myfile.tex`.

*C-c C-e l p*

Exportando a  $\LaTeX$  y procesando a PDF

*C-c C-e l o*

Exportando a  $\LaTeX$  y procesando a PDF, entonces se abre el fichero PDF resultante.

Por defecto, la salida  $\LaTeX$  usa la clase `article`. Se puede cambiar esto para añadir una opción como `#+LaTeX_CLASS: myclass` en su fichero. La clase debe ser listada en `org-export-latex-classes`.

El  $\LaTeX$  embebido como se describe en [Sección 11.5 \[ \$\LaTeX\$  embebido\]](#), [página 37](#), será correctamente insertado dentro del fichero  $\LaTeX$ . De manera similar, al exportador HTML, se puede usar `#+LaTeX:` y la construcción `#+BEGIN_LaTeX ... #+END_LaTeX` para añadir código  $\LaTeX$  literal.

## 12.6 Exportar iCalendar

*C-c C-e c f*

Crea entradas iCalendar para el fichero actual en un fichero `.ics`.

*C-c C-e c c*

Crea un fichero simple pero largo iCalendar de todos los ficheros en `org-agenda-files` y los escribe al fichero dado por `org-icalendar-combined-agenda-file`.

### **Más lecturas**

Capítulo 12 del manual

Tutorial de Sebastian Rose de manejo de imágenes

Tutorial de exportación de LaTeX de Thomas Dye Tutorial de presentaciones de BEAMER de Eric Fraga

## 13 Publicación

Org incluye un sistema de gestión de publicación que permite configurar la conversión HTML de *proyectos* compuestos de ficheros org entrelazados. Se puede también configurar Org para automáticamente subir las páginas HTML y adjuntos relacionados, tales como imágenes y ficheros de código fuente, a un servidor web. Para instrucciones detalladas acerca de la configuración, ver el manual.

Aquí hay un ejemplo:

```
(setq org-publish-project-alist
  '(("org"
     :base-directory "~/org/"
     :publishing-directory "~/public_html"
     :section-numbers nil
     :table-of-contents nil
     :style "<link rel=\"stylesheet\"
           href=\"../other/mystyle.css\"
           type=\"text/css\"/>")))
```

*C-c C-e P x*

Pregunta por un proyecto específico y publica todos los ficheros que pertenecen a él.

*C-c C-e P p*

Publica el proyecto que contiene el fichero actual.

*C-c C-e P f*

Elimina la columna actual.

*C-c C-e P a*

Publica cada proyecto

Org usa marcas de tiempo para trazar cuando un fichero ha cambiado. Las funciones de debajo normalmente solo publican ficheros modificados. Se puede sobrescribir esto y forzar la publicación de todos los ficheros dando un argumento prefijo para cualquiera de los comandos de debajo.

### Más lecturas

[Capítulo 13 del manual](#)

[Tutorial de publicación de Sebastian Rose](#)

[Configuración de Jekyll de Ian Barton](#)

## 14 Trabajando con código fuente

Org-mode proporciona un número de funcionalidades para trabajar con código fuente, incluyendo la edición de bloques código en sus modo mayor nativo, evaluación de bloques código, estructurar de bloques de código, y exportar bloques de código y sus resultados en varios formatos.

### Estructura de bloques de código

La estructura de bloques de código es como sigue:

```
#+NAME: <name>
#+BEGIN_SRC <language> <switches> <header arguments>
  <body>
#+END_SRC
```

Donde `<name>` es una cadena usada para nombrar el bloque de código `<language>` que especifica el lenguaje del bloque de código (por ej. `emacs-lisp`, `shell`, `R`, `python`, etc..., `<switches>` puede usarse para controlar la exportación del bloque de código, `<header arguments>` puede usarse para controlar muchos aspectos del comportamiento de bloques de código como se demuestra debajo, y el `<body>` contiene el código fuente actual.

### Editando código fuente

Usa `C-c '`  para editar el actual bloque de código. Esto trae un lenguaje en modo mayor que edita el búffer conteniendo el cuerpo del bloque de código. Salvando este búffer escribirá los nuevos contenidos atrás al búffer de Org. Usa `C-c '`  atrás para salir del búffer de edición.

### Evaluando códigos de bloques

Use `C-c C-c` para evaluar el actual bloque código e inserte sus resultados en el búffer de Org-mode. Por defecto, la evaluación solo cambia para bloques de código `emacs-lisp`, sin embargo, el soporte existe para evaluar bloques en muchos lenguajes. Para una lista completa de lenguajes ver el manual. Lo siguiente muestra un bloque de código y sus resultados.

```
#+BEGIN_SRC emacs-lisp
  (+ 1 2 3 4)
#+END_SRC

#+RESULTS:
: 10
```

### Extrayendo código fuente

Usa `C-c C-v t` para crear ficheros de código fuente puro para extraer código desde bloques fuente en el búffer actual. Esto es referido como *tagling*

“enredo”— un término adoptado desde la comunidad de programación litera-  
 l. Durante “el enredo” de bloques código sus cuerpos están expandidas  
 usando `org-babel-expand-src-block` que puede expandir ambas variables  
 y referencias de estilo “noweb”. En orden para enredar un bloque de código  
 ello debe tener un argumento de cabecera de `:tagling`, ver el manual para  
 detalles.

## Librería de Babel

Usa `C-c C-v l` para cargar los bloques de código desde ficheros Org-mode  
 dentro de la “Librería de Babel”, estos bloques pueden ser evaluados desde  
 cualquier búffer Org-mode. Una colección de generalmente bloques de código  
 útiles están distribuidos con Org-mode en `contrib/library-of-babel.org`.

## Argumentos de cabecera

Muchos aspectos de la evaluación y exportación de bloques código están con-  
 trolados a través de argumentos de cabecera. Estos pueden ser especificados  
 globalmente, en el nivel del fichero, en el nivel de perfil de subárbol, y en  
 el nivel de código de bloque individual. Lo siguiente describe algunos de los  
 argumentos de cabecera.

- `:var` El argumento de cabecera `:var` es usado para pasar argumentos  
 a bloques de código. Los valores pasados a argumentos pueden  
 ser valores literales, valores desde tablas org-mode y ejemplos  
 de bloque literales, o los resultados de otros bloques de código  
 nombrados.
- `:results` El argumento de cabecera `:results` controla el *collection*, *type*,  
 y *handling* de resultados de bloques de código. Valores de `output`  
 o `value` (por defecto) especifican como los resultados son recogi-  
 dos desde la evaluación de bloques de código. Valores de `vector`,  
`scalar`, `file`, `raw`, `html`, `latex` and `code` especifican el tipo de  
 resultados del bloque código que dicta cómo serán incorpora-  
 dos dentro del buffer Org-mode. Valores de `silent`, `replace`,  
`prepend`, y `append` especifica el manejo de resultados de bloques  
 de código, específicamente si y cómo los resultados deberían ser  
 insertados dentro del búffer Org-mode.
- `:session` Un argumento de cabecera de `:session` causará el bloque de  
 código para ser evaluada en un proceso inferior persistente en  
 Emacs. Esto permite el estado persistente entre evaluaciones de  
 bloque de código, y para inspección de manual de los resultados  
 de evaluación.
- `:exporta` Cualquier combinación del *code* ó los *resultados* de un bloque  
 puede ser retenido en exportar, esto está especificado por config-  
 uración de los `:results` de argumentos de cabecera para *código*  
*resultados ninguno* ó *ambos*.

- :tangle** Un argumento de cabecera de **:tangle yes** causará de bloques de código se enreda a un fichero llamado después del nombre del fichero del búffer Org-mode. Un nombre de fichero alternativo puede ser especificado con **:tangle filename**.
- :cache** Un argumento de cabecera de **:cache yes** causará asociar un hash del bloque código expandido con los resultados, asegurando que los bloques de código están solo rearrancados con sus entradas han cambiado.
- :noweb** Un argumento de cabecera de **:noweb yes** expandirá “noweb” referencias de estilo en evaluación y enredo.
- :file** Los bloques de código cuya salida resulta a ficheros (por ej. grafos, diagramas y figuras) puede aceptar un **:file nombre de fichero** argumento de cabecera en cuyo caso los resultados están a salvo para el fichero nombrado, y un enlace al fichero se insertan dentro del búffer Org-mode.

### Más lecturas

Capítulo 11.3 del manual

El sitio Babel en Worg

## 15 Miscelánea

### 15.1 Terminación

Org soporta autocompleción en búffer con *M-TAB*. Este tipo de autocompletado no hace uso del minibúffer. Simplemente escribe unas pocas letras dentro del búffer y usa la clave para completar texto correctamente. Por ejemplo, este comando completará símbolos  $\TeX$  después ‘\’, palabras clave TODO al principio de la línea de cabeza, y etiquetas después de ‘:’ en una línea de cabeza.

### 15.2 Una vista del contorno de limpieza

Algunas personas encuentran ruidoso y distraído que las cabeceras Org empiecen con un número potencialmente largo de estrellas, y que el texto de debajo de las líneas de cabecera no están indentadas. Mientras esto no es problema cuando se escribe un documento *tipo libro* donde el contorno de cabeceras son realmente cabeceras de sección, en una idea general *orientada a listas*, la estructura indentada es muy limpia:

* Cabecera de alto nivel		* Cabecera de alto nivel
** Segundo nivel		* Segundo nivel
*** Tercer nivel		* Tercer nivel
algún texto		algún texto
*** Tercer nivel		* Tercer nivel
más texto		más texto
* Otra cabecera de alto nivel		* Otra cabecera de alto nivel

Si se está usando al menos Emacs 23.1.50.3 y la versión 6.29 de Org, este tipo de vista puede ser lograda dinámicamente

```
#+STARTUP: indentar
```

Si se quiere un efecto similar en una versión temprana de Emacs y/o Org, o si se quiere la indentación para endurecer caracteres de espacio, así que el texto plano del fichero se vea tan similar como sea posible para la visualización de Emacs, Org soporta ayudar para indentar texto (con TAB) debajo de cada cabecera, ocultando estrellas destacadas, y solo usando niveles 1, 3, etc. para obtener dos caracteres de indentación para cada nivel. Para tener este soporte en un fichero, use:

```
#+STARTUP: oculta estrellas impares
```

### 15.3 Org móvil

*MobileOrg* es el nombre de la compañía de móviles para Org mode actualmente disponible para iOS y para Android. *MobileOrg* ofrece una vista offline y soporta captura para un sistema Org mode ruteado en un ordenador “real”. Esto permite grabar cambios a entradas existentes.

La **implementación iOS** para la serie de dispositivos *iPhone/iPod Touch/iPad*, fué por Richard Moreland. Los usuarios de Android



deberían descargar [MobileOrg Android](#) hecho por Matt Jones. Las dos implementaciones no son idénticas pero ofrecen funcionalidades similares.

**Más lecturas**

[Capítulo 15 del manual](#)

[Apéndice B del manual](#)

[tarjeta de referencia](#)

# Apéndice A GNU Free Documentation License

Version 1.3, 3 November 2008

Copyright © 2000, 2001, 2002, 2007, 2008, 2013 Free Software Foundation, Inc.

<http://fsf.org/>

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 worldwide, 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 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.

The “publisher” means any person or entity that distributes copies of the Document to the public.

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.

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.

#### 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.

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.

## 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 under this License. Any attempt otherwise to copy, modify, sublicense, or distribute it is void, and will automatically terminate your rights under this License.

However, if you cease all violation of this License, then your license from a particular copyright holder is reinstated (a) provisionally, unless and until the copyright holder explicitly and finally terminates your license, and (b) permanently, if the copyright holder fails to notify you of the violation by some reasonable means prior to 60 days after the cessation.

Moreover, your license from a particular copyright holder is reinstated permanently if the copyright holder notifies you of the violation by some reasonable means, this is the first time you have received notice of violation of this License (for any work) from that copyright holder, and you cure the violation prior to 30 days after your receipt of the notice.

Termination of your rights under this section does not terminate the licenses of parties who have received copies or rights from you under this License. If your rights have been terminated and not permanently reinstated, receipt of a copy of some or all of the same material does not give you any rights to use it.

## 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 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. If the Document specifies that a proxy can decide which future versions of this License can be used, that proxy’s public statement of acceptance of a version permanently authorizes you to choose that version for the Document.

## 11. RELICENSING

“Massive Multiauthor Collaboration Site” (or “MMC Site”) means any World Wide Web server that publishes copyrightable works and also provides prominent facilities for anybody to edit those works. A public wiki that anybody can edit is an example of such a server. A “Massive Multiauthor Collaboration” (or “MMC”) contained in the site means any set of copyrightable works thus published on the MMC site.

“CC-BY-SA” means the Creative Commons Attribution-Share Alike 3.0 license published by Creative Commons Corporation, a not-for-profit corporation with a principal place of business in San Francisco, California, as well as future copyleft versions of that license published by that same organization.

“Incorporate” means to publish or republish a Document, in whole or in part, as part of another Document.

An MMC is “eligible for relicensing” if it is licensed under this License, and if all works that were first published under this License somewhere other than this MMC, and subsequently incorporated in whole or in part into the MMC, (1) had no cover texts or invariant sections, and (2) were thus incorporated prior to November 1, 2008.

The operator of an MMC Site may republish an MMC contained in the site under CC-BY-SA on the same site at any time before August 1, 2009, provided the MMC is eligible for relicensing.

## ADDENDUM: 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 1.3  
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.