El manual de GNU TeXmacs

Table of contents

Chapter 1

Iniciar

1.1. Convenciones para este manual

A través del manual de $\text{T}_{\text{E}}\text{X}_{\text{MACS}}$, las entradas de menu serán escritas usando una fuente *sans serif*, como en Document, File → Load o Text → Font shape → Italic. Las entradas de teclado serán escritas en fuente de *máquina de escribir* dentro de cajas, como en in `C-s`. En la parte lateral izquierda de las entradas de menu, verá el equivalente de teclado, cuando estos estén disponibles. Las siguientes abreviaciones son usadas para tales teclas:

`S-`. Para combinaciones con la tecla shift.

`C-`. Para combinaciones con la tecla control.

`A-`. Para combinaciones con la tecla alternate.

`M-`. Para combinaciones con la tecla meta.

`H-`. Para combinaciones con la tecla hyper.

Por ejemplo, `A-C-b` representa `alt-ctrl-b`. Epacios dentro de los atajos de teclado indican múltiple presiones de tecla. Por ejemplo, `M-t N b` representa `meta-t` `N` `b`.

Las teclas `alt`, `meta` e `hyper` no están disponibles en todos los teclados. En los PC's recientes, la tecla `meta` es a menudo reemplazada por la tecla `windows`. In the case when one or several modifier keys are missing on your keyboard, you may use `escape` instead of `M-`, `escape escape` instead of `A-` and `F5`, `escape escape escape` or `A-C-` instead of `H-`. For instance, `escape w` is equivalent to `A-w`. Uste puede también configurar los modificadores de teclado a fin de tomar ventaja total del poderoso conjunto de atajos de teclado que es proveido by $\text{T}_{\text{E}}\text{X}_{\text{MACS}}$.

Note que el comportamiento de los menus de $\text{T}_{\text{E}}\text{X}_{\text{MACS}}$ y el teclado son *contextuales*, e.d. ellos dependen del modo actual (e.j. mode texto o "modo matemático"), el lenguaje actual y la posición del cursor dentro de su documento. Por ejemplo, dentro del modo matemático, usted tiene atajos de teclado especiales los cuales son útiles para teclear fórmulas matemáticas, pero inútiles en modo texto

1.2. Configurar $\text{T}_{\text{E}}\text{X}_{\text{MACS}}$

Cuando inicie $\text{T}_{\text{E}}\text{X}_{\text{MACS}}$ por primera vez, el programa automáticamente se configura a sí mismo en la forma en la cual piensa ser más apropiado para usted. Por ejemplo, $\text{T}_{\text{E}}\text{X}_{\text{MACS}}$ intentará determinar las configuraciones de su sistema y el tipo de papel en su impresora. Sin embargo, la configuración automática algunas veces falla o puede quere usar una configuración alternativa. En ese caso, debería ir al menu Edit → Preferences y especificar sus preferencias.

En particular, le recomendamos configurar el "aspecto y comportamiento" deseado de $\text{T}_\text{E}\text{X}_\text{MACS}$. Por defecto usamos el aspecto y comportamiento de EMACS, que asegura compatibilidad limitada de los atajos de teclado de $\text{T}_\text{E}\text{X}_\text{MACS}$ con aquellos EMACS. También, $\text{T}_\text{E}\text{X}_\text{MACS}$ viene con un poderoso sistema de atajos de teclado, el cual intenta optimizar el uso de modifcadores como `shift` y `control` en su teclado. Sin embargo, en muchos sistemas X Window esos modificadores no están bien configurados, así que puede desear rehacer esto por usted mismo. Más detalles pueden ser encontrados en la sección configuración of $\text{T}_\text{E}\text{X}_\text{MACS}$.

1.3. Crear, guardar y cargar documentos

Cuando se lanza $\text{T}_\text{E}\text{X}_\text{MACS}$ sin ninguna opción de línea de comandos, el editor automáticamente crea un nuevo documento para usted. Puede también crear un documento por usted mismo usando File → New. Los documentos creados como nuevos no tienen aún un nombre. A fin de darles un nombre, debe hacer click en File → Save as.

Recomendamos dar un nombre a los documentos inmediatamente después de su creación; esto evitará que usted pierda documentos. Es también importante especificar la configuración globarl para su documento cuando sera necesario. Primero que todo, usted debe especificar un estilo de documento, como artículo, libro o seminario usando Document → Style. Si escribe documentos en varios lenguajes, entonces puede querer espeficiar el lenguaje de su documento usando Document → Language. Similarmente, puede especificar un tipo de paepl usando Document → Page → Size.

Después de moficar su documento, puede guardarlo usando File → Save. Los documentos viejos pueden ser recuperados usando File → Load. Note que puede editar varios documentos en la misma ventana usando $\text{T}_\text{E}\text{X}_\text{MACS}$; puede cambiar entre diferentes *buffers* usando Go.

1.4. Imprimir documentos

Puede imprimir el fichero actual usando Fichero → Imprimir → Imprimir todo. Por defecto $\text{T}_\text{E}\text{X}_\text{MACS}$ supone que tiene una impresora de 600dpi para papel a4. Esta configuración por defecto puede ser cambiada en Preferences → Printer. También puede imprimir a un archivo Postscript usando Fichero → Imprimir → Imprimir todo en un fichero (en cuyo caso la configuración por defecto de la impresora se utilizan para generar la salida) o Fichero → Exportar → Postscript (en cuyo caso la configuración de la impresora es ignorada).

Cuando se configura adecuadamente $\text{T}_\text{E}\text{X}_\text{MACS}$, el editor esta garantizado para ser *wysiwyg*[1.1]: el resultado de la salida de impresión es exactamente lo que usted ve en su pantalla. A fin de obtener por completo esta característica, debe en particular seleccionar Document → Page → Type → Paper y Document → Page → Screen layout → Margins as on paper. Debe también asegurarse de que los caracteres en la pantalla tienen el mismo número de punto por pulgada que su impresora (dpi, por su sigla en inglés). Esta precisión de visualización de los caracteres puede ser cambiada usando Document → Font → Dpi. Actualmente, cambios menoes pueden ocurrir cuando se cambia los dpi, que pueden afectar globalmente el documento a través del salto de línea y de página. En una versión futura este inconveniente debería ser removido.

1.1. *wysiwyg* se refiere a las siglas en inglés para *What You See Is What You Get*, es decir, Lo que ves es lo que tienes.

Chapter 2

Escribir documentos sencillos

2.1. Generalidades para teclear de texto

Tan pronto como haya realizado las acciones preparatorias explicadas arriba, puede iniciar a teclear. Los caracteres y símbolos de puntuación pueden ser fácilmente obtenidos en la mayoría de los teclados. Los caracteres acentuados pueden ser sistemáticamente obtenidos usando la tecla de escape. Por ejemplo "é" es obtenida tecleando `A-' e`. Similarmente, obtenemos "à" via `A-‘ a` y así. Las palabras largas de líneas sucesivas son automáticamente separadas silábicamente. A fin de obtener correctamente separación silábica para lenguajes extrageros, debe especificar el lenguaje del documento en el menu Document → Language.

En la parte inferior de la ventana a mano izquierda, ve el estilo del documento, las propiedades del texto en la posición actual. Inicialmente, se muestra "texto roman 10", lo que significa que puede teclear en modo texto usando una fuente roman de 12 puntos y sin estilo de documento. Puede cambiar las propiedades de texto (fuente, tamaño de fuente, color lenguaje) en el menu Text. Puede también cambiar las propiedades del texto que ya ha tecleado seleccionando una región y luego cambiar las propiedades del texto en el menu Text. Algunas propiedades del texto pueden también ser cambiadas para todos el documento en los menus Document → Font y Document → Language,

En la parte inferior de la ventana a mano derecha, se se muestra el carácter o la acción (por ejemplo, un cambio en las propiedades del texto) justo antes del cursor. También se muestran todos los entornos que están activos en la posición del cursor. Esta información debería ayudarlo a orientarse por sí mismo en el documento.

2.2. Tecleando texto estructurado

Usualmente, los documentos largos tienen estructura: están organizados en capítulos, secciones y subsecciones, contienen diferentes tipos de texto tales como, texto normal, citas, notas al pie, teoremas, etc. Después de seleccionar un *estilo de documento* en Document → Style, TeXmacs se encarga de los detalles esquemáticos, tales como, numeración de secciones, páginas, teoremas, composición tipográfica de citas y notas al pie en forma adecuada.

Actualmente existen cuatro estilos estándar de documento: carta, artículo, libro y seminario. El estilo seminario se utiliza para hacer transparencias. Tan pronto como haya escogido uno de estos estilos podrá organizar su texto en secciones (vea Text → Section) y usar *entornos* específicos. Teorema, proposición, comentario, etc. son ejemplos de diversos entornos disponibles (vea Text → Environment). Otros ejemplos son listas de items (vea Text → Itemize) y listas numeradas (vea Text → Enumerate).

Cuando esté más familiarizado con TeXmacs, es posible adicionar sus propios entornos nuevos en su propio archivo de estilos. Imagine por ejemplo que suele hacer citas y quiere que éstas aparezcan en cursiva, con márgenes izquierdo y derecho de 1cm. En lugar de cambiar manualmente las propiedades del texto y del párrafo cada vez que hace una cita, es mejor crear un entorno de cita. No sólo será más rápido crear citas cuando se haga esto, sino que además es posible cambiar sistemáticamente el esquema de sus citas a lo largo del documento, sólo cambiando la definición del entorno de cita. Esta última situación ocurre por ejemplo si descubre *a posteriori* que prefiere que las citas aparezcan en una fuente más pequeña.

2.3. Etiquetas basadas en contenido

Los más simples ejemplos de estructura en un texto son las etiquetas basadas en contenido. En Text → content tags puede ver una lista de ellas. Las etiquetas basadas en contenido indican que una porción de un texto es de una clase particular o que sirve para un propósito especial. Por ejemplo, el texto importante debería ser marcado usanso la etiqueta strong (resaltado). Su visualización por defecto muestra un tipo de aspecto acentuado, como en este **texto resaltado**. Sin embargo, el texto resaltado puede ser mostrado en una forma distinta de acuero al estilo del documento. Por ejemplo, el texto resaltado puede ser mostrado en un color diferentes en las transparencias para presentaciones. A continuación sique una corta lista de las etiquetas basadas en contenido más comunes y su propósito:

Etiqueta	Ejemplo	Propósito
strong	esto es **importante**	Indica una región importante de texto
em	la cuestión *real*	Enfatiza una región de texto
dfn	Un *gnu* es una bestia cornuda	Definición de algún concepto
samp	La ligadura `ae`, æ	Una secuencia de caracteres literales
name	El sistema Linux	El nombre de una cosa particular
person	Soy Joris	El nombre de una persona
cite*	*Moby Dick* de Melville	Una cita bibliográfica
abbr	Trabajo en el C.N.R.S.	Una abreviación
acronym	El formato HTML	Un acrónimo
verbatim	El programa dijo `hello`	Texto literal como la salida de un programa
kbd	Por favor teclee `enter`	Texto que debería se ingresado en un teclado
code*	`cout << 1+1;` produce 2	Código de un programa de computadora
var	`cp` *`src-file dest-file`*	Variables en un programa de computadora

Table 2.1. Algunas de las etiquetas basadas en contenido más comunes.

2.4. Listas

Usando Text → Itemize usted puede empezar una lista no numerada. Puede también seleccionar una marca particular como • (bolos), − (guiones) o → (flechas) para indicar las entradas en la lista o la marca por defecto. Las listas pueden ser *anidadas* dentro de otras marcas como en las siguiente lista:

- Primer item.

- Ahora viena la sublista:
  - Un subitem.
  - Otro.
- Un item final.

La marca por defecto es mostrada en una forma diferente dependiendo del nivel de anidamiento. En el nivel más externo, usamos la marca •, en el segundo nivel ◦ y así. Cuando está dentro de una lista, note que presionando `return` automáticamente inicia un nuevo item. Si usted necesita items que son de varios párrafos de largo, entonces usted puede usar siempre `S-return` a fin de iniciar un nuevo párrafo.

Los entornos enumerados, que son iniciados usando Text → Enumerate, se comporta en una forma similar como una lista no numerada, excepto que los items son numerados. A continuación sigue un ejemplo de una enumeración que puede ser iniciada usando Text → Enumerate → Roman:

I. Un primer item.

II. Un segundo

III. Y un último

El último tipo de listas son las listas descriptivas. Ellas son iniciadas usando, Text → Description y permite describir una lista de conceptos:

Gnu. Una peluda pero gentil bestia.

Gnat. Sólo vive en un zoológico

2.5. Entornos

En forma similar a las etiquetas basadas en contenido, los entornos son usados para marcar porciones de un texto con un significado especial. Sin embargo mientras las etiquetas basadas en contenido usualmente encierran pequeñas porciones de texto, los entornos a menudo encierran porciones que son de varios párrafos de longitud. Los entornos frecuentemente usados en matemáticas son theorem y proof, como en el ejemplo de abajo:

Theorem 2.1. *No existen enteros positivos a, b, c y n con $n \geqslant 3$, tales que $a^n + b^n = c^n$.*

Proof. No tengo espacio suficiente aquí para escribir la prueba. □

Usted puede ingresar entornos usando Text → Environment. Otros entornos con una visualización similar a los teoremas con proposition, lemma, corollary, axiom, definition. Puede usar el macro dueto (ingresado usando `\ d u e t o return`) a fin de especificar la(s) persona(s) a la(s) cual(es) el teorema es debido, como en

Theorem 2.2. (Pitágoras) *Bajo adecuadas circunstancias, tenemos $a^2 + b^2 = c^2$.*

Otros entornos frecuentemente usados con una visualización similar a los teoremas, pero que no enfatizan el texto encerrado, son remark, note, example, warning, exercise y problem. Los entornos restantes verbatim, code, quote, quotation y verse pueden ser usados a fin de entrar texto o código multipárrafo, citas o poesía.

2.6. Aspectos de presentación

Por regla general, TeXmacs se ocupa de la presentación de su texto. Por lo tanto, aunque no queremos impedir esta posibilidad, no lo animamos a componer su documento visualmente. Por ejemplo, no debería insertar espacios o líneas en blanco como sustitutos del espacio vertical u horizontal entre palabras y líneas; en lugar de ello, el espacio adicional debería insertarse utilizando Insert → Space. Esto hará su texto más robusto en el sentido de que no tendrá que reconsiderar la presentación cuando apliques pequeños cambios que afecten a los saltos de línea o página, o grandes cambios, como los de estilo del documento.

Se han incorporado en TeXmacs diversos tipos de comandos explícitos de espaciado. Para empezar, puede insertar espacios rígidos de una anchura o altura dada. Los espacios horizontales no tienen altura y pueden ser elásticos o no. La longitud de un espacio elástico depende del modo de guionar el párrafo. Además se pueden insertar espacios tabulados. Los espacios verticales pueden insertarse o bien al comienzo o al final de un párrafo: el espacio vertical adicional entre dos párrafos es el máximo de entre el espacio vertical después del primero y el espacio vertical antes del segundo (en contraposición a TeX, lo que impide que aparezca un espacio superfluo entre dos teoremas consecutivos).

En lo referente a la presentación de los párrafos, puedes especificar el estilo de párrafo (justificado, alineado a la izquierda, centrado o alineado a la derecha), los márgenes del párrafo y el sangrado izquierdo (resp. derecho) de la primera (resp. última) línea de un párrafo. El usuario también controla los espacios entre párrafos y las líneas sucesivas en los párrafos.

Puede especificar la presentación de la página en el menú Document → Page. En primer lugar, puede especificar el modo de mostrar páginas en la pantalla: al seleccionar "papel" como tipo de página en Document → Page → Type, puedes ver de modo explícito los saltos de página. Por defecto el tipo de página es "papiro", lo que evita los saltos de página en la presentación del documento. El tipo de página "automático" asume que tu tamaño de papel es exactamente del tamaño de tu ventana. Los márgenes de página y la anchura del texto se especifican en Documento → Página → Presentación. Habitualmente es conveniente reducir los márgenes de página para la edición en pantalla; esto se puede hacer en Documento → Página → Presentación en pantalla.

2.7. El sistema de selección de fuentes

En TeXmacs, las fuentes tienen cinco características principales:

- Su nombre (roman, pandora, concrete, etc.).
- Su familia (roman, ancho fijo o sans serif).
- Su tamaño (un tamaño base (en puntos) y un tamaño relativo (normal, pequeño, etc.).

- Su serie (negrita, media o ligera).
- Su forma (recta, cursiva, versalita, etc.).

Note que en el sistema de selección de fuentes de LATEX 2ε el nombre y la familia de un tipo son sólo uno (llamado familia). Note también que el tamaño base de la fuente se especifica para el documento completo en Document → Font → Size.

2.8. Dominando el teclado

2.8.1. Reglas generales de prefijos

Como hay muchos atajos de teclado, es importante tener algunas formas de clasificarlas en varias categorías, a fin de hacer más fácil memorizarlas. Como regla general, los atajos de teclado que caen en la misma categoría son identificados por un prefijo común. Las principales de tales prefijos son:

`C-x`. Los atajos de teclado basados en la tecla Control son usados para comandos de edición frecuentemente usados. Ellos dependen mucho del "aspecto y comportamiento" en Edit → Preferences. Por ejemplo, si usted usa un aspecto y comportamiento comatible con Emacs, entonces los atajos de la forma `C-x` corresponden a los comandos Emacs, como `C-y` para pegar texto.

`A-x`. La tecla alternate key es usada para comandos que dependen del modo en el cual usted está. Por ejemplo, `A-s` produce texto **resaltado** en el modo texto y una raíz cuadrada en $\surd$ en el modo matemático. Note que `escape escape` es equivalente a `A-`.

`M-x`. La tecla meta es usada para comandos TeXmacs de propósito general, que pueden ser usados en todos los modos. Por ejemplo, `M-!` produces una etiqueta. Es también usado para comandos de edición adicional, como `A-w` para copiar texto si usted usa el aspecto y comportamiento Emacs. Note que `escape` es equivalente a `M-`.

`H-x`. El modificador de teclado de usuario es usado para producer símbolos especiales como los catacteres griegos en el modo matemático. Puede configurar su teclado para dejar que la tecla de mayúsculas juegue el rol de la tecla hiper. El `F5` is equivalent to `M-`.

Recordamos que los modificadores particulares que son usados a fin de obtener los prefijos `M-` y `H-` pueden ser configurados en Edit → Preferences.

2.8.2. Algunos atajos de teclado fundamentales

Algunas acciones de teclado estándar que son válidas en todos los modos son:

`S-return`. siempre inicia un nuevo párrafo

`C-backspace`. remueve un objeto o entorno.

`A-space`. inserta un espacio pequeño.

`A-S-space`. insertar un pequeño espacio negativo.

`A-A-home`. fija manualmente el comienzo de una selección.

`A-A-end`. fija manualmente el fin de una selección

`M-<`. ir al principio del documento.

`M->`. ir al final del documento.

2.8.3. Atajos de teclado para el modo texto

Las principales acciones de teclado adicionales disponibles en modo texto son `$` para entrar en el modo matemático y los siguientes atajos para crear caracteres acentuados:

Atajo		Ejemplo		Atajo		Ejemplo	
`A-'`	Acute ´	`A-' e`	é	``A-` ``	Grave `	``A-` e``	è
`A-^`	Sobrero ˆ	`A-^ e`	ê	`A-^`	Umlaut ¨	`A-" e`	ë
`A-~`	Tilde ˜	`A-~ a`	ã	`A-C`	Cedilla ¸	`A-C c`	ç
`A-U`	Breve ˘	`A-U g`	ğ	`A-V`	Check ˇ	`A-V s`	š
`A-O`	Anillo superior ˚	`A-O a`	å	`A-.`	Punto superior ˙	`A-. z`	ż
`A-H`	Hungarian ˝	`A-H o`	ő				

Table 2.2. Tecleando caracteres acentuados con TeXmacs.

Los caracteres especiales ß, SS,œ,Œ, æ y Æ son obtenidos, respectivamente, tecleando `S-F5 s`, `S-F5 S`, `S-F5 o e`, `S-F5 A E`, `S-F5 a` y `S-F5 A`.

En francés puedes usar también los atajos `< <` y `> >` a fin de obtener las *guillemets* francesas. En español, el signo de apertura de exclamación y el de apertura de interrogación son obtenidos, respectivamente, tecleando `! *` o ``! ` `` y `? *` o ``? ` ``.

2.8.4. Comandos híbridos y simulación LaTeX

TeXmacs le permite introducir comandos LaTeX directamente del teclado del modo siguiente. Primero presiones la tecla `\`-key a fin de entrar en el modo híbrido de comandos LaTeX/TeXmacs. Después teclee el comando que quiere ejecutar. Tan pronto como haya acabado de teclear su comando, el lado izquierdo del panel de información muestra algo como

```
<return>: acción que se realizará
```

Cuando pulse la tecla `enter` (return) su comando será ejecutado. Por ejemplo, en modo matemático puedes crear una fracción `\ f r a c enter`.

Si el comando que ha tecleado no es un comando LaTeX reconocido, entonces primero miramos si es un macro, función o entorno de TeXmacs (definido por el archivo de estilo). Si es así, la correspondiente expansión del macro, aplicación de la función o del entorno, es creada (con el número debido de argumentos). En otro caso, se supone que su comando corresponde a una variable de entorno y pedimos su valor. La tecla `\` es siempre equivalente a uno de los comandos `M-i l`, `M-i e`, `M-i a`, `M-i #` o `M-i v`.

2.8.5. Objetos dinámicos

Algunos objetos más complejos tienen varios estados durante el proceso de edición. Ejemplos de tales *objetos dinámicos* son los rótulos y las referencias, porque la aparición de la referencias depende de un número dinámicamente determinado. Muchos otros ejemplos de etiquetamiento dinámcio pueden ser encontrados en la documentación acerca de la escritura de archivos de estilo.

Cuando se ingresa un objeto dinámico como un rótulo usando `M-!`, el estado por defecto es *inactivo*. Este estado inactibo le permite teclar información que es relevandte al objeto dinámico, tal como el nombre del rótulo en nuestro caso. Ciertos objetos dinámicos toman un número arbitrario de parámetros y los nuevos pueden ser insertados usando `tab`.

Cuando termine de teclar la información relevante para su objeto dinámico, puede teclar `enter` a fin de *activar* el objeto. Un objeti dinámico activo puede ser desactivado colocando su cursor justo después del objeto y pulsando `backspace`.

2.8.6. Personalización del teclado

Es posible para el usuario modificar el comportamiento del teclado. A fin de hacer eso, sugerimos primero mirar los archivos en el directorio `$TEXMACS_PATH/progs/keyboard`, donde el comportamiento estándard del teclado es definido. Entonces usted puede redefinir el comportamiento en su archivo de inicialización privado.

Chapter 3

Fórmulas matemáticas

A fin de teclear fórmulas matemáticas, debe primer entrar en "modo matemático" presionando la tecla `$`- o mediante la inserción de una ecuación (usando Insert → Mathematics → Equation). En modo matemático, puede tener comandos específicos y combinaciones de teclas para teclaer símbolos matemáticos y fórmulas. Por ejemplo el prefijo `H-` puede ser usado a fin de ingresar símbolos griegos (recuerdo que `H-` es equivalente a `F5`, `escape escape escape` o `A-C-`).

El editor favorece teclear matemáticas con un cierto significado. Esta característica será desarrollada aún más en las versiones futuras, es útil cuando se comunica con un paquete de álgebra computacional. En este momento, usted debería por ejemplo explícitamente teclear la multiplicación `*` entre símbolos a y b. Por defecto, teclear `a b` producirá ab y no ab.

3.1. Principales constructos matemáticos

Los principales objetos matemáticos son creados usando el prefijo `A-` como sigue:

Atajo	Propósito	Ejemplo
`A-$`	Texto	$L=\{x\|x \text{ es suficientemente grande}\}$
`A-f`	Fracciones	$\frac{a}{b+c}$
`A-s`	Raices cuadradas	$\sqrt{x+y}$
`A-S`	Raices n-ésimas	$\sqrt[3]{x^3+y^3}$
`A-n`	Negaciones	$\cancel{\frac{a}{b+c}}$

Table 3.1. Creación del principal etiquetamiento matemático.

Las primas, subíndices and superíndices son creados como sigue:

Atajo	Propósito	Ejemplo
`'`	Primas	f' o $(g+h)'''$
`` ` ``	Primas anteriores	${}^{`}f$
`_`	Subíndices	x_n o x_{i_3}
`^`	Superíndices	x^2, x_n^2 o $\mathrm{e}^{\mathrm{e}^x}$
`A-l _`	Subíndices a izquierda	${}_2x$
`A-l ^`	Superíndices a izquierda	${}^{\pi}x$ or ${}^{*}_{*}\mathrm{He}^{*}_{*}$

Table 3.2. Creación de primas, subíndices y superíndices

3.2. Tecleando símbolos matemáticos

Los caracteres griegos son obtenidos en TeXmacs por la combinación de la tecla modificadora `H-` con una letra. Por ejemplo, `H-a` produce α y `H-G` produce Γ. Recuerde que la tecla `F5` es equivalente a `H-`, así que ρ puede también ser obtenido tecleando ρ. Similarmente, `F6`, `F7`, `F8` y `S-F6` pueden ser usados a fin de teclear caractéres resaltados, caligráficos, fraktur y blackboard. Por ejmplo `F8 m` produce $\mathfrak{m}$, `S-F6 R` produce $\mathbb{R}$ y `F6 F7 Z` produce $\boldsymbol{\mathcal{Z}}$.

Los caracteres griegos pueden también ser obtenidos como "variantes" de caracteres latinos usando la tecla `tab`. Por ejemplo, `p tab` produce π. La tecla `tab` es tambíen usada para obtener variantes de las letras griegas en sí mismas. Por ejemplo, tanto `H-p tab` como `p tab tab` producen ϖ.

Muchos otros símbolos matemáticos son obtenidos por combinaciones "naturales" de teclas. Por ejemplo `- >` produce $\rightarrow$, `- - >` produce $\longrightarrow$ y `> =` produce $\geqslant$. Similarmente, `| -` produce $\vdash$, `| - >` produce $\mapsto$ y `- > < -` produce $\rightleftarrows$. Algunas reglas generales se mantienen a fin de obtener variantes de símbolos:

- `tab`. es la tecla principal para obtener variantes. Por ejemplo, `> =` produce $\geqslant$, pero `> = tab` yields $\geq$. Similarmente, `< tab` produce $\prec$, `< tab =` produce $\preccurlyeq$ y `< tab = tab` produce $\preceq$. También, `P tab` produce $\wp$ y `e tab` produce la constante $\mathrm{e} = \exp(1)$. Usted puede "realizar el ciclo hacia atrés" using `S-tab`.

- `@`. es usada para poner símbolos dentro de circunferencias o cuadrados. Por ejemplo, `@ +` produce $\oplus$ y `@ x` yields $\otimes$. Análogamente, `@ tab +` produce $\boxplus$.

- `/`. es usado para las negaciones. Por ejemplo, `= /` produce $\neq$ and `< = /` produce $\nleqslant$. Note que `< = tab tab /` produce $\nleqq$, mientras `< = tab tab / tab` produce $\lneqq$.

- `!`. es usado después de las flechas a fin de forzar los superíndice a ser colocados arriba o abajo de la flecha. Por ejemplo, `- - > ^ x` produce $\longrightarrow^x$, pero `- - > ! ^ x` produce $\xrightarrow{x}$.

Varios otros símbolos que no pueden ser ingresados manualmente en la forma anterior son obtenidos usando el prefijo `S-F5`. A continuación se muestra una pequeña tabla de tales símbolos:

Atajo	Símbolo	Atajo	Símbolo
`S-F5 a`	$\amalg$		
`S-F5 n`	$\cap$	`S-F5 u`	$\cup$
`S-F5 v`	$\vee$	`S-F5 w`	$\wedge$

Table 3.3. Algunos símboloss que no pueden ser obtenidos usando reglas generales en una forma natural.

3.3. Tecleando operadores grandes

Las siguientes combinaciones de teclas son usadas a fin de crear símbolos grandes:

Atajo	Resultado	Atajo	Resultado
S-F5 I	$\int$	S-F5 O	$\oint$
S-F5 P	$\prod$	S-F5 A	$\coprod$
S-F5 S	$\sum$	S-F5 @ +	$\bigoplus$
S-F5 @ x	$\bigotimes$	S-F5 @ .	$\bigodot$
S-F5 U	$\bigcup$	S-F5 N	$\bigcap$
S-F5 V	$\bigvee$	S-F5 W	$\bigwedge$

Table 3.4. Operadores matemáticos grandes.

Los símbolos grandes de integral admiten dos variantes, dependiendo de donde quiere colocar los subíndices y superíndices. Por defecto, los índices son colocados como sigue:

$$\int_0^\infty \frac{\mathrm{d}x}{1+x^2}.$$

La visualización alternativa "con límites"

$$\int\limits_0^\infty \frac{\mathrm{d}x}{1+x^2}.$$

es obtenida usando S-F5 L I. Análogamente, puede teclear S-F5 L O a fin de obtener $\oint$ con límites.

3.4. Teclear delimitadores grandes

Los delimitadores grandes son creados como sigue:

Atajo	Resultado	Atajo	Resultado
A-(	$\big($	A-)	$\big)$
A-[	$\big[$	A-]	$\big]$
A-{	$\big\{$	A-}	$\big\}$
A-<	$\big\langle$	A->	$\big\rangle$
A-/	$\big/$	A-\	$\big\backslash$

Table 3.5. Atajos de teclado para los delimitadores grandes.

En TeXmacs los delimitadores grandes pueden ser "delimitadores izquierdos", "delimitadores derechos" o "delimitadorse centrales". Por defecto, $\big($, $[$, $\{$ y $\langle$ son delimitadores izquierdos, $)$, $]$, $\}$ y $\rangle$ son delimitadores derechos y $\big|$, $\big/$ y $\big\backslash$ son delimitadores centrales. Pero su naturaleza puede cambiarse usando las combinaciones de teclas A-l, A-r y A-m. Por ejemplo, A-l) produce $\big)$ considerado como delimitador izquierdo grande.

En TeX o LaTeX, los delimitadores centrales o “separadores” no existen; se usan para producir las barras verticales en fórmulas como

$$\left\langle \frac{a}{b+c} \middle| \frac{p}{q+r} \middle| \frac{a}{b+c} \right\rangle.$$

Puede haber tantos delimitadores centrales entre uno izquierdo y uno derecho como uno desee.

3.5. Acentos matemáticos anchos

La tabla inferior muestra como teclar acentos matemáticos arriba de los símbolos o fórmulas enteras. De hecho, algunos de estos acentos automáticamente se hacen tan largos como las fórmulas debajo de ellos.

Atajo	Ejemplo	Variante ancha	Atajo	Resultado
`A-~`	$\tilde{x}$	$\widetilde{x+y}$	`A-'`	$\acute{x}$
`A-^`	$\hat{x}$	$\widehat{x+y}$	`` A-` ``	$\grave{x}$
`A-B`	$\bar{x}$	$\overline{x+y}$	`A-.`	$\dot{x}$
`A-V`	$\vec{x}$	$\overrightarrow{\mathrm{AB}}$	`A-"`	$\ddot{x}$
`A-C`	$\check{x}$	$\widecheck{x+y}$		
`A-U`	$\breve{x}$	$\widebreve{x+y}$		

Table 3.6. Atajos de teclado para acentos matemáticos anchos.

Chapter 4

Material tabular

4.1. Creando tablas

A fin de crear una tabla usted puede usar o bien Insert → Table o uno de los siguientes atajos de teclado:

`M-t N t`. Crea una tabla normal.

`M-t N T`. Crea una tabla cuyas celdas están centradas.

`M-t N b`. Crea un "bloque" normal cuyas celdas entán separadas por líneas.

`M-t N B`. Crea un bloque cuyas celdas están centradas.

En modo matemático algunas estructuras tipo tabla son proveidas:

`M-t N m`. Crea una matriz.

`M-t N d`. Crea un determinante.

`M-t N c`. Crea una lista de elecciones.

El entorno `\eqnarray*` es también una clase especial de estructura tipo tabla, que se extiende una línea entere. Puede crear una lista de ecuaciones usando Insert → Mathematics → Equations.

Cuando creas una nueva tabla, su tamaño es mínimo (usualmente 1×1) y sus celdas están vacías. Puedes insertar nuevas filas y columnas usando los atajos de teclado `A-izquierda`, `A-derecha`, `A-arriba` y `A-abajo`. Por ejemplo, `A-derecha` crea una nueva columna a la derecha de la columna que ocupa actualmente el cursor. También puede crear una nueva fila debajo de la fila actual del cursor pulsando `enter`.

4.2. El modo para dar formato

En TeXmacs, bloques arbitrarios de celdas dentro de las tablas pueden ser formateados de diversas maneras. Por ejemplo, puede dar color individualmente a celdas individuales, pero también puedes seleccionar una columna al completo para que esté centrada horizontalmente. Por defecto los comandos para dar formato operan sobre celdas individuales, pero esto podemos cambiarlo a través de Table → Cell operation mode. Los siguientes modos de operación están disponibles:

`M-t m c`. Operar sobre celdas individuales.

`M-t m h`. Operar sobre filas.

`M-t m v`. Operar sobre columnas.

`M-t m t`. Operar sobre toda la tabla.

También es posible seleccionar un bloque de celdas usando el ratón y realizar una operación sobre ese sólo rectángulo.

4.3. Especificando el alineamiento de la celda y la tabla

La operación más frecuente de formateo es el alineamiento horizontal o vertical de un bloque de celdas. Puede usar los atajos de teclado `M-←`, `M-→`, `M-↑` y `M-↓` para alinear rápidamente más a la izquierda, derecha, arriba o abajo, respectivamente.

Un alineamiento más específico puede ser seleccionado en los menús Table → Horizontal cell alignment y Table → Vertical cell alignment. Alternativamente, usted puede usar los atajos de teclado `M-t h` x y `M-t v` x para alinear horizontal y verticalmente respectivamente.

De manera similar puedes especificar como debe ser el alineamiento de la misma tabla respecto al texto que la rodea. Esto se puede hacer, bien a través de los submenús Table → Horizontal table alignment y Table → Vertical table alignment, o bien usando los atajos de teclado `M-t H` x o `M-t V` x. aquí x representa `l` para "left" (izquierda), `c` para "centered" (centrado), `r` for "right" (derecha), `b` para "bottom" (abajo) y `t` for "top" (arriba).

4.4. Especificando el tamaño de la celda y la tabla

Usando Table → Cell width → Set width y Table → Cell height → Set height, respectivamente, puede especificar el ancho o el alto de una celda. De hecho hay tres modos de fijar el alto (o el ancho) especificados:

Modo mínimo. El ancho actual de la celda será el mínimo entre el ancho específicado y el ancho de la caja interior de la celda.

Modo exacto. El ancho de la celda será exactamente el especificado.

Modo Máximo. El ancho actual de la celda será el máximo entre el ancho especificado y el ancho de la caja interior de la celda.

El ancho del borde y del acolchado de la celda (que explicaremos más abajo) son tomados en cuenta para el tamaño de la caja interior de la celda.

También puede especificar la anchura y la altura de la tabla entera en Table → Special table properties. En particular puede especificar que la tabla ocupe el ancho completo de un párrafo. Cuando se especifica una anchura (o altura) para la tabla completa, necesita indicar como el espacio no usado será distribuido sobre las celdas usando Table → Special cell properties → Distribute unused space. Por defecto este espacio no usado se distribuye equitativamente.

4.5. Bordes, acolchado y color de fondo

Puede especificar el ancho de los bordes y de los espacios de acolchado[4.1] de una celda en las cuatro posibles direcciones: a la izquierda, derecha, arriba y abajo (ver Table → Cell border). Usted tiene los atajos de teclado `M-t b x` y `M-t p x` para especificar los anchos de borde y de acolchado de la celda.

El ancho del borde por defecto para las celdas en el entorno de bloque es de `1ln`, por ejemplo, el ancho de linea estandar en el tipo de letra actual (similar al ancho de una linea de fracción). Este ancho se añade a la derecha y debajo de cada celda (exceptuando las celdas que están en la primera fila o columna). El acolchado horizontal por defecto de cada celda es `1spc`: la anchura de un espacio en blanco en el actual tipo de de letra. El acolchado vertical por defecto de la celda es `1sep`: la mínima separación estandar entre dos cajas pegadas.

Las celdas se les puede dar un color de fondo vía Table → Cell background color.

A la tabla entera se le puede dar un borde y un acolchado en Table → Special table properties → Border. En este caso, el acolchado se realiza en el exterior del borde.

4.6. Características avanzadas de la tabla

En los menús puede encontrar algunas propiedades especiales para las tablas. Brevemente podemos comentar las siguientes:

- Cambiar el "alcance" de una celda y hacer que ocupe el espacio de sus vecinas a la derecha y debajo.
- Creación de subtablas completas dentro de una celda.
- Corrección de la profundidad y altura del texto para hacerlas coincidir las líneas base.
- Separación ortográfica horizontal de los contenidos de la celda y separación ortográfica vertical de toda la tabla.
- Pegado de varias filas y/o columnas de forma que las celdas pegadas se conviertan en "parte de los bordes" de las celdas restantes.
- Desactivación de la tabla para poder ver su "código fuente".
- Fijado del "centro de extensión" de una tabla. A partir de ello, la propiedades de formato de esa celda serán usadas de patrón para nuevas celdas creadas alrededor de este centro.
- Especificación del tamaño mínimo y máximo de una tabla, que será respetado en toda edición posterior (esto es principalmente útil en la creación de macros para tablas).

4.1. Los espacios de acolchado son aquellos que se encuentran entre el borde de la celda y el texto propiamente dicho en el interior de la misma.

Normalmente, todas las tablas provienen de un entorno que puede ser tabular, block, matrix, etc. Cuando cree sus propios macros para tablas, debe usar Table → Special table properties → Extract format para extraer el formato de una tabla dada.

Chapter 5

Links and automatically generated content

5.1. Creating labels, links and references

You may create a new inactive label using `M-!` or Insert → Link → Label and a reference to this label using `M-?` or Insert → Link → Reference. Be careful to put the label at a point where its number will be correct. When labeling sections, the recommended place is just after the section name. When labeling single equations (created using Insert → Mathematics → Equation), the recommended place is at the start inside the equation. When labeling multiple equations (created using Insert → Mathematics → Equations), you must put the labels just behind the equation numbers. Recall that you may use `A-*` in order to transform an unnumbered environment or equation into a numbered one, and vice versa.

It is possible to create hyperlinks to other documents using `M-i >` or Insert → Link → Hyperlink. The first field of the hyperlink is the associated text, which is displayed in blue when activated. The second field contains the name of a document, which may be on the web. As is usual for hyperlinks, a link of the form `#label` points to a label in the same document and a link of the form `url#label` points to a label in the document located at `url`.

In a similar fashion, an action may be associated to a piece of text or graphics using `M-i *` or Insert → Link → Action. The second field now contains a Guile/Scheme script command, which is executed whenever you double click on the text, after its activation. For security reasons, such scripts are not always accepted. By default, you are prompted for acceptation; this default behaviour may be changed in Options → Security. Notice that the Guile/Scheme command

```
(system "shell-command")
```

evaluates `shell-command` as a shell command.

Finally, you may directly include other documents inside a given document using `M-i i` or Insert → Link → Include. This allows you for instance to include the listing of a program in your text in such a way that your modifications in your program are automatically reflected in your text.

5.2. Inserting images

You can include images in the text using the menu Insert → Image. Currently, TeXmacs recognizes the `ps`, `eps`, `tif`, `pdf`, `pdm`, `gif`, `ppm`, `xpm` and `fig` file formats. Here, `gs` (i.e. ghostscript) is used to render postscript images. If ghostscript has not yet been installed on your system, you can download this package from

```
www.cs.wisc.edu/~ghost/index.html
```

Currently, the other file formats are converted into postscript files using the scripts `tiff2ps`, `pdf2ps`, `pnmtops`, `giftopnm`, `ppmtogif`, `xpmtoppm`. If these scripts are not available on your system, please contact your system administrator.

By default, images are displayed at their design size. The following operations are supported on images:

- Clipping the images following a rectangle. The lower left corner of the default image is taken as the origin for specifying a rectangle for clipping.

- Resizing an image. When specifying a new width, but no height at the prompt (or vice versa), the image is resized so as to preserve the aspect ration.

- Magnifying the image. An alternative way to resize an image, by multiplying the width and the height by a constant.

We also included a script to convert pictures, with optional LaTeX formulas in it, into encapsulated postscript. In order to include a LaTeX formula in an `xfig` picture, we recall you should enter the formula as text, while selecting a LaTeX font and setting the special flag in the text flags.

5.3. Generating a table of contents

It is very easy to generate a table of contents for your document. Just put your cursor at the place where you want your table of contents and click on Text → Automatic → Table of contents.

In order to generate the table of contents, you should be in a mode where page breaks are visible (select paper in Document → Page → Type), so that the appropriate references to page numbers can be computed. Next, use Document → Update → Table of contents or Document → Update → All to generate the table of contents. You may have to do this several times, until the document does not change anymore. Indeed, the page numbers may change as a result of modifications in the table of contents!

5.4. Compiling a bibliography

At the moment, TeXmacs uses `bibtex` to compile bibliographies. The mechanism to automatically compile a bibliography is the following:

- Write a `.bib` file with all your bibliographic references. This file should have the format of a standard bibliography file for LaTeX.

- Use Insert → Link → Citation and Insert → Link → Invisible citation to insert citations, which correspond to entries in your `.bib` file.

- At the place where your bibliography should be compiled, click on Text → Automatic → Bibliography. At the prompt, you should enter a `bibtex` style (such as `plain`, `alpha`, `abbrv`, etc.) and your `.bib` file.

- Use Document → Update → Bibliography in order to compile your bibliography.

Notice that additional BiBTeX styles should be put in the directory `~/.TeXmacs/system/bib`.

5.5. GENERATING AN INDEX

For the generation of an index, you first have to put index entries in your document using Insert → Link → Index entry. At a second stage, you must put your cursor at the place where you want your index to be generated and click on Text → Automatic → Index. The index is than generated in a similar way as the table of contents.

In the Insert → Link → Index entry menu, you find several types of index entries. The simplest are "main", "sub", "subsub", which are macros with one, two and three arguments respectively. Entries of the form "sub" and "subsub" may be used to subordinate index entries with respect to other ones.

A complex index entry takes four arguments. The first one is a key how the entry has to be sorted and it must be a "tuple" (created using `M-i <`) whose first component is the main category, the second a subcategory, etc. The second argument of a complex index entry is either blank or "strong", in which case the page number of your entry will appear in a bold typeface. The third argument is usually blank, but if you create two index entries with the same non-blank third argument, then this will create a "range" of page numbers. The fourth argument, which is again a tuple, is the entry itself.

It is also possible to create an index line without a page number using "interject" in Insert → Link → Index entry. The first argument of this macro is a key for how to sort the index line. The second argument contains the actual text. This construct may be useful for creating different sections "A", "B", etc. in your index.

5.6. COMPILING A GLOSSARY

Glossaries are compiled in a similar way as indexes, but the entries are not sorted. A "regular" glossary entry just contains some text and a page number will be generated for it. An "explained" glossary entry contains a second argument, which explains the notation. A "duplicate" entry may be used to create a page number for the second occurence of an entry. A glossary line creates an entry without a page number.

5.7. BOOKS AND MULTIFILE DOCUMENTS

When a document gets really large, you may want to subdivide it into smaller pieces. This both makes the individual pieces more easily reusable in other works and it improves the editor's responsiveness. An entire file can be inserted into another one using Insert → Link → Include. In order to speed up the treatment of included documents, they are being buffered. In order to update all included documents, you should use Tools → Update → Inclusions.

When writing a book, one usually puts the individual chapters in files `c1.tm`, `c2.tm` until `cn.tm`. One next creates one file `book.tm` for the whole book, in which the files `c1.tm`, `c2.tm` until `cn.tm` are included using the above mechanism. The table of contents, bibliography, etc. are usually put into `book.tm`.

In order to see cross references to other chapters when editing a particular chapter `ci.tm`, one may specify `book.tm` as a "master file" for the files `c1.tm` to `cn.tm` using Document → Master → Attach. Currently, the chapter numbers themselves are not dealt with by this mechanism, so you may want to manually assign the environment variable *chapter-nr* at the start of each chapter file in order to get the numbering right when editing.

Chapter 6
Herramientas de Edición

6.1. Cortar y pegar

Puede seleccionar texto y fórmulas manteniendo pulsado el botón izquierdo del ratón. Para borrar la región seleccionada, utiliza Edit → Cut. Para copiar la región seleccionada, primero haga click en Edit → Copy. Después, péguelo tantas veces como quiera en la ubicación del cursor, utilizando Edit → Paste. También puede copiar una región seleccionada utilizando el botón central del ratón.

También se puede cambiar las propiedades de la región seleccionada. Por ejemplo, para transformar un trozo de texto negro en rojo, lo seleccionas con el botón izquierdo del ratón y pinchas en Texto → Color → Rojo. Por un mecanismo similar, si seleccionas una fórmula y pinchas en Insertar → Matemáticas → Fracción, la fórmula se convertirá en el numerador de una fracción.

También se puede cambiar las propiedades de la región seleccionada. Por ejemplo, para transformar un trozo de texto negro en rojo, lo seleccionas con el botón izquierdo del ratón y hace click en Text → Color → Red. Por un mecanismo similar, si selecciona una fórmula y hace click en Insert → Mathematics → Fraction, la fórmula se convertirá en el numerador de alguna fracción.

Cuando se usa el mecanismo de copiado/pegado para comunicarse con otras aplicaciones, el texto es copiado y pegado usando el formato de datos de TeXmacs. Puede especificar otros formatos de importación/exportación usando respectivamente Edit → Import y Edit → Export. Por defecto copiar y pegar utiliza el *buffer* de texto primario. Usando Edit → Copy to y Edit → Paste from, puede especificar tantos otros *buffers* como quiera.

6.2. Buscar y sustituir

Puede iniciar una búsquesda de texto tecleando `C-s` o Edit → Search. Durante una búsqueda la "cadena de búsqueda" se muestra en el lado izquierdo de la parte inferior de la ventana. La búsqueda es incremental, lo que significa que cada carácter que introduce se añade a la cadena de búsqueda y su siguiente aparición queda rodeada por una caja de bordes rojos. Cada vez que teclea `C-s` el programa busca la siguiente aparición de la palabra. Si TeXmacs no encuentra más apariciones de la cadena en el documento, debería oir un bip. La búsqueda es también cíclica, en el sentido de que, si llegado ese momento, vuelve a pulsar `C-s` continuará desde comienzo del documento. Puede teclear `backspace` para deshacer la digitación de teclas durante una búsqueda.

La búsqueda descrita es hacia adelante, empezando en la posición actual del cursor. También puede buscar hacia atrás, utilizando `C-r`. Las búsquedas se producen sólo en texto que esté en el mismo modo e idioma que los activos en la posición de comienzo. En otras palabras, al buscar x en modo matemático no encontrarás ninguna x de texto ordinario. Una limitación actual es que la cadena de búsqueda puede contener sólo texto ordinario y no símbolos matemáticos o texto estructurado más complicado.

La búsqueda con sustitución se arranca tecleando `C-=` o Edit → Replace. $\mathrm{T_EX_{MACS}}$ le pide la cadena que quiere sustituir y la cadena que aparecerá en su lugar. Cada vez que se encuentre una de las cadenas a sustituir se le pedirá que elija entre sustituirla (y), no sustituirla (n) y sustituir todas las apariciones siguientes (a). Como antes, la búsqueda con sustitución afecta sólo al texto que esté en el mismo modo e idioma.

6.3. Revisión ortográfica

Si el programa `ispell` ha sido instalado en su sistema, entonces puede usarlo para hacer la revisión ortográfica de sus documentos, pulsando `M-$` o mediante Edit → Spell. Esto actuará sobre el texto completo o sobre la región seleccionada, si la tienes. Observe que debes verificar que el diccionario de español (o del idioma en que escriba) está instalado en su sistema. El inglés suele venir por defecto.

Cada vez que la revisión ortográfica se encuentre una parabra de ortografía dudosa aparecerán una serie de opciones en la parte inferior de la ventana

a). Aceptar la palabra incorrecta, así como todas sus apariciones futuras en el ámbito de la revisión.

r). Sustituir la palabra culpable por una corrección (que tienes que introducir).

i). Indicar que la palabra "incorrecta" es en realidad correcta, y que quieres que se inserte en tu diccionario personal.

1-9). Diversas sugerencias de correción.

Observe que `ispell` sólo busca palabras ortográficamente incorrectas. No hay corrección gramatical.

Tenga en cuenta que, dado que el corrector ortográfico arranca en el idioma activo en la posición del cursor (o al inicio de una selección), sólo se va a revisar el texto en ese idioma. Si el documento es multilingüe, tendrá que arrancar una vez el corrector ortográfico por cada idioma que use.

6.4. Deshacer y rehacer

Es posible deshacer gradualmente todos los cambios que haya hecho en el documento hasta llegar al momento de lanzar $\mathrm{T_EX_{MACS}}$, usando Edit → Undo o las combinaciones `M-[` o `C-/`. Los cambios deshechos pueden ser "rehechos" mediante Edit → Redo o `M-]`.

Para ahorrar memoria el número de acciones sucesivas que se pueden deshacer está limitado por defecto a 100. Es posible incrementar este número, añadiendo un comando como:

```
(set-maximal-undo-depth 1000)
```

en nuestro archivo personal de arranque (ver Help → Scheme). Cuando se especifica un número negativo $\mathrm{T_EX_{MACS}}$ permite deshacer *todas* las acciones de la sesión.

Chapter 7
Características de esquema avanzadas

7.1. Flujos

Los documentos complejos contienen con frecuencia notas al pie u objetos flotantes ("flotantes" en adelante), que aparecen separados del texto principal. De hecho, el contenido de tales documentos complejos utiliza varios *flujos*, uno para el texto principal, uno para las notas al pie, uno para los flotantes e incluso otro más para el texto a dos columnas. Los diferentes flujos se rompen en páginas de un modo bastante independiente.

A fin de insertar una nota al pie puede usar Insert → Page insertion → Footnote. el número de columnas puede ser cambiado en Paragraph → Number of columns.

7.2. Objetos flotantes

A los objetos flotantes se les permite desplazarse por la página independientemente del cuerpo del texto. Normalmente, los flotantes contienen figuras o tablas demasiado grandes para ajustar bien en el cuerpo del texto. Se insertan usando Insert → Page insertion → Floating object.

Puede también en un objeto flotante insertar directamente una figura o tabla, mediante Insert → Page insertion → Floating figure o, respectivamente, Insert → Page insertion → Floating table. Sin embargo a veces es posible que uno quiera ubicar varias figuras o tablas pequeñas dentro de un mismo objeto flotante. Puede hacer esto mediante Insert → Image → Small figure en el caso de las figuras y Insert → Table → Small table en el caso de las tablas.

Una vez creado un flotante, puedes controlar su posición usando, desde dentro del flotante, Insert → Position float. Puedes especificar si se debe permitir al objeto flotante aparecer al principio de la página, al final, directamente en el texto ("aquí") o en otra página. Por defecto a los flotantes les están permitidas todas estas opciones. Existe una excepción: un flotante nunca aparecerá en el cuerpo del texto a menos de tres líneas del principio o del final de una página.

7.3. Saltos de página

La división en páginas del documento se puede controlar de un modo muy preciso desde Document → Page → Breaking. En el submenú Algoritmo puedes especificar qué algoritmo se debe usar. La división de páginas profesional es la mejor para imprimir, pero puede hacer lenta la edición cuando se utiliza interactivamente en modo papel (recuerde Documento → Página → Tipo). En ese caso, en función de su máquina, puede querer usar el algoritmo laxo. El algoritmo medio es igual al profesional excepto cuando hay material en varias columnas. En ese caso particular, la división profesional es significativamente más lenta.

Al algoritmo de división de páginas se le puede permitir también alargar o reducir la longitud de las páginas en casos excepcionales en el submenú Limits. La elasticidad del espacio vertical entre párrafos, etc. puede ser especificada en Flexibility. El factor 1 aparece por defecto, si lo disminuyes el espaciado será más rígido, pero la calidad de los saltos de página puede verse afectada.

CHAPTER 8

USANDO GNU TEXMACS COMO UNA INTERFACE

Una característica importante de TEXMACS es su habilidad para comunicarse con sistemas externos en sesiones tipo consola de comandos. Típicamente, es posible evaluar los comandos de uns programa externo de álgebra computacional dentro de tales sesiones y mostrar los resultados en una agradable forma gráfica. Es posible también evaluar comandos de la consola y programas de SCHEME dentro de tales sesiones.

8.1. SESIONES

8.1.1. Uso básico

Una sesión puede ser iniciada desde el menú Insert → Session. Una sesión consiste de una secuencia de entornos de entrada y salida y de un posible texto entre ellos. Cuanto teclea `enter` dentro de un entorno de entrada, el texto dentro del entorno es evaluado y el resultado es mostrado como un entorno de salida.

Cuando se ingresa un comando en una sesión, la aplicación intenta ejecutarlo. Varios comandos pueden ser lanzados concurrentemente en el mismo documento, pero la salida sólo será activada en la sesión donde el cursor está y en la posición del cursor. Por tanto, recomendamos usar diferentes *buffers* para ejecuciones paralelas. Las ejecuciones pueden ser interrumpidas desde la barra de iconos. También es posible desconectar (cerrar) la aplicación; en este caso ningún comando posterior puede ser ejecutado en la sesión correspondiente.

En la segunda barra de iconos usted puede tener también unos pequeños botones para seleccionar entradas matemáticas e interrumpir la ejecución. Cuando está implementada para un sistema dado, la entrada matemática le permite teclear la entrada en una forma gráfica bidimensional. Los otros dos botones le permite interrumpir la ejecución de un comando particular (aunque estno no trabaja bien para ciertos sistemas) o desconectar el sistema externo. Cuando presiona enter en la entrada de un sistema no conectado, el sistema será reiniciado automáticamente.

8.1.2. Personalizando los estilos de las sesiones

Cada entorno de sesión toma dos argumentos: El lenguaje de programación y el nombre para una sesión. Todas entrada a evaluarse es redireccionada al paquete que eimplementa el lenguaje de programación y el nombre de la sesión es pasado como un argumento extra.

Es posible redefinir el esquema de pantalla de una sesión como sigue: Para cada sesión en un lenguaje de programación 'p', usted tiene entonces que modificar los entornos 'entrada-p' y 'salida-p' los cuales corresponden respectivamente al esquema de la entrada y la salida. El primer argumento para 'entrada-p' es el *prompt* para la entrada.

Es posible darle a las sesiones un nombre (siendo el nombre por defecto “default”): haciendo *click* en Insert → Session → Other, uno puede tanto insertar un nombre de sesión como un tipo de sesión. Sesiones diferentes que tienen el mismo tipo y el mismo nombre compartido corresponden a una instancia de la aplicación que está siendo ejecutada. Recíprocamente, tales seciones comparten un ambiente común. Mediante el uso de nombres diferentes de sesiones, uno puede lanzar concurrentemente varias intancias de la misma aplicación.

8.2. Sistemas soportados

Cuando de una mirada al menú Insert → Session, sólo aquellos sistemas que están actualmente instalados en su sistema serán mostrados. Las únicas excepciones son las sesiones de *shell* (consola) y de scheme, las cuales siempre están disponibles.

Abajo, entonrará una pequeña lista de los sistemas de álgebra computacional libres que disponen de una interface con TeXmacs. También existen interfaces con varios sistemas propietarios, pero debería mirar la documentación de aquellos sistemas para más información.

8.2.1. Sesiones de Shell y sesiones de Scheme

En una “sesión de *shell*” es posible evaluar comando de *shell*. Toda entra y salida es literal. Ninguna utilidad particular de línea de comandos (tales como mecanismos de completar) han sido implementados aún.

En una “sesión de Scheme” usted puede evaluar programas de Guile/Scheme. La entrada debe ser texto literal. La entrada es evaluada y el resultado es mostrado. Ningún mecanismo de salida gradual ha sido implementado aún para una sesión de Scheme.

8.2.2. Giac

Giac es un sistema de álgebra computacional, que puede ser descargado de

`http://www-fourier.ujf-grenoble.fr/~parisse/english.html`

8.2.3. GTybalt

GTybalt es un sistema de álgebra computacional libre que ha sido construido sobre GiNaC, CLN y un programa para interpretar comandos de C y C++. Para mayor información vea

`http://www.fis.unipr.it/~stefanw/gtybalt.html`

* Macaulay 2

Maxima

Maxima no es sólo uno de los más viejos y mejores sistemas de álgebra computacional de los alrededores, es también uno de los únicos sistemas de propósito general para los cuales hay una implementación libre. Puede obtenerlo de

`http://www.ma.utexas.edu/users/wfs/maxima.html`

La versión soportada es Maxima 5.6 basada en GCL. Para Maxima 5.6 basada en CLisp, edite su archivo `tm_maxima` y reemplace `-load` por `-i`. Para Maxima 5.9-pre, reemplace -load por `-p`. Problemas conocidos:

- Si presiona `enter` cuando una sentencia no está completa (típicamente, terminada por `;` o `$`), la interface se colgará.
- Si uste causa que aparezce el *prompt* de interrupción de Lisp, la interface se colgará.
- El comando `info` no está soportado (está denifnido en el Lisp subyacente, y es difícil para soportar la portabilidad).
- Algunos comandos en el trabajo del depurador, pero algunos (incluyendo `:c`) no funcionan, nadie sabe por qué.
- El comando `load` algunas veces se comporta misteriosamente,

8.2.4. Pari

Pari es un paquete de software para teoría de números asistida por computador. Consiste de una biblioteca de C, `libpary` (con núcleos opcionales de ensamblador para algunas arquitecturas populares), y de la calculadora interactiva programable `gp`. Puede descargar Pari desde

```
ftp://megrez.math.u-bordeaux.fr/pub/pari
```

Necesitará una versión más nueva que Pari-2.1.0 para su uso dentro de TeXmacs (para un sistema Pari ya instalado, teclee `gp --version`).

8.2.5. Qcl

QCL es un lenguaje de programación de alto nivel, independiente de la arquitectura, para computadores cuánticos, con una sintaxis derivada de los lenguajes procedimentales clásicos como C o Pascal. Este permite la completa implementación y simulación de algoritmos cuánticos (incluyendo los componentes clásicos) en un formalismo consistente. La interface con TeXmacs es útil principalmente para mostrar estados quánticos en una forma legible. Para mayor información, vea

```
http://tph.tuwien.ac.at/~oemer/qcl.html
```

Iniciando desde la versión 1.0.0.8, TeXmacs soporta QCL 0.4.3 o posteriores. Los usuarios de versiones más viejas deberán actualizarse.

8.2.6. Yacas

Yacas is, as it's name suggest, yet another computer algebra system. Things implemented include: arbitrary precision, rational numeric, vector, complex, and matrix computations (including inverses and determinants and solving matrix equations), derivatives, solving, Taylor series, numerical solving (Newtons method), and a lot more non-mathematical algorithms. The language natively supports variables and user-defined functions. There is basic support for univariate polynomials, integrating functions and tensor calculations. You can get Yacas at

```
http://www.xs4all.nl/~apinkus/yacas.html
```

Chapter 9
Archivos de estilo TeXmacs

9.1. Archivos de estilo TeXmacs

Uno de las fortalezas fundamentales de TeXmacs es la posiblidad de escribir sus propios archivos y paquetes de estilo. El propósito de los archivos de estilo es múltiple:

- Permiten la abstracción de los elementos repetitivos en los textos, como secciones, teoremas, enumeraciones, etc.
- Forman un mecanismo que permite estruturar su texto. Por ejemplo, puede indicar que una porción dada de su texto es una abreviación, una cita o "importante".
- Estilos de documento estándar le capacitan para escribir documentos de apariencia profesional, porque los archivos de estilo correspondientes han sido escrito con mucho cuidad por gente que sabe mucho acerca de tipografía y estética.

Para un documento es posible asociar uno o varios estilos de documentos, lo cuales son estandar o definidos por el usuario. El principal estilo de un documento es seleccionado en el menú Document → Style. Los estilos extra pueden ser adicionados usando Document → Use package.

Desde el punto de vista del editor, cada estilo corresponde a un archivo `.ts`. Los archivos correspondientes para cada estilo son procesados como si fueran documentos usuales, pero al final, el editor sólo mantiene el entorno final como el entorno inicila para el documento principal. Más precisamente, los archivos de estilo son procesados en orden, al igual que sus propios estilos, de una manera recursiva.

9.2. Los estilos y paquetes TeXmacs estándar

Actualmente los siguientes estilos estándar han sido implementados:

- Libro,
- Artículo;
- Carta;
- Seminario (para transparencias).

Cada uno de estos estilos exporta cierto número de funciones y estilos estándar (listados abajo). Se espera que todos los futuros estilos estándar de documento den soporte al menos a los comandos y entornos que siguen, y sugerimos a los usuarios escribir archivos de estilo que lo hagan:

- Comandos de división en secciones.

- Listas de bolos y numeradas.
- Entornos tipo ecuación.
- Entornos tipo teorema.
- Entornos de programación.

Note que los entornos tipo teorema no son estándar en LaTeX, lo cual constituye una fuente conocida de incompatibilidades. Los nuevos "teoremas" pueden ser adicionados con el comando `newtheorem`. Es también posible añadir nuevas "observaciones", con el comando `newremark`. Las "observaciones" difieren de los "teoremas" en que en general el cuerpo de las primeras no se compone con énfasis (usualmente cursiva).

Por supuesto, los entornos de programación no están soportados por LaTeX. Tales entornos están actualmente en desarrollo.

9.3. Diseñando sus propios archivos de estilo

En cualquier momento que los archivos de estilo de TeXmacs sean inadecuados para un propósito dado, es posible escribir sus propios archivos de estilo. Sin embargo, diseñar su propio archivo de estilo de la nada puede ser una tarea compleja. Por esta razón, recomendamos la reutilización o personalización de los archivos de estilo y paquetes estándar de TeXmacs cuando sea posible. Consecuentemente, puede ser prudente leer más acerca de la personalización de los archivos de estilo y paquetes de TeXmacs primero.

9.3.1. Mirar un ejemplo

Antes de escribir su propio archivo de estilo, puede ser útil echar un vistazo a algunos de los archivos de estilo estándar. Por ejemplo, cargue `book.ts` usando File → Load (no es necesario especificar una ruta de búsqueda, ya que el directorio de estilos está incluido en la ruta de archivos por defecto).

Después de cargar `book.ts` (por ejemplo), verá muchas declaraciones de funciones y entornos (estas declaraciones son visibles, puesto que los archivos de estilo están en "modo preámbulo" (ver Opciones → Modo). Algunas declaraciones más están contenidas en los archivos `basic.ts`, `list.ts`, `theorem.ts` y `program.ts`, sobre los cuales `book.ts` está basado. Estos ficheros contienen, respectivamente, entornos básicos, tipo lista, tipo teorema y de programación.

9.3.2. Atajos de teclado para escribir archivos de estilo

9.3.2.1. Macros, funciones y variables de entorno

Las principales combinaciones de teclas que usted debe saber para escribir archivos de estilo son las siguientes:

`M-=`. crea una nueva asignación. El primer argumento es el nombre de un nuevo comando y el segundo una expresión.

`M-w`. permite cambiar localmente una o más variables de entorno. Las sentencias `with` son de la forma $\langle x_1|a_1|\cdots|x_n|a_n|b\rangle$, donde los x_i son los nombres de las variables, los a_i son sus valores locales, y b es el texto sobre el que se aplica el entorno local.

`M-m`. crea un macro. Los argumentos se insertan utilizando la tecla `tab`.

`M-f`. crea una función. Los argumentos se insertan utilizando la tecla `tab`.

`M-i #`. obtiene el valor de un argumento del macro.

`M-i v`. obtiene el valor de una variable de entorno.

`M-i e`. expande el macro con cero o más argumentos.

`M-i a`. aplica una función a cero o más argumentos.

Más precisamente, cuando se evalua una expasión de macro $\{a|x_1|\cdots|x_n\}$ creada por `M-i e`, las siguientes acciones son llevadas a cabo:

- Si a no es una cadena ni un macro, entonces a se evalúa una sola vez. Esto produce o bien un nombre de macro o una expresión de macro f.
- Si obtenemos el nombre de un macro, entonces reemplazamos f por el valor de la variable de entorno f. Si después de esto f no es todavía un macro, entonces devolvemos f.
- Sean $y_1, \ldots, y_n$ los argumentos de f y b su cuerpo (se desechan los argumentos superfluos; los argumentos faltantes toman la cadena vacía como su valor por defecto). Entonces sustituímos x_i por cada y_i en b y devolvemos el resultado que ha sido evaluado.

Las funciones son similares a los macros, salvo en que los argumentos de las aplicaciones de una función son evaluados y no pueden ser editados directamente (primero necesitas desactivar la aplicación de la función, después editar los argumentos y por último, reactivar). También, $y_1, \ldots, y_n$ se consideran ahora como variables de entorno locales, a las que se atribuyen $x_1 \ldots x_n$ como sus valores. Estas variables locales no se recuerdan cuando una función devuelve una función que involucra esas variables.

9.3.2.2. Etiquetado computacional

Los siguientes comandos pueden ser usados para realizar cáculos dinámicos:

`M-e |`. "o" secuencial de dos condiciones.

`M-e ^`. "o" exclusivo de dos condiciones.

`M-e &`. "y" secuencial de dos condiciones

`M-e !`. negación de una condición.

`M-e +`. sumar dos números o longitudes.

`M-e -`. restar dos números o longitudes.

`M-e *`. multiplicar dos números.

`M-e /`. dividir dos números.

`M-e ;`. concatenar dos cadenas.

`M-e #`. mostrar un número en Árabigo, roman, Roman, alfa o Alfa (utilizado por ejemplo en las enumeraciones).

`M-e >`. traducir una palabra de un idioma fuente a un idioma destino (ver los diccionarios en `$TEXMACS_PATH/data/dic`).

`M-e =`. comprobación de igualdad.

`M-e E-e`. comprobación de desigualdad.

`M-e ?`. insertar una instrucción `if` con una parte `else` opcional.

9.3.3. Rutas importantes de TEXMACS

Esto debería estar en cualquier otro lugar.

Antes de escribir sus propio archivo de estilo, es util saber las siguientes rutas importantes de TEXMACS:

- `$TEXMACS_PATH` es la ruta principal para TEXMACS.
- `$TEXMACS_HOME_PATH` es la ruta principal de usuario para los archivos de TEXMACS (documentos, estilos o programas). Por defecto, esta ruta está configurada hacia `~/.TeXmacs`.
- `$TEXMACS_STYLE_ROOT` es la raíz de los directorios para los archivos de estilo. Por defecto, esta ruta contiene `$TEXMACS_PATH/styles` y `$TEXMACS_HOME_PATH/styles`.
- $TEXMACS_PACKAGE_ROOT es la raíz de los directorios para los paquetes de estilo. Por defecto, la ruta contiene `$TEXMACS_PATH/packages` y `$TEXMACS_HOME_PATH/packages`.
- `$TEXMACS_STYLE_PATH` contiene la ruta para incluir los archivos de estilo. Por defecto, esta ruta contiene `.` y todos los subdirectorios en `$TEXMACS_STYLE_ROOT` y `$TEXMACS_PACKAGE_ROOT`.
- `$TEXMACS_FILE_PATH` contiene la ruta para buscar archivos de texto. Por defecto, esta ruta contiene `$TEXMACS_STYLE_PATH`, `$TEXMACS_PATH/texts` y `$TEXMACS_HOME_PATH/texts`.

9.4. Personalizando los estilos y paquetes estándar de TEXMACS

En cualquier momento que los archivos de estilo de TEXMACS sean inadecuados para un propósito dado, es posible escribir sus propios archivos de estilo. Diseñar su propio archivo de estilo de la nada puede ser una tarea compleja. Por esta razón los archivos de estilo de TEXMACS han sido subdivididos en paquetes más pequeños a fin de facilidar la reutilización de ciertas partes. La política de diseño también le permite redefinir muchos macros *a posteriori*, lo cual le permite personalizar los archivos de estilo existentes en una forma fácil.

Chapter 10

Resumen de las principales etiquetas TeXmacs

10.1. La base común para la mayoría de los estilos

La `common-base` d.t.d. (por la sigla para *Document Type Definition*) contiene el etiquetado que es común a virtualmente todos los estilos. Está subdivido en las siguientes partes:

10.1.1. Etiquetas estándar

Varias etiquetas estándar están definidas en `std-markup`. Los siguientes etiquetas de contenido textual toman todas un argumento. La maryoría puede ser encontradas en el menú Text → Content tag.

strong. Indica una **importante** región de texto. Pueden entrar esta etiqueta vía Text → Content tag → Strong.

em. Enfatiza una región del texto como en "la cosa *real*". Esta etiqueta corresponde a la entrada del menú Text → Content tag → Emphasize.

dfn. Para definiciones como "un *gnu* es una bestia cornuda". Esta etiqueta corresponde a Text → Content tag → Definition.

samp. Una secuencia de caracteres literales como la ligadura ae (æ). Puede obtener esta etiqueta vía Text → Content tag → Sample.

name. El nombre de una cosa o concepto particular como el sistema Linux. Esta etiqueta es obtenida usando Text → Content tag → Name.

person. El nombre de una persona como Joris. Esta etiqueta corresponde a Text → Content tag → Person.

cite*. Una cita bibliográfica particuar como un libro o revista. Ejemplo: *Moby Dick* de Melville. Esta etiqueta, que es obtenida usando Text → Content tag → Cite, no debe ser confundida con cite. La última etiqueta es también usada para citas, pero cuando el artumento se refiere a una entrada en una base de datos con referencias bibliográficas.

abbr. Una abreviación. Ejemplo: trabajo en el C.N.R.S. Una abreviación es creada usando Text → Content tag → Abbreviation o el atajo de teclado `A-a`.

acronym. Un acrónimo es una abreviación formada por la priemra letra de cada palabra en un nombre o frase, tal como HTML o IBM. En particular, las letras no están separadas por puntos. Puede entrar un acrónimo usando Text → Content tag → Acronym.

verbatim. Texto literal como la salida de un programa de computador. Ejemplo: El programa dijo `hola`. Puede ingresar texto literal vía Text → Content tag → Verbatim. La etiqueta también puede ser usada como un entorno para texto multipárrafo.

kbd. Texto que debería ser ingresado en el teclado. Ejemplo: por favor presione `enter`. Esta etiqueta corresponde a la entrada de menú Text → Content tag → Keyboard.

code*. Código de un programa de computadora como en "`cout << 1+1;` produce 2". Esto es entrado usando Text → Content tag → Code. Para trozos más largos de código, debería usar el entorno code.

var. Variables en un programa de computador como en `cp` *`src-file dest-file`*. Esta etiqueta corresponde a la entrada del menú Text → Content tag → Variable.

math. Esta es una etiqueta que será usada en el futuro para texto matemático dentro del texto regular. Ejemplo: la fórmula $\sin^2 x + \cos^2 x = 1$ es bien-conocida.

op. Esta es una etiqueta que puede ser usada dentro del entorno matemático para especificar que un operador puede ser considerado en sí mismo, sin algún otro argumento. Ejemplo: la operación $+$ es una función de $\mathbb{R}^2$ a $\mathbb{R}$. Esta etiqueta puede entrar en desuso.

tt. Este es una etiqueta física para una frase con texto de máquina de escribir. Es usado por compabilidad con HTML, pero no recomendamos su uso.

Los siguientes son ambientes estándar:

verbatim. Descrito arriba.

code. Similar a code*, pero para piezas de código de varias líneas.

quote. Entorno para citaciones cortas (de un párrafo).

quotation. Entorno para citaciones largas. (multi-párrafo).

verse. Entorno para poesía.

center. Esta es una etiqueta física para centrar una o varias líneas de texto. Es usado para compatibilidad con HTML, pero no recomendamos su uso.

Algunos ambientes tabulares estándar son:

tabular*. Tablas centradas.

block. Tablas alineadas a la izquierda con un borde estándar de `1ln` de ancho.

block*. Tablas centradas con un borde estándar de `1ln` de ancho.

Las siguientes etiquetas misceláneas no tomas argumentos:

TeXmacs. El logo de $\text{T}_\text{E}\text{X}_{\text{MACS}}$.

TeX. El logo de $\text{T}_\text{E}\text{X}$.

LaTeX. El logo de LaTeX.

hflush. Usado por los desarrolladores para nivelar a la derecha en la definición de entornos.

hrule. Una regla horizontal como la que ve abajo:

Las siguientes etiquetas misceláneas toman todas uno o más argumentos:

overline. Para $\overline{\text{texto con una supralínea}}$, que puede ser cubierto a lo largo de varias líneas.

underline. Para $\underline{\text{texto subrayado}}$, que puede ser cubierto a lo largo de varias líneas..

fold. Macro con dos argumentos. El primer argumento es mostrado y el segundo es ignorado: El macro corresponde a la presentación plegada de una pieza de contenido asociada a un corto título o resumen. El segundo argumento puede ser hecho visible usando Insert → Switch → Unfold.

unfold. Macro con dos argumentos x y y, que produce la presentación desplegada de una pieza de contenidos y asociado a un pequeño título o resumen x. El segundo argumento puede ser hecho visible usando Insert → Switch → Fold.

switch. Macro con dos argumentos x y y, donde y es un conjunto de posibles representaciones del intercambio y x la representación actual. Las teclas de función F9, F10, F11 y F12 pueden ser usadas para intercambiar entre diferentes representaciones.

phantom. Función con un argumento x. Esta etiqueta toma tanto espacio como la composición tipográfica del argumento x tomaría, pero x no es mostrado. Por ejemplo, el texto "fantasma" como un argumento de phantom produce "   ".

set-header. Función con un argumento para cambiar permanentemente la cabecera. Note que ciertas etiquetas en el archivo de estilo, como las etiquetas de sección, pueden sobreescribir tales cambios manuales.

set-footer. Función con un argumento para cambiar permanentemente el pie.

10.1.2. Símbolos estándar

El d.t.d. `std-symbol` define los símbolos especiales ¢, ¤, ¥, ©, 🄯, ®, °, ², ³, ¹, µ, ¶, ¼, ½, ¾, € y ™. Tan pronto como el soporta para las fuentes sea mejorado en el futuro, este d.t.d. se volverá obsoleto.

10.1.3. Etiquetamiento matemático estándar

El etiquetamiento matemático estándar está definido en `std-math`.

binom. Para coeficientes binomiales $\binom{n}{m}$.

choose. Nombres alternative para binom (en desuso).

shrink-inline. A macro which switches to scriptsize text when you are not in display style. This macro is mainly used by developers. For instance, the binom macro uses on it.

Los siguients son entornos matemáticos tabulares estándar:

matrix. Para matrices $M = \begin{pmatrix} 1 & 2 \\ 3 & 4 \end{pmatrix}$.

det. Para determinantes $\Delta = \begin{vmatrix} 1 & 2 \\ 3 & 4 \end{vmatrix}$.

choice. Para listas de selección $|x| = \begin{cases} -x, & \text{if } x \leqslant 0 \\ x, & \text{if } x \geqslant 0 \end{cases}$

10.1.4. Standard lists

Las listas $\mathrm{T_EX_{MACS}}$ estandar están definidas en `std-list`. Las listas no numeradas son:

itemize. La etiqueta antes de cada item depende de la profunidad de anidamiento.

itemize-minus. Usa $-$ para la etiqueta.

itemize-dot. Usa $\bullet$ para la etiqueta.

itemize-arrow. Usa $\rightarrow$ para la etiqueta.

Las lista numeradas corresponden a los siguientes entornos:

enumerate. La clase de número antes de cada item depende de la profunidad de anidamiento.

enumerate-numeric. Numera los items por 1, 2, 3, etc.

enumerate-roman. Numera los items por i, ii, iii, etc.

enumerate-Roman. Numera los items por I, II, III, etc.

enumerate-alpha. Numera los items por a), b), c), etc.

enumerate-Alpha. Numera los items por A, B, C, etc.

Los siguients entornos puedes ser usadas para listas descriptivas.

description. El entorno por defecto para listas descriptivas (usualmente description-compact).

description-compact. Alinea los lados izquierdos de los items en las listas y pone sus descripciones prontamente después de estos.

description-dash. Similar a description-compact, pero usa un — para seperar cada item de su descripción.

description-align. Alinea los lados izquierdos de las descripciones, mientras alínea los items a la derecha.

description-long. Pone los items y sus descripciones en líneas distintas.

Los nuevos items en una lista son indicados a través de la etiqueta item o la etiqueta unaria item* en el caso de las descripcioones. Los desarrolladores también encontrarán unos pocos macros adicionales, pero inestables en `std-list` para definir estructuras de lista adicionales.

10.1.5. Generación automática de contenido

El d.t.d. `std-automatic` se especifica para la generación automática de contenido auxiliar como tables de contenidos y bibliografías, así como también para la presentación de tal contenido auxiliar. Las siguientes etiquetas son usadas para las bibliografías:

cite. Una función con un número arbitrario de argumentos. Cada argumento es una citación correspondients a un item en un archivo BiB-TEX. Las citaciones son mostradas en la misma forma como ellas son referenciadas en la bibliografía y también proveen hiperenlaces a las referencias correspondientes. Las citaciones son mostradas como macas de pregunta si usted no ha generado la bibliografía.

nocite*. Similar a cite, pero las citaciones no son mostradas en el texto principal.

bibitem*. Una función que especifica como mostrar un item en la bibliografía.

Las siguientes etiquetas son usadas para compilar tablas de contenidos:

toc-main-1. Una función con un argumento para crear entradas primordiales en la tabla de contenidos. Esta función puede por ejemplo ser usada cuando un libro consiste de varias partes.

toc-main-2. Una función con un argumento para crear una entrada principal en una tabla de contenidos. Esta función es usada regularmente para capítulos.

toc-normal-1. Una función con un argumento para crear una entrada normal en la tabla de contenidos. Esta función es frecuentemente usada para secciones.

toc-normal-2. Similar a toc-normal-2 para entradas menos importantes como en subsecciones.

toc-normal-3. Similar a toc-normal-3 para entradas incluso menos importantes como subsubsecciones.

toc-small-1. Usada para entradas no muy importantes tales como párrafos (puede ser ignorada).

toc-small-2. Usada para entradas incluso menos importantes tales como subpárrafos.

toc-dots. La separación entre una entrada en la tabla de contenidos y el correspondiente número de página. Por defecto, usamos puntos horizontales.

Las siguientes etiquetas son usadas para índices:

index. Una función con un argumento x, que inserta x en el índice como una entrada principal.

subindex. Una función con dos argumentos `x` y `y`, que inserta `y` en el índice como una subentrada de `x`.

subsubindex. Una función con tres argumentos `x`, `y` y `z`, que inserta `z` en el índice como una subentrada de `y`, que es a su vez una subentrada de `x`.

index-complex. Una función con cuatro argumentos `key`, `how`, `range`, `entry`, que está documentada en la sección acerca de generación del índice.

index-line. Esta función toma un argumento `key`, que dice como ordenar una entrada y el argumento actual `entry`. Ningún número de página es generado.

index-1. Macro con una entrada índice y un número de página, que es usada para visualizar una entrada principal del index en el índice.

index-1*. Similar a index-1, pero sin el número de página.

index-n. (con n entre 1 y 5): macro con una entrada en el índice y un número de página, que es usado para visualizar una entrada en el índice de nivel n.

index-n*. Similar a index-n, pero sin el número de página.

index-dots. El macro que produce los puntos entre una entrada al índice y el (los) correspondiente(s) número(s) de página.

Las siguientes etiquetas son usadas en los glosarios:

glossary. Una función que inserta su único argumento en el glosario.

glossary-dup. Para crear un número de página adicional para una entrada que había sido insertada antes.

glossary-explain. Una función para insertar una entrada en el glorario con su explicación.

glossary-line. Inserta una entrada de glosario sin número de página.

glossary-1. Macro for mostrar una entrada de glosario y su número de página correspondiente.

glossary-2. Macro para presentar una entrada de glosario, su explicación y su número de página.

glossary-dots. El macro que produce los putnos entre una entrada del glosario y el (los) correspondiente(s) número(s) de página.

10.1.6. Etiquetamiento especial para programas y sesiones

El `program` d.t.d. mainly provides the following environments for computer algebra sessions:

session. Macro con tres argumentoss: el lenguaje de algebra computacional, el nombre de la sesión y el cuerpo de la sesión en sí mismo.

input. Macro con dos argumentos: un *prompt* y la entrada input en sí misma.

output. Macro con el cuerpo de la salida como su argumento.

De hecho, estos entornos son basados en entornos de la forma *lan*-session, *lan*-input y *lan*-output para cada lenguaje individual `lan`.

El d.t.d. `program` también probee algún ediquetado para el esquema de los programas de comptuadora. Sin embargo, esas etiquetas deben ser considerada muy inestables, puesto que pensamos reemplazarlas por un conjunto de etiquetas más detalladas:

algorithm. Macro con dos argumentos: el nombre del algoritmo y el algoritmo en sí mismo, junto con su posible especificación.

body. El cuerpo real del algoritmo.

indent. Para identar una parte de un algoritmo.

10.2. Entornos estándar dentro del texto

El d.t.d `env` contiene los entornos estándar que están disponibles en la mayoría de los estilos. Está subdividio en las siguientes partes:

10.2.1. Definir nuevos entornos

El d.t.d `env-manage` contiene etiquetado de alto nivle el cual puede ser usado por el usuario para defninir nuevos entornos para teoremas, ejercicios y figuras:

new-theorem. Define un entorno tipo teorema. Usted debe especificar un nombre para el entorno (como "experimento") y el texto correspondiente (como "Experimento").

new-remark. Similar a new-theorem, pero para avisos.

new-exercise. Similar a new-theorem, pero para ejercicios.

new-figure. Similar a new-theorem, pero para figuras (en pares grandes y pequeños).

El d.t.d. también contienen etiquetado de bajo nivel para las definiciones actuales de los entornos. De hecho, la definición de nuevos teoremas es hecha en dos estados. En un primer estado, la etiqueta new-theorem es usada a fin de especificar que un entorno tipo teorema debería ser definido. En un segundo estado (justo antes de que el documento del usuario es procesado) los entornos tipo teorema son definidos. Este mecanismo hace posible personalizar los entornos en paquetes que son procesados entre los dos estados. Por ejemplo, la numeración de teoremas es personalizada de esta forma.

Warning 10.1. En el momento, usted debería sólo usar la etiqueta new-theorem y similares dentro de archivos o paquetes de estilo personales. Si usa new-theorem directamente dentro de un documento, entoces la numeración puede ser incorrecta, debudo a el esquema de dos estados explicados arriba. Esta inconveniencia desaparecerá tan prnto como sea posible especificar preámbulos límpios para documentos TeXmacs.

10.2.2. Entornos matemáticos

El d.t.d. `env-math` esperifica cuales entornos matemáticos pueden ser usados dentro del modo de texto. En otras palabras, los entornos deberían ser usados detro del mode de texto, pero sus cuerpos pueden contener fórmulas matemáticas o tablas de fórmulas matemáticas.

equation. Una ecuación numerada.

equation*. Una ecuación no numerada.

eqnarray. Un arreglo de ecuaciones numeradas. (no debe sería usarse aún).

eqnarray*. Un arreglo de ecuaciones no numeradas.

Dentro del entorno eqnarray*, usted puede usar la etiqueta eq-number a fin de numerar una ecuación.

WARNING 10.2. La numeración de ecuaciones dentro de tablas no es aún como debería ser. En particular, la etiqueta eqnarray es equivalente a eqnarray* en el momento. Después, cuando la etiqueta eqnarray sea implementada correctamente, usted también dispondrá de una etiqueta no-number a fin de suprimir el número de una ecuación y un paquete de estilo para numerar ecuaciones al lado izquierdo.

WARNING 10.3. No hay opción disponible para numerar ecuaciones a al lado izquierdo aún. Sin embargo, puede usar la etiqueta manual leq-number para esto. También tiene la etiqueta next-number el cual directamente muestra el siguiente número e incremente el contador de ecuaciones.

WARNING 10.4. No alentamos el uso de los entornos AMS-TEX, `align`, `gather` y `split`. No obstante, están disponibles bajo los nombres align, gather, eqsplit junto con sus variantes align*, gather* and eqsplit*. En el futuro, planeamos proveer entornos más poderosos.

10.2.3. Entornos tipo teorema

El d.t.d. `env-theorem` provee etiquetas para la capa de los entornos tipo teorema. Las etiquetas de mayor importancia son

render-theorem. Un macro para mostrar entornos tipo teorema. El primer argumento especifica el nombre del teorema, como "Teorema 1.2" y el segundo argumento contiene el cuerpo del teorema. Este entorno es usado para entornos definidos por new-theorem.

render-remark. Similar a render-theorem, pero para entornos tipo observación.

render-exercise. Similar a render-theorem, pero para entornos tipo ejercicio.

render-proof. Similar a render-theorem, pero para pruebas. Este entorno es principalmente usado para personalizar el nombre de una prueba, como en "Fin de la prueba del teorema 1.2".

dueto. Un entorno que puede ser usado para especificar los inventores de un teorema.

corollary*. Para corolarios no numerados. Este entorno es basado en render-theorem.

proof. Para pruebas de teoremas. Este entorno está basado en render-proof.

Las siguientes etiquetas pueden ser usadas para personalización posterior de los entornos.

theorem-name. Un macro que controla la apariencia de los nombres de los ambientes tipo teorema *y* tipo observación. La mayoría de los estilos usa fuente resaltada o pequeñas mayúsculas.

exercise-name. Similar a theorem-name, pero para ejercicios.

theorem-sep. El separador entre el nombre de un entorno tipo teorema o tipo observación y su cuerpo principal. Por defecto, este es un punto seguido por un espacio.

exercise-sep. Similar a theorem-sep, pero para ejercicios.

10.2.4. Entornos para objetos flotantes

El d.t.d. `env-float` provee etiquetas para los objetos flotantes. Las siguientes etiqueta es la única de altno nivel:

footnote. Hace una nota de pie de página

Las siguientes etiquetas de bajo nivel pueden ser usadas para definiciones de alto nivel figura y entornos tipo tabla big-figure, small-figure, big-table y small-table:

render-small-figure. Un macro para mostrar una pequeña figura. Los argumentos son nombres cortos (como "figura" o "tabla") para la lista de figuras, su nombre real (como "Figura 2.3" o "Tabla 5"), la figura en sí misma y un texto explicativo que acompaña a dicha figura o tabla.

render-big-figure. Una variante de render-small-figure para mostrar una gran figura.

Las siguientes etiquetas pueden ser usadas para personalizar la apareciencia del texto alrededor de las figuras, tablas y notas de pie de página:

figure-name. Un macro que controla la apariencia del texto "Figura". Por defecto, usamos fuente resaltada.

figure-sep. El separador entre la figura y su número y texto explicativo. Por defecto es un punto seguido por un espacio.

footnote-sep. El separador entre el número del pie de página y el texto. Por defecto es un punto seguido por un espacio.

10.2.5. Entornos por defecto

El d.t.d. `env-default` contiene los entornos textuales por defecto. Ellos están subdivididos en los siguientes grupos[10.1]:

Variantes de teoremas. Los cuerpos de los entornos tipo teorema son usualmente enfatizados. Por defecto, los siguientes de tales entornos están disponibles vía Text → Environment: theorem, proposition, lemma, corollary, axiom, definition, notation, conjecture.

10.1. Los nombres de las etiquetas se han mantenido en inglés para preservar la convención que se obtiene cuando se revisa el etiquetamiento de los documentos de TeXmacs.

Variantes de observaciones. Los siguientes está disponibles vía Text → Environment: remark, example, note, warning, convention.

Variantes de ejercicios. Dos de tales entornos son provistos por defecto y están disponibles vía Text → Environment: exercise y problem.

Variantes de figuras. Estos entornos siempre vienen por pares: grandes y pequeños. Por defecto, proveemos big-figure, small-figure, big-table and small-table. Puede acceder a ellos a través de Insert → Image e Insert → Table.

Otros entornos útiles. Proveemos keywords y AMS-class (para la clasificación de temas de A.M.S.). Estos entornos deberían ser ingresados en el interior de un abstract.

10.3. Cabeceras y pies de página

10.3.1. Cabeceras estándard

El d.t.d. `header` provee etiquetas para personalizar las cabeceras y pies de página. La personalización es basada en la idea de que podemos especificar un *texto de página* para cada página. Este texto de página puede por ejemplo ser el título corriente o el nombre de la sección actual. El texto de la página puede depender de la paridad de una página y aparecer en una forma diferentes para páginas especiales como comienzos de nuevos capítulos. Las siguientes etiquetas controlan la disposición física de los diferentes tipos de páginas:

start-page. Esta etiqueta, con el texto de la página como su único argumento, especifica la disposición de una primera página de un capítulo o sección.

odd-page-text. Similar a start-page, pero para la disposición de páginas impares ordinarias.

even-page-text. Similar a start-page, pero para la disposición de páginas pares ordinarias.

Las siguientes etiquetas controlan las acciones lógicas relacionadas con la cabecera a ser tomadas, cuando se especifica un título, un autor, o cuando se empieza una nueva sección.

header-title. Una etiqueta con el “argumento título” que es usado en la especificación del título del documento.

header-author. Una etiqueta con el “argumento autor” que es usado en la espeficicación del autor del documento.

header-primary. Una etiqueta con el “argumento nombre de sección” que es usado en el comienzo de cada nueva sección primaria. (i.e. chapter para el estilo libro, o section para el estilo artículo).

header-secondary. Una etiqueta con el “argumento nombre de sección” que es usado en el comienzo de cada nueva sección secundaria. (i.e. section para el estilo libro, o subsection para el estilo artículo).

10.3.2. Títulos estándar

El d.t.d. header-title provee etiqeutas para la información del título. Las siguientes etiquetas de alto nivel sólo pueden ser usados cuando son encapsuladas dentro de la etiqueta make-title:

title. Espeficica un título para el documento.

author. Especifica uno o varios autores para el documento.

address. Especifica la dirección del autor.

address-block. Especifica una dirección de un autor (en caso de múltiples direcciones).

title-email. Especifica la dirección de correo electrónico del autor.

title-date. Especifica la fecha de creación del artículo.

Los campos title y author usan las etiquetas header-title y header-author para especificar el título y la cabecera corrientes. Usted puede sobreescribir estos reusando header-title y header-author respectivamente. Las etiquetas anteriores también dependen de las siguientes etiquetas de bajo nivel para su disposición física:

title*. Macro con un argumento que especifica la disposición física de los títulos.

author*. Macro con un argumento que especifica la disposición física de los autores.

address*. Macro con un argumento que especifica la disposición física de las direcciones.

title-email*. Macro con un argumento que especifica la disposición física de las direcciones de correo electrónico.

title-date*. Macro con un argumento que especifica la disposición física de las fechas de creación.

El d.t.d. header-title también defines la etiqueta abstract para resúmenes de documentos.

10.4. Secciones estilo LATEX

El d.t.d. section-latex provee las etiquetas estándar para las secciones, que son las mismas en LATEX. Las etiquetas más seccionales toman sólo un argumento: el nombre de la sección. En el futuro, planeamos proveer etiquetas alternativas con dos argumentos, que permitirán ver el cuerpo de una sección como parte de la estructura. Las siguientes etiqetas usualmente permiten secciones numeradas, que son referenciadas en la tabla de contenidos:

chapter. Macro para producir un potencialmente numerado título de capítulo.

section. Macro para producir un potencialmente numerado título de sección.

subsection. Macro para producir un potencialmente numerado título de subsección.

subsubsection. Macro para producir un potencialmente numerado título de subsubsección.

paragraph. Macro para producir un potencialmente numerado título de párrafo.

subparagraph. Macro para producir un potencialmente numerado título de subpárrafo.

Las etiquetas chapter*, section*, subsection*, subsubsection*, paragraph* y subparagraph* pueden ser usados para producir variantes no numeradas de las etiquetas anteriores, las cuales no son referenciadas en la tabla de contenidos. El d.t.d. `section-latex` también provee las siguientes etiquetas:

chapter.** Macro con dos argumentos: un tipo especial de capítulo (como "Epílogo") y el nombre del capítulo.

appendix. Una variante de chapter o section para producir apéndices.

section-sep. Un macro para personalizar el separador entre el número de una sección y su título. Por defecto, usamos dos espacios.

10.5. ETIQUETAMIENTO PARA LA NUMERACIÓN AUTOMÁTICA

10.5.1. Numeración de entornos

El d.t.d. `number-env` provee etiquetas de bajo nivel para la numeración de entornos estándar. Una de las más importantes etiquetas es init-stdenv que es usada para reconfigurar todos los contadores del entorno. Esto es usualmente hecho al comienzo de cada capítulo o sección, o una vez para el documento entero.

El d.t.d. también exporta las etiquetas de muy bajo nivel newliststdenv, newlistfigure and newliststdenv-counter, que controlan la numeración en colaboración con `env-manage`. Los paquetes `number-us` y `number-europe` son provisto para numeración de estilo americano y estilo europeo.

10.5.2. Numeración de secciones

El d.t.d. `number-section` provee etiquetas de bajo nivel para numerar secciones. Define las etiquetas reset-chapter, reset-section, reset-subsection y reset-subsubsection, para reconfigurar los contadores apropiados en cada nuevo capítulo, sección, subsección o subsubsección. También define una etiqueta reset-top para reconfigurar tolos los contadores de alto nivel.

El d.t.d. también define las etiquetas the-chapter, the-section, the-subsection y the-subsubsection para imprimir el nombre de capítulo, sección, subsección o subsubsección actual. Finalmente provee la etiqueta the-prefix que produce el prefijo cuando se numeran entornos (ecuaciones, teoremas, tablas, etc.).

Chapter 11

Personalizando TeXmacs

Una de las principales características de TeXmacs es que puede ser altamente personalizado. Primero que todo, los más importantes aspectos del programa pueden ser confgiurados en Edit → Preferences. La mayoría de las otras partes de TeXmacs pueden ser enteramente adaptadas o reprogramadas usando el lenguaje de extensión Guile/Scheme. En lo siguiente daremos una corta panorámica de como funciona esto en los casos más simples.

Introducción al lenguaje de extensión Guile

Como Emacs, TeXmacs viene con un lenguaje de extensión tipo Lisp, llamado el dialecto Guile Scheme del proyecto Gnome. Para documentación acerca de Guile Scheme, nos referimos a

```
http://www.gnu.org/software/guile/guile.html
```

Scheme tiene la ventaja de que puede ser extendido con tipos y rutinas externas C y C++. En nuestro caso, hemos extendido Scheme con rutinas que puede usar para crear sus propios menús y combinaciones de teclas, e incluso escribir sus propias extensiones a TeXmacs.

Si usted ha descargado los archivos fuentes de TeXmacs, entonces puede ser interesante para usted echar un vistazo a los archivos

```
Guile/Glue/build-glue-basic.scm
Guile/Glue/build-glue-editor.scm
Guile/Glue/build-glue-server.scm
```

Estos tres archivos "pegamento" contiene las rutinas C++, que son visibles dentro de Scheme. EN lo que sigue, discutiremos algunas de las rutinas más importantes. Planeamos escribir una guía de referencia más completa después. Usted puede también echar un vistazo a los archivos Scheme `.scm` en el directorio `$TEXMACS_PATH/progs`.

11.1. Escribir sus propios archivos de inicialización

Cuando inicia, TeXmacs ejecuta el archivo `$TEXMACS_PATH/progs/init-texmacs.scm`, a menos que usted haya especificado su propio archivo de inicialización en `$TEXMACS_HOME_PATH/progs/my-init-texmacs.scm`. Pro defecto, la ruta `$TEXMACS_HOME_PATH` iguala a `.TeXmacs`. Usualmente, usted quiere añadir algunas acciones extra al archivo de inicialización por defecto. En este caso, no debería olvidar incluir el comando

```
(exec-file "$TEXMACS_PATH/progs" "init-texmacs.scm")
```

en su archivo de inicialización personal. Similarmente, el archivo `$TEXMACS_PATH/progs/init-buffer.scm` es ejecutado, cada vez que usted crea un nuevo *buffer*, a menos que usted provea su propio archivo de inicialización `$TEXMACS_HOME_PATH/progs/my-init-buffer.scm`

11.2. Crear sus propios menús dinámicos

En particular, el archivo de inicialización por defecto ejecuta

```
$TEXMACS_PATH/progs/menu/main_menu.scm
```

a fin de configurar el menú principal de $\TeX_{\text{MACS}}$. Le sugerimos echar un vistazo a este archivo a fin de ver cómo son creados los menús.

De hecho, cualquier menú o parte de un menú es representada por un programa. El programa consiste de una lista de programas de una de las siguientes formas:

```
(=> "pulldown menu name" menu-definition)
(-> "pullright menu name" menu-definition)
("entrada" accion)
("entrada" "shorthand" accion)
---
(if condition menu-definition)
(link variable)
```

Los constructores `=>` y `->` son usados para crear un menú desplegable hacia abajo o un menú desplegable hacia la derecha y la condición `menu-definition` debe contener un programa que crea un submenú. El constructor `("entrada" accion)` crea una entrada ordinaria, donde la `accion` será compilado y ejecutado cuando haga click en la `entrada`. El código opcional `"shorthand"` se establece para un macro de teclado con la misma acción. Los items de un menú pueden separarse usando `---`. El constructor `if` es usado para insertar items de menú sólo si una cierta condición es satisfecha (por ejemplo, si estamos en el modo matemático).

Finalmente, si hemos declarado una variable $\TeX_{\text{MACS}}$ ser un menú mediante

```
(define variable menu-definition)
```

entonces podemos usar este menú indirectamente usando el constructor `link`. esta forma indirecta de declarar submenús tiene dos ventajas

- Un menú "indirecto" puede ser vinculado a tantos menús como nos guste.
- Nuevos items pueden ser adicionados a los submenús "indirectos" *a posteriori* usando

  ```
  (set! variable (menu-merge variable menu-declaration))
  ```

  De hecho, los términos existentes pueden ser también sobrescritos en esta forma.

Algunos menús indirectos estándar en $\TeX_{\text{MACS}}$ son `texmacs-menu`, `file-menu`, `edit-menu`, `insert-menu`, `text-menu`, `paragraph-menu`, `document-menu`, `options-menu` y `help-menu`. El comando

```
(menu-main menu-declaration)
```

es usado a fin de configurar de hecho el menú principal. Por ejemplo, en la inicialización, ejecutamos

```
(menu-main '(link texmacs-menu))
```

11.3. Crear sus propios atajos de teclado

Los mapas de teclas son especificados usando el comando

```
(set-keymap (lista de predicados) (list of keymaps))
```

La lista de los predicados espeficica bajo cuales circunstancias los mapas de teclas son validas. Ejemplso de predicados son `always?`, `in-math?` y `in-french?`, pero el usuario puede definir sus propios predicados. Cada mapa es de una de las seguiente formas

```
(combinacion-de-teclas accion_1 ... accion_n)
(combinacion-de-teclas resultado)
(combinacion-de-teclas resultado mensage-ayuda)
```

En el primer caso, las `accion_i` son comandos de Scheme asociados con la cadena `combinacion-de-teclas`. En el segundo y tercer caso, `resultado` es una cadena que está para ser insertada en el texto cuando la `combinacion-de-teclas` ha sido completada. Un `mensage-ayuda` opcional puede ser mostrado cuando la `combinacion-de-teclas` es finalizada.

Los mapas de teclas pueden ser limpiados de nuevo usando el comando

```
(remove-keymap (lista de predicados) (lista de combinaciones de
teclas))
```

donde el segundo argumento es una lista de cadenas, cada uno de las cuales es una `combinacion-de-teclas` como las anteriores.

11.4. Otros archivos interesantes

Algunos otros archivos puede ser valiosos de mirar en:

- `$TEXMACS_PATH/fonts/enc` contiene codificaciones para diferentes fuentes TeX.
- `$TEXMACS_PATH/fonts/virtual` contiene definiciones de caracteres virtuales.
- `$TEXMACS_PATH/langs/natural/dic` contiene los diccionarios actuales usados por TeXmacs.
- `$TEXMACS_PATH/langs/natural/hyphen` contiene los patrones de separación silábica para varios lenguajes.
- `$TEXMACS_PATH/progs/fonts` contiene programas Scheme para configurar las fuentes.

Compatibilidad con otros formatos

TeXmacs es completamente compatible con postscript, que es usado como el formato a fin de imprimir documetnos. TeXmacs también provee conversores desde y hacia LaTeX y un filtro de entrada para Html.

11.5. Compatibility with LaTeX

Although TeXmacs has not been designed to be fully compatible with LaTeX, it is possible to convert documents from TeXmacs to LaTeX and *vice versa*, although the result will not always be perfect. Also, conversions from TeXmacs to LaTeX will generally yield better results than conversions the other way around. In particular, TeXmacs may reasonably well be used to write articles, which need to be converted to LaTeX for submission purposes. In this chapter, we will describe more precisely the conversion mechanisms, which will help you to obtain a result as satisfactory as possible.

11.5.1. Conversion from TeXmacs to LaTeX

The most common situation is that you want to convert an article from TeXmacs to LaTeX, in order to submit it to some journal. Given a TeXmacs file `name.tm`, you may convert it into a LaTeX file `name.tex` using File → Export → Latex. At a first stage, you may try to run LaTeX on `name.tex`, and see whether you obtain a satisfactory result. If so, then you should submit `name.tex` together with the style file `TeXmacs.sty`, which can be found in the directory `$TEXMACS_PATH/misc/latex`.

Often, the journal to which you submit uses its own style file, say `journal.sty`. In that case, you should also copy the file

```
$TEXMACS_PATH/styles/article.ts
```

to

```
~/.TeXmacs/styles/journal.ts
```

and use `journal` as your document style in Document → Style → Other. You may optionally edit `journal.ts`, so that the article layout becomes closer to the journal's style. In some cases, you also have to create a new copy of `TeXmacs.sty`, and modify some of the environments for compatibility with the journal's style file `journal.sty`.

If your first try to convert your document into LaTeX did not yield a satisfactory result, then you will usually observe that only minor parts of the texts were not converted correctly. This may be due to three main reasons:

- Your text uses some specific TeXmacs features.
- You used a TeXmacs feature, which has not yet been implemented in the conversion algorithm.
- You found a bug in the conversion algorithm.

These issues will be discussed in more detail in the next section.

In case of problems, a naive strategy would be to correct the produced LaTeX file and to send it to the journal. However, this strategy has the disadvantage that you have to make these corrections over and over again, each time that you convert your TeXmacs file `name.tm`, after having made some extra modifications. A better strategy is to use the Insert → Specific → Latex and Insert → Specific → Texmacs constructs to write text which is visible in the converted resp. original file only.

For instance, assume that the word "blauwbilgorgel" is hyphenated correctly in the TeXmacs source, but not in the LaTeX conversion. Then you may proceed as follows:

1. Select "blauwbilgorgel".

2. Click on Insert → Specific → Texmacs to make the text "blauwbilgorgel" TEXMACS-specific.

3. Click on Insert → Specific → Latex.

4. Type the latex code `blauw\-bil\-gor\-gel` with the correct hyphenation.

5. Press `return` to activate the LATEX-specific text.

In a similar fashion, you may insert LATEX-specific line breaks, page breaks, vertical space, style parameter modifications, etc.

11.5.2. Possible conversion problems

11.5.2.1. Specific TEXMACS features

Some TEXMACS typesetting primitives have no analogues in LATEX, and the conversion algorithm will simply transform them into blank space. Some main features which are specific to TEXMACS are the following:

- Left primes.
- Big separators between big parentheses.
- Mosaics.
- Trees.
- Complex user macros.
- Vertical spaces "before" and "after".
- Indentation flags "before" and "after".

You should avoid to use these specific TEXMACS features, if your document needs to be converted into LATEX. Nevertheless, in the far future, the conversion program might generate encapsulated postscript by default of a more intelligible translation.

11.5.2.2. Not yet implemented conversions

Although we try to keep the conversion algorithm as complete as possible for your needs, certain things have not yet been implemented. Some examples of not yet implemented issues are

- Non standard fonts.
- Conversion of tabulars.
- Style parameters.

Any suggestion about desirable extensions of the conversion algorithm should be reported to

`contact@texmacs.org`

and we will try to incorporate it as quickly as possible. It may take some time to implement the correct conversion of style parameters, since these are not the same in TEXMACS and LATEX. Furthermore, layout differences between TEXMACS and LATEX can not entirely be eliminated.

11.5.2.3. Bugs in the conversion algorithm

The most annoying situation if when a converted TEXMACS document produces lots of errors at the compilation or if the result has nothing to do with the original. In that case you have probably detected a bug in the conversion algorithm (or in the installation of LATEX on your system). Please try to figure out the source of the bug in this case and report it by sending an email to

`TeXmacs@math.u-psud.fr`

11.5.2.4. Work-arounds

TEXMACS has not been designed to be fully compatible with LATEX. As to the conversion from LATEX to TEXMACS, our main aim is to *help* you in converting old documents to TEXMACS. As long as you did not define weird environments and as long as you did not use weird style files and commands, you should be able to convert your old documents reasonably well. Otherwise, we suggest to modify your old document in a way that is does convert reasonably well and to apply some final changes in the result.

11.5.3. Conversion from LATEX to TEXMACS

The current aim of the conversion program from LATEX to TEXMACS, is to *help* you in translating old documents into TEXMACS. *Grosso modo*, conversions from LATEX to TEXMACS are more problematic than conversions the other way around. Nevertheless, as long as you restrict yourself to using the most common LATEX commands, you should be able to convert your old documents reasonably well. For example, all TEXMACS help files have been written in LATEX in order to validate the LATEX to TEXMACS conversion program.

You may convert a LATEX document `name.tex` into TEXMACS using File → Import → Latex and save it under `name.tm`. If your LATEX document was written sufficiently well, then the converted result should be more or less acceptable, apart from certain unrecognized commands, which appear in red. A good solution would be to write your own style file for converted documents, based on the original style, and in which the unrecognized commands are defined.

However, in certain less fortunate cases, the converted document will look like a great mess. This usually stems from the fact that TEX and LATEX allow users to modify the parser dynamically, for instance using the `\catcode` command. In this case, the conversion program may get confused, by making erroneous assumptions on the mode or the environment. As a result, text may be converted as mathematics, mathematics as verbatim, and so on. Nevertheless, the commands in your source file `name.tex` which confused the conversion program are usually easily localized by comparing the LATEX source with its TEXMACS conversion. Modulo some hacking of the source, you should be able to remove the litigious code, so that the document converts reasonably well.

In the future, we also plan to extend the conversion program with a style file converter and some additional features which facilitate the translation of user defined commands, which are defined in another document than the one you want to convert.

11.6. Conversión de documentos TeXmacs a Html

Hemos empezado a implementar la convsrsión entre HTML y TeXmacs. En este momento, sólamente es posible importar documentos HTML utilizando File → Import → Html. La mayoría de HTML 2.0 y parte de HTML 3.0 está actualmente soportado. Sin embargo, todavía no se han añadido capacidades navigacionales. En el futuro implementaremos Math-ML.

Cuando se importan documentos HTML los ficheros cuyos nombres comiencen con `http:` o `ftp:` se descargan de la red utilizando `wget`. Si compiló TeXmacs usted mismo, entonces puedes descargar `wget` de

`ftp://ftp.gnu.org/pub/gnu/wget/`

En las distribuciones binarias `wget` está incluído.

Appendix A

Configuración de TeXmacs

A.1. Introducción

Antes de que inicie a usar TeXmacs, puede ser prudente configurar el programa primero en Edit → Preferences, así se acomodará mejor a sus necesidades. Más importante, puede escoger el "aspecto y comportamiento" en Edit → Preferences → Look and feel. Esto le permitirá por ejemplo establecer los atajos de teclado usados por TeXmacs sean similares a los que usted usa en otras aplicaciones.

También, TeXmacs viene con un poderoso sistema de atajos de teclado, el cual intenta optimizar el uso de modifcadores como `shift` y `control` en su teclado. Sin embargo, en muchos sistemas X Window esos modificadores no están bien configurados, así que puede desear rehacer esto por usted mismo.

A.2. Configuración de las teclar modificadoras

TeXmacs usa cinco modificadores de teclado principales: `shift`, `control`, `alternate`, `meta` e `hyper`, que son abreviadas como `S-`, `C-`, `A-`, `M-` and `H-`. Las teclas `shift` y `control` están presentes en virtualmente todos los teclados y la tecla `alternate` en la mayoría. Muchos fabricantes de PC's estos días tienen también una tecla `windows`, la cual es usualmente equivalente a la tecla `meta` para TeXmacs.

Antes de reconfigurar su teclado, debería primero chequear que esto es de hecho necesario. Si tiene teclas que corresponden a `shift`, `control`, `alternate` y `meta` en una forma apropidad, entonces problamente no requiera hacer algo. Una posible excepción es cuando usted quiera sar una tecla simple como `caps-lock` para teclear símbolos matemáticos. En ese caso, debería hacer corresponder `caps-lock` a `hyper`.

A fin de reconfigurar el teclado, simplemente selccione el moficador lógico que quiere corresponder a una tecla física dada en Edit → Preferences → Keyboard. Por ejemplo, seleccionado Windows key → Map to M modifier, la tecla `windows` corresponderá al modificador `meta`. Similarmente, cuanco seleccione Caps-lock key → Map to H modifier, la tecla `caps-lock` corresponderá al modificador `hyper`.

Infortunadamente, X Window sólo permite una reconfiguración a lo ancho del sistema. Consecuentemente, si usted reconfigura la tecla `caps-lock` dentro de TeXmacs, entonces el nuevo comportamiento de `caps-lock` afectará todas las otras aplicaciones también. Es por esto importante reconfigurar sólo aquellas teclas que usted no usa para algo más en otras aplicaciones. Por ejemplo, la tecla `windows` no es usada por muchas aplicaciones, así que generamente no hace ningún daño reconfigurarla. Puede también preferir realizar alguna configuración a lo ancho del sistema. Esto puede ser hecho con el comando `xmodmap`; vea la página correspondiente del manual para mayor información.

En ciertos casos, usted ya tiene teclas en su teclado que corresponde a `alter`, `meta` e `hyper`, pero no en la forma en que usted quiere. Esto puede ser hecho reasinando los prefijos `A-`, `M-` y `H-` a otros modificadores lógicos en el primer grupo de submenus de Edit → Preferences → Keyboard.

Por ejemplo, para compatibilidad con EMACS, podría querer permutar la tecla `meta` o `windows` con `alter` sin hacer ningún cambio a lo amplio de sistema. Esto puede ser hecho encontrando que modificadores corresponde a estas teclas; usualmente esto será `Mod1` para `alter` y `Mod4` para `meta` o `windows`. Realizaremos las permutaciones necesarias en Edit → Preferences → Keyboard, selecionando A modifier → Equivalent for Mod4 y M modifier → Equivalent for Mod1.

A.3. Notes for Russian and Ukranian users

In order to type Russian (and similarly for Ukranian) text, you may several options:

- Select Russian as your default language in Edit → Preferences → Language → Russian. If TEXMACS starts with Russian menus, then this is done automatically if the Russian locale is set.
- Select Russian for an entire document using Document → Language → Russian.
- Select Russian for a portion of text in another document using Format → Language → Russian.

If your X server uses the xkb extension, and is instructed to switch between the Latin and Russian keyboard modes, you need not do anything special. Just switch your keyboard to the Russian mode, and go ahead. All the software needed for this is included in modern Linux distributions, and the xkb extension is enabled by default in `XF86Config`. With the xkb extension, keysyms are 2-byte, and Russian letters are at 0x6??. The keyboard is configured by `setxkbmap`. When X starts, it issues this command with the system-wide `Xkbmap` file (usually living in `/etc/X11/xinit`), if it exists; and then with the user's `~/.Xkbmap`, if it exists. A typical `~/.Xkbmap` may look like

```
ru basic grp:shift_toggle
```

This means that the keyboard mode is toggled by `l-shift r-shift`. Other popular choices are `control shift` or `control alternate`, see `/usr/X11R6/lib/X11/xkb/` for more details. This is the preferred keyboard setup for modern Linux systems, if you plan to use Russian often.

In older Linux systems, the xkb extension is often disabled. Keysyms are 1-byte, and are configured by `xmodmap`. When X starts, it issues this command with the system-wide `Xmodmap` (usually living in `/etc/X11/xinit`), if it exists; and then with the user's `~/.Xmodmap`, if it exists. You can configure the mode toggling key combination, and use a 1-byte Russian encoding (such as koi8-r) in the Russian mode. It is easier to download the package `xruskb`, and just run

```
xrus jcuken-koi8
```

at the beginning of your X session. This sets the layout jcuken (see below) and the encoding koi8-r for your keyboard in the Russian mode. If you use such keyboard setup, you should select Options → international keyboard → russian → koi8-r.

It is also possible to use the Windows cp1251 encoding instead of koi8-r, though this is rarely done in UNIX. If you do use `xrus jcuken-cp1251`, select cp1251 instead of koi8-r.

All the methods described above require some special actions to "russify" the keyboard. This is not difficult, see the Cyrillic-HOWTO or, better, its updated version

`http://www.inp.nsk.su/~baldin/Cyrillic-HOWTO-russian/Cyrillic-HOWTO-russian.html`

Also, all of the above methods globally affect all X applications: text editors (emacs, nedit, kedit...), xterms, TeXmacs etc.

If you need to type Russian only once, or very rarely, a proper keyboard setup may be more trouble than it's worth. For the benefit of such occasional users, TeXmacs has methods of Russian input which require no preliminary work. Naturally, such methods affect only TeXmacs, and no other application.

The simplest way to type some Russian on the standard US-style keyboard with no software setup is to select Edit → Preferences → Keyboard → Cyrillic input method → translit. Then, typing a Latin letter will produce "the most similar" Russian one. In order to get some Russian letters, you have to type 2- or 3-letter combinations:

Shorthand	for	Shorthand(s)	for
`A-" e`	ё	`A-" E`	Ё
`y o`	ё	`Y o` `Y O`	Ё
`z h`	ж	`Z h` `Z H`	Ж
`j tab`	ж	`J tab`	Ж
`c h`	ч	`C h` `C H`	Ч
`s h`	ш	`S h` `S H`	Ш
`s c h`	щ	`S c h` `S C H`	Щ
`e tab`	э	`E tab`	Э
`y u`	ю	`Y u` `Y U`	Ю
`y a`	я	`Y a` `Y A`	Я

Table A.1. Typing Cyrillic text on a Roman keyboard.

If you want to get, e.g., "сх", and not "ш", you have to type `s / h`. Of course, the choice of "optimal" mapping of Latin letters to Russian ones in not unique. You can investigate the mapping supplied with TeXmacs and, if you don't like something, override it in your `~/.TeXmacs/progs/my-init-texmacs.scm`.

If you select jcuken instead of translit, you get the "official" Russian typewriter layout. It is so called because the keys "qwerty" produce "йцукен". This input method is most useful when you have a Russian-made keyboard, which has additional Russian letters written on the key caps in red, in the jcuken layout (a similar effect can be achieved by attaching transparent stickers with red Russian letters to caps of a US-style keyboard). It is also useful if you are an experienced Russian typist, and your fingers remember this layout.

Those who have no Russian letters indicated at the key caps often prefer the yawerty layout, where the keys "qwerty" produce "яверты". Each Latin letter is mapped into a "similar" Russian one; some additional Russian letters are produced by `shift`-digits. TeXmacs comes with a slightly modified yawerty layout, because it does not redefine the keys `$`, `£`, `\`, which are important for TeXmacs, are not redefined. The corresponding Russian letters are produced by some `shift`-digit combinations instead.

Appendix B

Acerca de GNU TeXmacs

Sumario

GNU TeXmacs	
Sistemas soportados	La mayoría de los sistemas GNU/Linux
Copyright	© 1998–2002 por Joris van der Hoeven
Licencia	Licencia Pública General GNU
Sitios web	`http://www.texmacs.org` `http://www.gnu.org/software/texmacs`
Contacto	`contact@texmacs.org`
Correo regular	Dr. Joris van der Hoeven Dépt. de Mathématiques (Bât. 425) Université Paris-Sud 91405 Orsay Cedex Francia

Table B.1. Sumario de la principal información acerca de GNU TeXmacs.

B.1. La filosofía detrás de TeXmacs

B.1.1. Una corta descripción de GNU TeXmacs

GNU TeXmacs es un editor de texto cientifico libre, el cual fue tanto inspirado por TeX como por GNU Emacs. El editor le permite escribir documentos estructurados vía una interface wysiwyg (what-you-see-is-what-you-get, lo-que-ves-es-lo-que-tienes) y amigable al usuario. Nuevos estilos pueden ser creados por el usuario. El programa implementa algoritmos de composición tipográfica de alta-calidad y fuentes TeX, las cuales le ayudan a producir documentos de apariencia professional.

La alta calidad de composición tipográfica va más allá para las fórmulas automáticamente generadas, lo cual hace a TeXmacs apropiado como una interface para sistemas de álgebra computacional. TeXmacs también soporta el lenguage de extensión Guile/Scheme, así que usted puede personalizar la interface y escribir su propias extensiones al editor.

TeXmacs actualmente corre en la mayoría de los sistemas GNU/Linux (un procesador >200MHz y una memoria >32Mb son recomendadas) y sobre computadores sun. Conversores existen para TeX/LaTeX y están bajo desarrollo para HTML/MathML/XML. En el futuro, TeXmacs está planeado para evolucionar a una completa suite de oficina científica, con capacidades de hoja de cálculo, un editor de dibujos técnicos y un modo de presentación.

B.1.2. Por qué la libertad es importante para los científicos

Uno de los objetivos principales de TEXMACS es promover el desarrollo de software libre por y para los científicos, mediante la reducción significativa de los costos de producir interfaces de usuario de alta calidad. Si usted planea escribir una interface entre TEXMACS y otro software, entonces por favor contáctenos.

Como un matemático, estoy profundamente convencido que sólo los programas libres son aceptables desde un punto de vista científico. Veo dos razones principales para esto:

- Un resultado computado por un sistema "matemático", cuyo código fuente no es público, no puede ser aceptado como parte de una prueba matemática.

- Así como a un matemático debería ser capaz de construir sobre teoremas encima de otros teoremas, debería ser posible modificar y liberar algoritmos de software matemático libremente.

Sin embargo, es extraño, y una vergüenza, que los principales programas matemáticos que están siendo usados actualmente sean propietarios. La principal razón para esto es que los matemáticos muchas veces no consideran la programación como una actividad científica completa. Consecuentemente, el desarrollo de software útil es delegado a "ingenieros" y los programas resultantes son usados como cajas negras.

Esta subdivision de la actividad científica es muy artificial: con frecuencia es muy importante desde un punto de vista científico saber que está dentro de la caja negra. Recíprocamente, un profundo conocimiento científico usualmente conduce a la producción de mejor software. Consecuentemente, pienso que los científicos deberían abogar por el desarrollo de software como una actividad científica total, comparable a la escritura de artículos. Entonces es claro también que tal software debería ser difundido en una forma que sea compatible con los requirimientos de la ciencia: disponibilidad pública, reproductibilidad y usabilidad libre.

B.2. LOS AUTORES DE TEXMACS

El sistema GNU TEXMACS, el cual es parte del proyecto, GNU fue diseñado y escrito por Joris van der Hoeven. El sistema fue inspirado tanto por el sistema TEX, escrito por D. Knuth, y por EMACS, escrito por R. Stallman. Gracias especiales van a ellos, como también al C.N.R.S. (el instituto nacional francés para la investigación científica), el cual me emplea y me autorizó a distribuir líbremente este programa. Gracias ulteriores van a los contrubuyentes abajo.

B.2.1. Desarrolladores de TEXMACS

- Andrey Grozin nos ha ayudado constantemente con muchos issues: interfaces a varios sistemas de algebra computacional, soporte para Cyrilic, herramientas para la manipulación de diccionarios, etc.

- David Allouche reemplazó el preprocesador gencc por el sistema más estandard de plantillas C++. El también hizo muchos otros parches, reporte de errores y hizo mucha de la administración de TEXMACS.

- Dan Grayson me ayudó a implementar comunicaciones con sistemas de algebra computacional via tubos. El también proveyó algún soporte monetario para TeXmacs, e hizo muchos cometarios y sugerencias útiles.
- Karim Belabas diseño y desarrollo conmigo el primer protocolo para hacer una interface de TeXmacs con sistemas de computación científica o algebra computacional. El también implementó la interface con el sistema pari.
- Stéphane Payrard hizo un importante arreglo de errores para la destrucción de ventanas.
- Michael Graffam por su ayuda con la interface con GNU Octave.
- Michael Lachmann por su trabajo en la interface por venir con GNU R.
- Gwenael Gabard por algunos arreglos en el conversor LaTeX a TeXmacs.
- Felix Breuer por su ayuda en el soporte XML y una donación.
- Igor V. Kovalenko por su ayuda en la depuración de TeXmacs y unos pocos parches.
- Gareth McCaughan hizo varios parches y comentarios.
- Jonas Lööf por un procedimiento de instalación preciso sobre Cygwin.
- Rob Clark hizo un parche con mejoras al sistema de soporte de tiempo.

B.2.2. Administración de TeXmacs y material de soporte

- Jean-Claude Fernandez, Fabien Salvi y las otras personas del CRI albergan y administran el sitio web de TeXmacs.
- Álvaro Tejero Cantero mantiene el TeXmacs Wiki.
- Loic Dachary hizo TeXmacs accesible en Savannah.

B.2.3. Portando TeXmacs a otras plataformas

- Marciano Siniscalchi portó TeXmacs a Cygwin.
- Martin Costabel portó TeXmacs a MacOSX.
- Bruno Haible ayudó con portar TeXmacs al sistema SUN.
- Dan Martens y Stéphane Payrard están are trabajando sobre un porte a Windows.

B.2.4. Contribuyentes a los paquetes TeXmacs

- Ralf Treinen mantiene el paquete Debian para TeXmacs.
- Christophe Merlet y Bo Forslund ayudaron con hacer un paquete RPM portable.
- Lenny Cartier mantiene el RPM TeXmacs para Mandrake Cooker.

- Jean Pierre Demailly y Yves Potin hicieron a TeXmacs parte del proyecto CNDP para soportar el software libre.

B.2.5. Internacionalización de TeXmacs

Alemán. Hans Dembinski, Jan Ulrich Hasecke, Joris van der Hoeven, Thomas Langen, Ralf Treinen.

Checo. David Rezac.

Español. Álvaro Cantero Tejero, Pablo Ruiz Múzquiz, David Moriano Garcia, Offray Vladimir Luna Cárdenas.

Finlandés. Teemu Ikonen.

Francés. Michèle Garoche, Joris van der Hoeven.

Holandés. Joris van der Hoeven.

Hungaro. András Kadinger.

Italiano. Andrea Centomo, Lucia Gecchelin, Xav and Daniele Pighin.

Polaco. Robert Janusz.

Portugués. Márcio Laurini and Alexandre Taschetto de Castro.

Romano. Dan Ignat.

Ruso. Andrey Grozin.

Suizo. Harald Ellmann.

Ucraniano. Volodymyr Lisivka.

B.2.6. Otros contribuyentes

Gracias final van a todos aquellos quienes han contribuido a TeXmacs, por ejemplo enviando reportes de errores o dándonos sugerencias para lanzamientos futuros: Alexandre Abbes, Alessio Abogani, Till Adam, Murali Agastya, Guillaume Allègre, Larry D'Anna, Eizo Akiyama, Doublet Alban, Tom Alsberg, James Amundson, Ayal Anis, Javier Arantegui Jimenez, André Arnold, Uwe Assmann, Philippe Audebaud, Daniel Augot, Olaf Bachmann, Franky Backeljauw, Nick Bailey, Pierre Barbier de Reuille, Giovanni Maniscalco Basile, Luc Béhar, Odile Bénassy, Paul Benham, Roy C. Bentley, Attila Bergou, Christophe Bernard, Konrad Bernloehr, Karl Berry, Matthias Berth, Cédric Bertolini, Matthew Bettencourt, Anne-Laure Biolley, Benedikt Birkenbach, Jim Blandy, Christof Boeckler, Mohsen Bouaissa, Thierry Bouche, Adrien Bourdet, Didier Bretin, Jean-Yves Briend, Simon Britnell, Alexander M. Budge, Yoel Callev, Niclas Carlsson, Dominique Caron, António Carvalho, Michel Castagner, Topher Cawlfield, Carlo Cecati, Henri Cohen, Teddy Fen-Chong, Dominique Colnet, Claire M. Connelly, Christoph Conrad, Riccardo Corradini, Paulo Correia, Olivier Cortes, Robert J. Cristel, Maxime Curioni, Jason Dagit, Mike Davidson, Jean-Pierre Demailly, Peter Denisevich, Alban Doublet, Steingrim Dovland, Michael John Downes, Benjamin Drieu, Amit Dubey, Daniel Duparc, Tim Ebringer, Magnus Ekdahl, Ulf Ekström, Robin Fairbairns, Tony Falcone, Hilaire Fernandes, Juan Flynn, Jens Finke, Thomas Fischbacher, Cedric Foellmi, Christian Forster, Charlie Fortner, Stefan Freinatis, Michael P Friedlander, Nils Frohberg, Rudi Gaelzer, Maciej Gajewski, Lionel Garnier, Björn Gohla, Patrick Gonzalez, Nirmal Govind, Michael Graffam, Frédéric Grasset, Wilco Greven, Cyril Grunspan, Laurent Guillon, Harri Haataja, Irwan Hadi, James W. Haefner, Ola Hamfors, Aaron Hammack, Guillaume Hanrot, Karl M. Hegbloom, Jochen Heinloth, Ralf Hemmecke, Alain Herreman, Andreas Horn, Chu-Ching Huang, Ed Hurst, Karl Jarrod Hyder, Richard Ibbotson, Benjamin T. Ingram, Alexander Isacson, Michael Ivanov, Maik Jablonski, Frederic de Jaeger, Pierre Jarillon, Paul E. Johnson, Pierre-Henri Jondot, Antoun Kanawati, Tim Kaulmann, Mukund S. Kalisi, Jeremy Kephart, Iwao Kimura, Simon Kirkby, Ronny Klein, Matthias Koeppe, Denis Kovacs, Jeff Kowalczyk, Ralph Krause, Neel Krishnaswami,, Friedrich Laher, Winter Laite, Russell Lang, Christopher Lee, Milan Lehocky, Joerg Lippmann, Pierre Lorenzon, V.S. Lugovsky, Duraid Madina, Yael Maguire, Paul Magwene, Jeremiah Mahler, Vincent Maillot, Giacomo Mallucci, Sylvain Marchand, Bernd Markgraf, Eric Marsden, Chris Marston, Evan Martin, Alisdair McDiarmid, Phil Mendelsohn, Sébastien de Menten, Jean-Michel Mermet, Jon Merriman, Herve le Meur, Amir Michail, Arkadiusz Miskiewicz, Sasha Mitelman, Dirk Moebius, Jack Moffitt, Julian Morrison, Bernard Mourrain, Stephan Mucha, Nathan Myers, Nix N. Nix, Eduardo Nogueira, Immanuel Normann, Jean-Baptiste Note, Ralf Nuetzel, Kostas Oikonomou, Bill Page, Pierre Pansu, Bernard Parisse, Frédéric Parrenin, Fernández Pascual, Yannick Patois, Alen L. Peacock, François Pellegrini, Antonio Costa Pereira, Jacob Perkins, Bernard Perrot, Jean Peyratout, Jacques Peyriere, Yves Pocchiola, Benjamin Poussin, Isaías V. Prestes, Rui Prior, Julien Puydt, Nguyen-Dai Quy, Ramakrishnan, Adrien Ramparison, Kenneth Reinhardt, Diego Restrepo, Christian Requena, Chris Retford, Staffan Ringbom, Will Robinson, Juan Pablo Romero, Juergen Rose, Mike Rosellini, Filippo Rusconi, Philippe Sam-Long, Duncan Sands, Breton Saunders, Claire Sausset, David Sauzin, Gilles Schaeffer, Guido Schimmels, Rainer Schöpf, David Schweikert, Rui Miguel Seabra, Sami Sieranoja,, Vasco Alexandre da Silva Costa, Marciano Siniscalchi, Daniel Skarda, Murray Smigel, Luke Snow, Rodney Sparapani, Bas Spitters, Bas Spitters, Starseeker, Harvey J. Stein, Bernard Stloup, Peter Stoehr, James Su, Przemyslaw Sulek, Ben Sussman, Roman Svetlov, Milan Svoboda, Dan Synek, Pan Tadeusz, Sam Tannous, John Tapsell, Dung TaQuang, Gerald Teschl, Eric Thiébaut, Nicolas Thiery, Helfer Thomas, Reuben Thomas, Kurt Ting, Janus N. Tøndering, Marco Trevisani, Andreas Umbach, Miguel A. Valle, Rémi Vanicat, Harro Verkouter, Sawan Vithlani, Guy Wallet, Adam Warner, Thomas Wawrzinek, Maarten Wegewijs, Lars Willert, Grayson Williams, Ben Wise, Damien Wyart, Volker Zell, Oleg Zhirov, Richard Zidlicky, Sascha Ziemann, Reinhard Zierke, Paul Zimmermann.

B.2.7. Contactándonos

Usted puede contactarnos bien sea por correo electrónico a

```
contact@texmacs.org
```

o por correo regular a

```
Joris van der Hoeven
Dépt. de Mathématiques (Bât. 425)
Université Paris-Sud
91405 Orsay Cedex
Francia
```

Hay también varias listas de correo $\mathrm{T_EX_{MACS}}$:

```
texmacs-users@texmacs.org
texmacs-info@texmacs.org
texmacs-dev@gnu.org
```

B.3. Cambios importantes en TeXmacs

Abajo, describimos brevemente los cambios más importantes que han ocurrido en TeXmacs desde la versión 0.3.3.15. También mantenemos una más detallada bitácora de cambios.

En general, cuando se actualice a una nueva versión, le recomendamos hacer copias de seguridad de sus viejos archivos TeXmacs antes de abrirlos con una nueva versión de TeXmacs. En el improbable caso de que sus viejos archivos no abran en la forma correcta, por favor envie un mensaje de error a

```
bugs@texmacs.org
```

y envie su viejo documento como un archivo anexo. No olvide mencionar su versión de TeXmacs y el sistema que está usando.

B.3.1. Teclado (1.0.0.11 – 1.0.1)

Los atajos de teclado de TeXmacs han sido racionalizados. A continuación sigue una lista de los cambios principales:

- El `E-` prefijo ha sido renombrado a `M-`.
- `escape` es equivalente a `M-` y `escape` `escape` a `A-`.
- Comando dependientes del modo están ahora precedidos por `A-`. En particular, los acentos son tecleados usando `A-` en lugar de `E-`.
- Las variantes son ahora obtenidas usando `tab` en lugar de `*` y puede recorrerlas al revés usando `shift-tab`.
- Los caracteres griegos son ahora tecleados usando `A-C-`, `F5`, o el modificador hiper, el cual puede ser configurado en Edit → Preferences. También puede obtener caracteres griegos como variantes de caracteres latinos. Por ejemplo, `p tab` produce π.
- La significación de las teclas del cursor en combinación con lata teclas control, alt y meta ha cambiado.

Puede ahora escoger entre varios “aspectos y comportamientos” para el comportamiento del teclado en Edit → Preferences → look and feel. Por defecto está en Emacs, pero usted puede esciger Old style si usted quiere mantener el comportamiento al que puede estar acostumbrado.

B.3.2. Menus (1.0.0.7 – 1.0.1)

Varios cambios han sido hechos en los menus. A continuación se listan los cambios más importantes:

- Buffer has been renamed as Go.

- Varios items para File have been moved to View.
- Los items Edit → Import y Edit → Export han sido movidos a Tools → Selections.
- El menu Insert ha sido dividido en los menus Insert, Text y Mathematics.
- Los menús Text y Paragraph han sido colocado juntos en un menu Format.
- Options ha sido exparsido a lo largo de Document, Document, View y Edit → Preferences.

B.3.3. Archivos de estilo (1.0.0.4)

Muchos cambios han sido hechos en la organización de los archivos de estilo TeXmacs. Los archivos de estilo personales que dependen de paquetes TeXmacs intermediarios pueden requerir algunas ligeras adaptaciones.

Estamos trabajando hacia una estabilización de los archivos de estilo estándar y los paquetes. Al final de este proceso, debería ser fácil adaptar los archivos de estilo LaTeX existentes para revistas a TeXmacs a través de la personalización de estos archivos de estilo y paquetes estandard. Tan pronto como tengamos tiempo, planeamos proveer documentación en línea sobre como hacer esto en Help → Online documentation.

B.3.4. Material Tabular (0.3.5)

La forma en que el material tabular es tratado ha cambiado completamente. Se ha vuelto mucho más fácil editar tablas, matrices, arreglos de ecuaciones, etc. También, muchas nuevas características han sido implementadas, tales como color de fondo, bordes, acolchados, separación silábica, subtablas, etc. Sin embargo, la actualización de material tabular viejo podría algunas veces ser errónea, caso en el cual lo invitasmo a enviar un reporte de error.

B.3.5. Formato de documento (0.3.4)

El formato de documento TeXmacs ha cambiado profundamente a fin de hacer TeXmacs compatible con XML en el futuro. Principalmente, los viejos estilos de entornos como

```
<assign|env|<environment|open|close>>,
```

que son aplicados vía pares coincidentes `<begin|env>text<end|env>`, han sido reemplazados por macros

```
<assign|env|<macro|body|open<body>close>>,
```

que son aplicados via expasiones simples de macro `<expand|env|text>`. Similarmente, los pares coincidentes `<set|var|val>text<reset|var>` de cambios de variables de entorno son reemplazados por un constructo `<with|var|val|text>` (cercanos a los atributos XML). Desde un punto de vista técnico, estos cambios permiten varias complicaciones si e cuerpo de `text` consite de varios párrafos. Como una consecuencia, documentos malamente estructurados pueden algunas veces mostrarse diferente en una nueva versión (aunque yo sólo noté un cambio menor en mis propios documentos). Más aún, a fin de mantener un alto nivel de estructura en el documento, el comportamiento del editor con respecto a ambientes multipárrafo ha cambiado ligeramente.

Appendix C

Contribuyendo a GNU TeXmacs

C.1. Usar TeXmacs

Una de las mejores maneras de contribuir a GNU TeXmacs es usarlo mucho, hablar acerca de él a sus amigos y colegas, y reportarme acerca de los errores u otros comportamientos no naturales. Por favor mencione el hecho de que usted escribió artículos usando TeXmacs cuendo los envíe. Usted puede hacer esto poniendo la etiqueta made-by-TeXmacs en algún lugar dentro de su título usando la nota Texto → Título → Aviso de TeXmacs.

Además estas generales (pero muy importantes) formas de contribuir, su ayuda en los siguientes temas más específicos sería apreciada. No vacile contactarnos si usted quiere contribuir a esos o cualesquiera otros asuntos. En el menú Ayuda usted puede encontrar documentación acerca del código fuente de TeXmacs, su formato de documento, como escribir interfaces con otros formatos, y así.

C.2. Hacer donaciones al proyecto TeXmacs

Haciendo donaciones a TeXmacs a través de la organización SPI.

Una muy importante forma de soportar TeXmacs es donando dinero al proyecto. TeXmacs es actualemtne uno de los proyecto de SPI (Software in the Public Interest, Software en el Interés Público; vea `http://www.spi-inc.org`). Usted puede hacer donaciones de dinero a TeXmacs vía esta organización, mediante una notificación en su cheque o correo electrónico por una transferencia por cable que su dinero debe ir al proyecto TeXmacs. Usted puede hacer donaciones de equipos o servicios o donaciones a través de vendedores. Vea al sitio SPI para más información. Nosotros mantendremos una página Web con una lista de los donantes pronto (si usted accede a estar en la lista).

Detalles sobre como donar dinero.

Para hacer una donación, escriba un cheque o una orden monetaria a:

> *Software in the Public Interest, Inc.*

y envie un correo a la siguiente dirección:

> Software in the Public Interest, Inc.
> P.O. Box 502761
> Indianapolis, IN 46250-7761
> Estados Unidos

Para hacer una transferencia electrónica (esto funcionará para fuera de Estados Unidos también), usted necesita dar a su banco un número de enrutado y el número de la cuente como sigue:

> The SPI bank account is at American Express Centurion Bank.
> `Routing Number: 124071889`
> `Account Number: 1296789`

No olvide solicitar en su cheque o correo por tranferencia vía cable que su dinero debe ser gastado en el proyecto TeXmacs. En adició nusted puede especificar una propósito más específico sobre el cual le gustaría que gastaramos el dinero. Usted puede también contactarnos para una discusión más detallada sobre este aspecto.

Notas importantes.

Permita al tesorero de la SPI (`treasurer@spi-inc.org`) saber si usted tiene problemas. Cuando usted ha completado el cable electrónico, por favor envíe una copia del recibo a la dirección de arriba, así hay una copia de su donación. La copia que usted envia al tesorero es importante. Usted puede tambien querer contactar al equipo TeXmacs a fin de asegurarse que el dinero ha llegado a la cuenta TeXmacs.

Nota: *La dirección SPI y los números de cuenta pueden cambiar de tiempo en tiempo. Por favor no copia la dirección y el número de cuentas, sino en su lugar diríjase a la página* `http://www.spi-inc.org/donations` *para asegurarse que los donantes siempre verán la información más actualizada.*

Donaciones es Europa pueden ser hechas a través de nuestro socio en Alemana, ffis e.V. Si usted está interesado en su cuenta bancaria (para ahorrar costos de transferencia en moneda internacional), por favor verifique las instrucciones en `http://www.ffis.de/Verein/spi-en.html`.

C.3. CONTRIBUIR A LA DOCUMENTACIÓN DE GNU TEXMACS

Existe una alta necesidad de buena documentación sobre TEXMACS como también gente que esté deseosa de traducir la documentación existente a otros lenguajes. La intensión de este sitio es proveer documentación de alta calidad. Por esto, usted debe cuidadosamente leer las líneas guía sobre como escribir tal documentación.

C.3.1. Introducción sobre cómo contribuir

La documentación de alta calidad es tanto un asunto de contenido como de estructura. El contenido en sí mismo tiene que ser tan pedagógico como sea posible para el grupo de lectores objetivo. A fin de lograr esto, usted no debe vacilar en proveer tantos ejemplos y capturas de pantalla ilustrativas cuando sea adecuado. Aunque la documentación no pretende ser necesariamente completa, intemas proveer documentación relativamente estable. En particular, usted debe haber revisado su texto confrontando errores de pronunciación. La documentación más experimental debería ponerse en el directorio `incoming` o en el TEXMACS Wiki.

Es también importante que usted de a su documentación tanta estructura como sea posible, usando etiquetas especiales del archivo de estilo `tmdoc`. Esta estructura puede ser usada a fin de compilar automáticamente libros imprimibles desde su documentación, para hacerla apropiada para diferentes formas de visión, o para hacerla posible para buscar eficientemente cierto tipo de información en la documentación. En particular, debe siempre proveer información sobre copyright y licencia, como también indicaciones sobre cómo recorrer su información, si contiene muchos archivos.

C.3.2. Usando cvs

La presente documentación de TeXmacs es actualmente mantenida en `http://savannah.gnu.org` usando cvs (Concurrent Version System, Sistema de Versión Concurrente). A fin de contribuir, usted debe primero crear una cuenta allí. Cuando haya hecho esto, debe enviarme un correo a `vdhoeven@texmacs.org` y pedirme adicionar su nombre a la lista de documentadores. Después de eso, será capaz de modificar la documentación usando cvs. Para información sobre como hacer eso, debe consultar `http://savannah.gnu.org/cvs/?group_id=1747`.

De hecho, el sistema cvs no es ideal para nuestro propósito de documentación, porque no es muy dinámico. En el futuro, planeamos crear un sitio Web dedicado a la publicación, el cual nos permitirá salvar documentos directamente en la Web. Debería también permitir la conversión de la documentación a otros formatos, la compilación de libros, etc.

C.3.3. Convenciones para los nombres de archivos

La mayoría de la documentación debe ser organizada como una función del tópico en un arbol de directorios. Los subdirectorios en el directorio superior son los siguientes:

devel. Documentación para desarrolladores.

examples. Ejemplos de documentos TeXmacs.

incoming. Documentación por venir, la cual es aún un poco experimental.

main. La documentación principal.

meta. Como escribir documentación y la compilación de documentación.

Por favor trate de mantener el número de entradas por directorio razonablemente pequeño.

Los nombres de archivos en el directorio principal deben ser de la forma `nombre-tipo.lenguaje.tm`. En los otros directorios, ellos son de la forma `nombre.lenguaje.tm`. Aquí `tipo` es una indicación principal del tipo de documentación; debe ser uno de los siguientes:

adv. Documentación para usuarios avanzados.

man. Para inclusión en el manual TeXmacs.

tut. Para inclusión en el tutorial TeXmacs.

Usted debe tratar de mantener la documentación sobre el mismo tema junta, sin consideraciones del tipo. De hecho, esto le permite encontrar más fácilmente toda la documentación existente sobre un tópico particular. También, puede pasar que usted quiere incluir documentación la cual fue incialmente concebida para el tutorial o el manual. El `lenguaje` en el cual esta documentación ha diso escrita debe ser un código de dos letras como `en`, `fr`, etc. El `nombre` principal de su archivo debería ser el mismo para traducciónes en otros lenguajes. Por ejemplo `man-keyboard.en.tm` no debe ser traducido como `man-clavier.fr.tm`.

C.3.4. Información de Copyright & La Licencia de Documentación Libre

Toda la documentación en el sitio `texmacs-doc` cae bajo la Licencia GNU de Documentación Libre . Si usted escribe documentación para TeXmacs en este sitio, entonces tiene que estar de aceurdo que esta será distribuida bajo esta licencia también. La nota de copyright

```
Permission is granted to copy, distribute and/or modify this
document under the terms of the GNU Free Documentation License,
Version 1.1 or any later version published by the Free Software
Foundation; with no Invariant Sections, with no Front-Cover
Texts, and with no Back-Cover Texts. A copy of the license
is included in the section entitled "GNU Free Documentation
License".
```

debe ser especificada al final de *cada* archivo. Esto debe ser hecho dentro del macro tmdoc-license, en forma similar a como está al final del presente documento. Cuando automáticamente se genere un libro impreso desde varios archivos de documentación, esto nos habilitará para incluir la licencia sólamente una vez.

Usted mantiene (parte de) el copyright de toda la documentación que usted escribirá para TeXmacs en el sitio oficial `texmacs-doc`. Cuando usted u otros hagan adiciones a (o modificaciones en, o traducciones de) el documento, entonces usted debería adicionar su propio nombre (en un lugar apropiado, usualmente al final) a la información de copyright existente. La nota de copyright debe ser especificada usando la función tmdoc-copyright justo antes de la información de la licencia al final del documento. El primer argumento de esta función contien un año o un periodo. Cada argumento restante indica uno de los propietarios del copyright. Cuando se combinan (piezas de) varios documentos en otro, usted debería mezclar los propietarios del copyright. Para la información de cubierta (en un libro impreso por ejemplo), le está permitido listar sólo los autores principales, pero una lista completa debería ser dada en un lugar cláramente indicado.

C.3.5. Recorriendo la documentación TeXmacs

Como una regla general, usted debe evitar el uso de comandos de sección dentro de la documentación de TeXmacs y tratar de escribir pequeñas páginas de ayuda sobre tópicos bien identificados. En una segunda etapa, usted debe escribir "meta archivos de ayuda" que indique como recorrer la documentación en una forma automática. Esto le permite el uso de una página de ayuda para diferentes propósitos (un manual impreso, un tutorial orientado a la Web, etc.).

El estilo `tmdoc` provee tres macros de etiquetamiento para indicar cómo recorrer la documentación. El macro traverse es usado para encapsular regiones con información transversal. El macro branch indica una página de ayuda que debe ser considerada como una subsección y el macro continue indica una página que sigue. Tanto el macro branch como el continue toman dos argumentos. El primer argumento describe el enlace y el segundo argumento da la dirección física relativa del archivo enlazado.

Típicamente, al final de un meta archivo de ayuda usted encontrará varios macros branch o continue, dentro de un macro traverse. En la parte superior del documento, usted debe espeficicar un título para su documento usando el macro tmdoc-title. Cuanod se genera un manual impreso desde la documentación, una estructura capítulo-sección-subsección será automáticamente generada a partir de esta información y los títulos de los documentos. Alternaticamente, uno podría generar botones adicionales para la navegación dentro de la documentación usando un navegador.

C.3.6. Usar el estilo tmdoc

Además de los macros de información de copyright y macros transversales, los cuales han sido docuementados antes, el estilo `tmdoc` viene con cierto número de otros macros y funciones, las cuales usted debe usar cuando sea apropiado:

key. Este macro es usado para indicar una entrada de teclado como `C-x C-s`. Los macros especializados kbd-gen, kbd-text, kbd-math, kbd-symb, kbd-big, kbd-large, kbd-ia, kbd-exec y kbd-table son usados para entradas de teclado correspondientes a tipos específicos de acción o modos. Por ejemplo, kbd-math corresponde a atajos de teclado para operaciones matemáticas, tales como `A-f`, el cual inicia una fracción.

menu. Esta función con un número arbitrario de argumentos indica un menú como File o Document → Language. Las entradas de menu con automáticamente traducidas por esta función.

markup. Esta macro es usado a fin de indicar un macro o una función como section.

tmstyle. Este macro indica el nombre de un archivo estilo TeXmacs o un paquete como `article`.

tmpackage. Este macro indica el nombre de un paquete TeXmacs como `std-markup`.

tmdtd. Este macro indica de un d.t.d. de TeXmacs like `number-env`.

Note que los contenidos de ninguna de las etiquetas de arriba deben ser traducidos a lenguajes extrangeros. De hecho, para etiquetas de menú, las traducciones son hechas automáticamente, así se mantienen sincronizadas con las traducciones en los menus acutales de TeXmacs. En los casos de etiquetas, estilos, paquetes y d.t.d.s, es importante mantener el nombre original, porque a menudo corresponde a un nombre de archivo.

Los siguientes macros y funciones son usadas para propósitos de enlace e indexación, aunque ellos deben ser mejorados en el futuro:

simple-link. Este macro toma una URL x como argumento y es un hiperenlace con un nombre y destinación x.

hyper-link. Este macro is un hiperenlace usual.

concept-link. Este macro toma un concepto como argumento. Después un hiperenlace apropriado podría ser creado automaticamente desde esta y la otra documentación.

only-index. Indexa una cadena simple.

def-index. Definición de un nuevo concepto; el texto es impreso en itálica e indexado.

re-index. Reaparición de un concepto ya definido; el texto es impreso en romano y puesto en el índice.

Las siguientes etiquetas son también frecuentemente usadas:

icon. Enlace a un icono en un directorio central como `$TEXMACS_PATH/doc/images/pixmaps`.

screenshot. Enlace a una captura de pantalla. Las capturas de pantalla actuales son almacenadas en un directorio central como `$TEXMACS_PATH/doc/images/screenshots`.

scheme. El lenguage SCHEME.

framed-fragment. Para mostrar un pedazo de código en un marco agradable.

scheme-fragment. Para código SCHEME multi-párrafo.

tm-fragment. Para un pedazos de etiquetas de código $\mathrm{T_EX_{MACS}}$ en formato SCHEME.

descriptive-table. Para tablas descriptivas; tales tablas pueden ser usadas para documentar listas de atajos de teclado, diferentes tipos de etiquetas, etc.

El estilo `tmdoc` herededa de el estilo `generic` y usted debe usar macros como em, verbatim, itemize, etc. para este estilo donde sea apropiado.

C.4. INTERNACIONALIZACIÓN

El soporte de un máximo número de lenguajes extrangeros es otros de los mayores desafios en el cual su ayuda sería apreciada. Hacer las traducciones para soportar un nuevo lenguaje usualmente requiere varios días de trabajo. Por tanto recomendamos encontrar algnos amigos o colegas que estén deseosos de ayudarlo.

El procedimiento para adicionar un nuevo lenguaje es el siguiente

- Copie el archivo `english-new.scm` a `english-`*`sulenguaje`*`.dic` en `langs/natural/dic` y llene las correspondientes traducciones. Uste puede querer usar la herramienta de diccionario de Andrey Grozin en

  `http://www.texmacs.org/Data/dictool.py.gz`

  A fin de usarla, esté seguro de que Python está instalado en su sistema, descargue el archivo, ejecute gunzip sobre él, hágalo ejecutable y córralo.

- Cuénteme acerca de cualquier regla tipográfica especial en su lenguaje y conbinaciones de teclas útiles para producir caracteres especiales.

- Me preocupo de los asuntos de separación silábica y tipográficos, pero pruébelos.

- Si usted tiene suficiente tiempo, puede considerar la traducción de (parte de) la documentación existente.

Por supuesto, el soporte para lenguajes se desactualiza cada vez que nuevas características son adicionadas a $\mathrm{T_EX_{MACS}}$. Por esta razón, nosotros tambien mantenemos un archivo `miss-english-`*`sulenguaje`*`.dic` con toda la traducción perdida para su lenguaje, una ve que esta ha sido adicionada. Por favor no vacile enviar versiones incompeltes de `english-`*`sulenguaje`*`.dic` o `miss-english-`*`sulenguaje`*`.dic`; alguién más estará deseoso de completarlos

C.5. ESCRIBIR CONVERSORES DE DATOS

Si usted está familiarizado con $\mathrm{T_EX}$, $\mathrm{L^AT_EX}$, Html, Xml, Sgml, Mathml, Pdf, Rtf, o cualquier otro formato de datos usado frecuentemente, por favor considere contribuir para escribir buenos conversores para uno o más de esos formatos.

Escribir un conversor específico para Pdf no debería ser muy difícil, adaptando el archivo `src/Window/PsDevice/printer.cpp`. Conversores para otros formatos pueden ser más complicados de escribir, y podrían algunas veces requerir una colaboración más cercana con los autores principales de TeXmacs. En Ayuda → Formato del documento → Examinar encontratá detalles acerca del formato de datos TeXmacs y en Ayuda → Código fuente → Convertidores a otros formatos de datos damos algunas sugerencias que podrían ser útiles para esos proyectos.

C.6. Portar TeXmacs a otras plataformas

Teniendo sólo acceso a sistemas PC/Linux y SUN, estoy interesado en gente que quiera portar TeXmacs a otros sistemas Unix con X Window y mantener las correspondientes distribuciones. Si usted quiere hacer esto, debe tomar un vistaso a los archivos

```
configure.in
src/Basic/fast_alloc.cpp
```

Especialistas en `autoconf`, redhat y paquetes rpm son también bienvenidos a comunicar sus sugerencias, parches, etc.

Además de portar a TeXmacs a otros sistemas basados en Unix, sería agradable portar TeXmacs a Windows (y Mac OC). Por favor únase a la lista de correo `texmacs-dev@gnu.org` si usted quiere ayudar. Han habido discusiones sobre como hacer el puerto y en particular sobre cual interface gráfica de usuario portable (como Gtk, Qt, Wxwindows o GNUstep) debemos usar. Nuestra estratégia será primer poner todo el código dependiente de la GUI en un limpiamente espefcicada API TMGUI y entonces hacer el puerto actual. De hecho, esto nos permitirá soportar múltiples toolkits gráficos. Más detalles pueden ser encontrados en los archivos de la lista de correo `texmacs-dev@gnu.org`.

C.7. Haciendo interfaces de TeXmacs con otros sistemas

Es bien fácil escribir interfaces entre TeXmacs y otros sistemas de álgebra computacional u otros programas científicos con salida estructurada. Por favor considere escribir interfaces entre TeXmacs y su(s) sistema(s) favorito(s). Ya ha sido hechas interfaces entre TeXmacs con varios otros sistemas libres, como Giac, Macaulay 2, Maxima, GNU Octave, Pari, Qcl, gTybalt, Yacas. Documentación detallada sobre cómo adicionar nuevas interfaces está disponible en el menu Help → Interfacing.

C.8. Convertirse en un desarrollador TeXmacs

Aparte de la clase de contribuciones que han sido descritas en más detalle antes, hay muchos más asuntos donde su ayuda sería apreciada. Por favor tomé un vistazo a nuestros planes para el futuro para más detalles. Por supuesto, debería sentirse libre de venir con sus propias ideas y compartirlas con nosotros en la lista de correo `texmacs-dev@gnu.org`!

Index