Manuale per l'utente di GNU TeXmacs

Indice

Capitolo 1

Per cominciare

1.1. Convenzioni di questo manuale

In tutto il manuale di TeXmacs le voci dei menu saranno scritte utilizzando font di tipo *sans serif*, come in Documento, File → Carica o Testo → Forma del carattere → Corsivo. I comandi da tastiera saranno scritti con un font di tipo *typewriter* all'interno di un riquadro, come ad esempio `C-s`. A destra delle voci dei menu si vedono, se disponibili, le equivalenti combinazioni di tasti. Per tali combinazioni si utilizzano le abbreviazioni seguenti:

`S-`. Per le combinazioni che includono il tasto shift (Maius).

`C-`. Per le combinazioni che includono il tasto control (Ctrl).

`A-`. Per le combinazioni che includono il tasto alternate (Alt).

`M-`. Per le combinazioni che includono il tasto meta.

`H-`. Per le combinazioni che includono il tasto hyper.

Per esempio, `A-C-b` rappresenta `alternate-control-b`. Gli spazi all'interno delle abbreviazioni da tastiera indicano che sono premuti più tasti, o combinazioni di tasti, in sequenza. Per esempio, `M-t N b` rappresenta `meta-t` `N` `b`.

I tasti `alternate`, `meta` e `hyper` non sono disponibili su tutte le tastiere. Sui PC recenti, il tasto `meta` è spesso sostituito con il tasto `windows`. Nel caso in cui uno o più dei tasti modificatori non siano presenti sulla propria tastiera, si può utilizzare `escape` invece di `M-`, `escape escape` invece di `A-` e `F5`, `escape escape escape` o `A-C-` invece di `H-`. Per esempio, `escape w` è equivalente a `A-w`. È anche possibile configurare i tasti modificatori in modo da ottenere il massimo vantaggio dal potente insieme di abbreviazioni da tastiera fornito da TeXmacs.

Si noti che i menu e il comportamento della tastiera durante l'uso di TeXmacs sono *contestuali*, cioè essi dipendono dalla modalità corrente (modalità testo o "modalità matematica"), dal linguaggio corrente e dalla posizione del cursore nel documento. Ad esempio, in modalità matematica, ci sono delle abbreviazioni da tastiera speciali che servono per comporre le formule matematiche, ma che sono inutili nella modalità testo.

1.2. Configurare TeXmacs

Quando si utilizza TeXmacs per la prima volta, il programma si configura automaticamente in un modo che viene ritenuto il più adatto. Ad esempio, TeXmacs tenterà di determinare i parametri della lingua del sistema su cui lavora e il tipo di carta della stampante collegata. Talvolta la configurazione automatica può fallire oppure si può desiderare una configurazione alternativa. In tal caso, si deve accedere al menu Modifica → Preferenze e specificare le proprie preferenze.

In particolare, si raccomanda di configurare "l'aspetto" desiderato di TeXmacs. Per default, si utilizza l'aspetto di Emacs, che assicura una limitata compatibilità delle abbreviazioni da tastiera dei comandi di TeXmacs con quelle di Emacs. Inoltre, TeXmacs possiede un potente sistema di comandi rapidi da tastiera, che tenta di ottimizzare l'uso di tasti modificatori come `shift` e `control`. Comunque, su molti sistemi X Window, questi tasti modificatori non sono ben configurati e in tal caso si può provvedere alla loro riconfigurazione. Maggiori dettagli a questo proposito si possono trovare nella sezione riguardante la configurazione di TeXmacs.

1.3. Creare, caricare e salvare documenti

Quando si lancia TeXmacs senza alcuna opzione da linea di comando, l'editore crea automaticamente un nuovo documento. Si può creare un nuovo documento anche utilizzando File → Nuovo. I documenti appena creati non possiedono alcun nome. Per potergliene assegnare uno, si deve cliccare su File → Salva come.

Si raccomanda di assegnare un nome ai documenti subito dopo averli creati; questo eviterà di smarrire i documenti. Inoltre, si raccomanda di specificare il contesto globale dei propri documenti, quando necessario. Innanzitutto, si può specificare per il documento uno stile come articolo, libro o seminario utilizzando il menu Documento → Stile. Se si scrivono documenti utilizzando lingue diverse, e si vuole specificare la lingua del proprio documento si utilizza il menu Documento → Lingua. Analogamente, si può specificare un tipo di pagina utilizzando Documento → Pagina → Dimensioni.

Dopo aver modificato un documento, lo si può salvare mediante File → Salva. I documenti esistenti possono essere recuperati utilizzando File → Carica. Si noti che, utilizzando TeXmacs, si possono comporre diversi documenti in una stessa finestra; si può passare da un *buffer* all'altro usando il menu Vai.

1.4. Stampare documenti

Si può stampare il file corrente utilizzando File → Stampa → Stampa tutto. Per default, TeXmacs assume che si possieda una stampante a 600dpi per fogli di formato a4. Questi parametri di default possono essere modificati in Modifica → Preferenze → Stampante. Si può anche stampare in un file postscript File → Stampa → Stampa tutto su file (in questo caso i parametri di default per la stampante sono utilizzati per creare l'output) o File → Esporta → Postscript (in questo caso i parametri della stampante sono ignorati).

Quando TeXmacs è configurato adeguatamente, l'editor è *wysiwyg* (acronimo di "what you see is what you get", ossia ciò che vedi è ciò che scrivi): il risultato che si ottiene stampando il documento è esattamente quello che si vede sullo schermo. Per ottenere questo risultato, si deve selezionare Documento → Pagina → Tipo → Foglio e Documento → Pagina → Impaginazione su schermo → Margini come sul foglio. Ci si deve, inoltre, assicurare che i caratteri sullo schermo utilizzino lo stesso numero di dpi (punti per pollice) della propria stampante. La precisione di rendering dei caratteri può essere modificata utilizzando Documento → Carattere → Dpi. Attualmente, quando si modifica il numero di dpi, possono verificarsi cambiamenti minori dell'impaginazione che possono ripercuotersi globalmente sul documento nei salti tra le righe e tra le pagine. Questo problema sarà risolto in una versione futura.

Capitolo 2
Scrivere documenti semplici

2.1. Generalità sulla composizione di testi

Appena terminate le azioni preparatorie spiegate in precedenza, si può cominciare a comporre un testo in TeXmacs. I caratteri inglesi usuali e i simboli di punteggiatura si ottengono facilmente sulla maggior parte delle tastiere. I caratteri accentati delle lingue straniere possono essere ottenuti sistematicamente utilizzando i tasti di escape. Per esempio, "é" si ottiene scrivendo `A-' e`. Similmente, si ottiene "à" mediante `A-` a` e così via. La sillabazione delle parole lunghe ai bordi di righe successive viene gestita automaticamente. Affinchè la sillabazione delle lingue straniere avvenga correttamente, si deve specificare la lingua in cui viene scritto il documento nel menu Documento → Lingua.

Nella parte sinistra della barra a piè di pagina, compaiono lo stile del documento e le proprietà del testo nella posizione corrente del cursore. All'inizio, compare l'indicazione "testo roman 10", che significa che si sta scrivendo in modalità testo utilizzando un font roman a 10 punti e che non è stato scelto alcuno stile di documento. Si possono modificare le proprietà del testo (font, dimensioni del font, colore, lingua) nel menu Formato. Inoltre si possono cambiare le proprietà del testo su cui si sta lavorando selezionandone una regione e poi modificando le proprietà del testo attraverso il menu Testo. Alcune delle proprietà del testo possono anche essere modificate per il documento complessivo utilizzando i menu Documento → Carattere e Documento → Lingua.

Nella parte desta della barra a piè di pagina, compare il carattere o l'oggetto (come un cambiamento nelle proprietà del testo) che precede il cursore. Inoltre vengono mostrati tutti gli ambienti che sono attivi nella posizione del cursore. Queste informazioni dovrebbero aiutare l'utilizzatore ad orientarsi all'interno del documento.

2.2. Comporre testi strutturati

In genere, i documenti lunghi possiedono una struttura: essi sono organizzati in capitoli, sezioni e sottosezioni, contengono diversi tipi di testo, come testo ordinario, citazioni, note a piè di pagina, teoremi, ecc. Dopo aver selezionato uno *stile di documento* in Documento → Stile, TeXmacs si occupa dei problemi specifici di impaginazione, come la numerazione delle sezioni, delle pagine, dei teoremi e della composizione delle citazioni e delle note a piè di pagina, e così via.

Attualmente, sono stati implementati quattro stili standard per i documenti: lettera, articolo, libro e seminario. Lo stile seminario viene utilizzato per comporre dei lucidi per una presentazione. Appena viene selezionato uno di questi stili, si può organizzare il proprio testo in sezioni (si veda Testo → Sezione) e utilizzare specifici *ambienti*. Esempi di ambienti sono teorema, proposizione, nota e così via (si veda Testo → Ambiente). Altri esempi sono le liste di oggetti (si veda Testo → Lista puntata) o le liste numerate (si veda Testo → Lista numerata).

Quando si è acquisita una certa dimestichezza nell'uso di TeXmacs, è possibile aggiungere dei nuovi ambienti in un proprio file di stile. Assumiamo per esempio che si facciano spesso delle citazioni e che si voglia che queste appaiano in italico, con margini sinistro e destro di 1 cm. Invece di cambiare manualmente le proprietà del testo e del paragrafo ogni volta che si fa una citazione, è più comodo creare un ambiente citazione. Non solo questo renderà più rapida la creazione di una nuova citazione, ma renderà possibile modificare sistematicamente l'aspetto di tutte le citazioni presenti nel documento cambiando la definizione dell'ambiente citazione. Quest'ultima situazione si presenta per esempio quando si scopre *a posteriori* che si preferisce che le citazioni appaiano con un font più piccolo.

2.3. Tag di contesto

Gli esempi più semplici di struttura in un testo sono i tag di contesto. In Testo → content tag se ne può vedere una lista. I tag di contesto indicano che una certa porzione di testo è di un tipo particolare o che serve ad uno scopo specifico. Per esempio, i testi importanti dovrebbero essere evidenziati utilizzando il tag Enfatizzato. Il suo rendering di default utilizza caratteri in grassetto, come in questo **testo enfatizzato**. Comunque, il testo enfatizzato potrebbe avere delle rese diverse a seconda dello stile del documento. Per esempio, un testo enfatizzato potrebbe comparire con un colore diverso sui lucidi per una presentazione. Qui di seguito vi è una breve lista dei tag più comuni e il loro scopo:

Tag	Esempio	Scopo
strong	questo è **importante**	Indica una regione importante del testo
em	la *vera* via	Enfatizza una regione del testo
dfn	lo *gnu* è una bestia con le corna	Definizione di un concetto
samp	la legatura di `ae`: æ	Una legatura di caratteri letterali
name	il sistema LINUX	Il nome di una cosa particolare
person	mi chiamo JORIS	Il nome di una persona
cite*	*Moby Dick* di Melville	Una citazione bibliografica
abbr	io lavoro al C.N.R.S.	Una abbreviazione
acronym	il formato HTML	Un acronimo
verbatim	il programma dice `hello`	Testo verbatim come l'output dei programmmi
kbd	premere `return`	Testo di un comando da tastiera
code*	`cout << 1+1;` restituisce 2	Codice di un programma
var	`cp` *`src-file dest-file`*	Variabili in un programma

Tabella 2.1. Alcuni dei content tag più comuni.

Tag	Menu
strong	Enfatizzato
em	Enfatizza
dfn	Definizione
sample	Esempio
name	Nome
person	Persona
cite*	Citazione
abbr	Abbreviazione
acronym	Acronimo
verbatim	Testo semplice
kbd	Tastiera
code*	Codice
var	Variabile

Tabella 2.2. Equivalenze nel menu italiano.

2.4. LISTE

Usando Testo → Lista puntata si può costruire una lista non numerata. Per indicare gli elementi della lista si può utilizzare il tag di default oppure si può selezionare un tag particolare come • (puntine), − (trattini) o → (frecce). Le liste possono essere *nidificate*, come si vede nella lista di esempio qui di seguito:

- Primo elemento.
- Sottolista:
  - Un sottoelemento.
  - Un altro sottoelemento.
- Ultimo elemento.

Il tag di default appare in modi differenti a seconda del livello di nidificazione. Al livello più esterno, si utilizza il tag •, al secondo livello ∘, e così via. Si noti che, quando il cursore è all'interno di una lista, premendo il tasto `invio` viene creato automaticamente un nuovo elemento. Se la lunghezza di un elemento supera la lunghezza della riga, allora si può utilizzare `S-invio` per iniziare un nuovo paragrafo senza passare ad un altro elemento della lista.

Le liste numerate, che sono create utilizzando Testo → Lista numerata, si comportano in maniera analoga alle liste puntate, con l'unica differenza che gli elementi sono numerati. Qui di seguito vi è un esempio di una lista numerata creata utilizzando Testo → Lista numerata → I, II, III:

I. Primo elemento.

II. Secondo elemento.

III. Ultimo elemento.

L'ultimo tipo di liste serve per creare delle descrizioni. Vengono create utilizzando Testo → Descrizione e consentono di descrivere una lista di concetti:

Gnu. Una bestia pelosa ma gentile.

Moscerino. Vive solamente negli zoo.

2.5. AMBIENTI

Analogamente ai tag di contesto, gli ambienti sono utilizzati per caratterizzare delle porzioni del testo che hanno un significato particolare. Tuttavia, mentre i tag di contesto generalmente racchiudono piccole porzioni di testo, gli ambienti spesso racchiudono delle parti che si sviluppano su più paragrafi. Degli ambienti usati frequentemente in matematica sono teorema e dimostrazione, come nell'esempio seguente:

TEOREMA 2.1. *Non esistono interi positivi a, b, c e n con $n \geqslant 3$, tali che $a^n + b^n = c^n$.*

DIMOSTRAZIONE. Non ho lo spazio sufficiente per scrivere qui la dimostrazione. □

Si può attivare un ambiente utilizzando il menu Testo → Ambiente. Altri ambienti che forniscono un aspetto simile ai teoremi sono propositione, lemma, corollario, assioma, definizione. Si può utilizzare la macro dueto (attivata utilizzando `\ d u e t o invio`) per specificare la persona (o le persone) a cui appartiene il teorema, come nel seguente esempio

TEOREMA 2.2. (PITAGORA) *In circostanze opportune, vale* $a^2+b^2=c^2$.

Altri ambienti utilizzati frequentemente con un rendering simile a quello dei teoremi, ma che non enfatizzano il testo in essi racchiuso, sono importante, nota, esempio, avviso, esercizio e problema. Gli altri ambienti testo semplice, codice, citazione, citazione (più paragrafi) e verso possono essere utilizzati per introdurre testi con molti paragrafi o del codice, delle citazioni o delle poesie.

2.6. IMPAGINAZIONE

Come regola generale, TeXmacs si occupa dell'impaginazione del testo. Pertanto, sconsigliamo l'utilizzatore di impaginare personalmente il proprio documento, sebbene ciò non sia proibito. Per esempio, non si dovrebbero inserire degli spazi o delle righe bianche con l'intento di sostituire spazi orizzontali o verticali tra le parole o tra le righe; gli spazi addizionali dovrebbero essere inseriti esplicitamente utilizzando Inserisci → Spazio. In questo modo si renderà il documento più robusto nel senso che non si dovrà rimettere mano all'impaginazione se si apporta qualche piccola modifica, che influisce su alcune righe o alcune pagine, o una modifica maggiore, come un cambiamento dello stile del documento.

Sono stati implementati diversi tipi di comandi di spaziatura esplicita. Innanzitutto, si possono inserire degli spazi rigidi con larghezza e altezza fissate. Gli spazi orizzontali non possiedono un'altezza e possono essere più o meno estensibili. La lunghezza degli spazi estensibili dipende da come vengono sillabate le parole nel paragrafo. Inoltre, si possono inserire delle tabulazioni. Gli spazi verticali possono essere inseriti sia all'inizio che alla fine di un paragrafo: lo spazio addizionale tra due paragrafi è il massimo tra lo spazio verticale dopo il primo dei due paragrafi e lo spazio verticale che precede il secondo (diversamente dal TeX, questo evita che si creino degli spazi superflui tra due teoremi consecutivi).

Per quanto riguarda l'impaginazione dei paragrafi, l'utilizzatore può specificarne lo stile (giustificato, allineato a sinistra, centrato o allineato adestra), i margini e l'indentazione a sinistra (a destra) della prima (ultima) riga del paragrafo. L'utilizzatore può inoltre controllare gli spazi tra paragrafi e tra righe successive all'interno dei paragrafi.

Si può specificare l'impaginazione nel menu Documento → Pagina. Innanzitutto, si deve specificare come verranno visualizzate le pagine sullo schermo: quando si seleziona "foglio" come tipo di pagina in Documento → Pagina → Tipo, si vedranno esplicitamente le separazioni tra pagine successive. Per default, il tipo di pagina è "papiro", che evita le interruzioni tra le pagine durante la stesura del documento. Il tipo di pagina "automatico" imposta le dimensioni del foglio in modo che coincidano con le dimensioni della finestra su cui si sta lavorando. I margini della pagina e la larghezza del testo vengono specificate in Documento → Pagina → Impaginazione. Spesso, quando si lavora sullo schermo, conviene ridurre i margini della pagina; ciò si può fare utilizzando il menu Documento → Pagina → Impaginazione su schermo.

2.7. Il sistema di selezione dei font

In $\mathrm{T_EX_{MACS}}$, i font (i caratteri) hanno cinque caratteristiche principali:

- Il nome (roman, pandora, concreto, ecc.).
- La famiglia (o variante) (roman, macchina da scrivere o sans serif).
- La dimensione (una dimensione di base (in punti) e una dimensione relativa (normale, piccolo, ecc.)).
- La serie (grassetto, medio, leggero).
- La forma (destra, inclinata, maiuscoletto, ecc.).

Si noti che nel sistema di selezione dei font di $\mathrm{\LaTeX}\ 2\varepsilon$, il nome dei font e la loro famiglia sono un unico oggetto (ovvero, la famiglia). Si noti anche che la dimensione base dei font viene specificata per l'intero documento in Documento → Carattere → Dimensioni.

2.8. Personalizzare la tastiera

2.8.1. Regole generali sui prefissi

Poiché ci sono molte scorciatoie da tastiera, è importante avere qualche indicazione per classificarle in diverse categorie, per poterle memorizzare più facilmente. Come regola generale, le scorciatoie da tastiera che appartengono alla stessa categoria sono identificate da un prefisso comune. I più comuni di tali prefissi sono:

`C-x`. Le scorciatoie basate sul tasto control vengono utilizzate per i comandi di redazione usati frequentemente. Essi dipendono molto dalla "apparenza" definita in Modifica → Preferenze. Per esempio, se si utilizza un aspetto compatibile con EMACS, allora le combinazioni di tasti della forma `C-x` corrispondono ai comandi EMACS, come `C-y` per copiare parti di un testo.

`A-x`. Il tasto alternate è utilizzato per i comandi che dipendono dalla modalità in cui si sta lavorando. Per esempio, `A-s` produce del testo **enfatizzato** in modalità testo e una radice quadrata $\sqrt{}$ in modalità matematica. Si noti che che `escape escape` è equivalente a `A-`.

`M-x`. Il tasto meta è utilizzato per comandi $\mathrm{T_EX_{MACS}}$ generici, che possono essere usati in tutte le modalità. Per esempio, `M-!` produce un'etichetta. Esso viene anche utilizzato per comandi addizionali di redazione, come `A-w` per copiare del testo se si usa l'aspetto EMACS. Si noti che `escape` è equivalente a `M-`.

`H-x`. Il tasto hyper è utilizzato per produrre simboli speciali come i caratteri greci in modalità matematica. Si può configurare la propria tastiera in modo tale che il tasto Maius svolga il ruolo del tasto hyper. Il tasto `F5` è equivalente a `H-`.

Ricordiamo che i particolari tasti modificatori, che sono utilizzati per ottenere i prefissi `M-` e `H-`, possono essere configurati in Modifica → Preferenze.

2.8.2. Alcune scorciatoie da tastiera fondamentali

Alcune azioni standard della tastiera, valide in tutte le modalità sono:

`S-invio`. per iniziare un nuovo paragrafo.

`C-backspace`. per rimuovere un oggetto o un ambiente.

`A-spazio`. per inserire un piccolo spazio.

`A-S-spazio`. per inserire un piccolo spazio negativo.

`A-A-inizio`. per marcare manualmente l'inizio di una selezione.

`A-A-fine`. per marcare manualmente la fine di una selezione.

`M-<`. per andare all'inizio del documento.

`M->`. per andare alla fine del documento.

2.8.3. Scorciatoie da tastiera per la modalità testo

Le principali azioni della tastiera disponibili in modalità testo sono `$` per attivare la modalità matematica e le seguenti combinazioni di tasti per i caratteri accentati:

Abbreviazione		Esempio		Abbreviazione		Esempio	
`A-'`	Acuto ´	`A-' e`	é	`A-‘`	Grave `	`A-‘ e`	è
`A-^`	Circonflesso 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`	Ceco ˇ	`A-V s`	š
`A-O`	cerchio in alto ˚	`A-O a`	å	`A-.`	punto in alto ˙	`A-. z`	ż
`A-H`	Ungherese ˝	`A-H o`	ő				

Tabella 2.3. Caratteri accentati in TeXmacs.

I caratteri speciali ß, SS, œ, Œ, æ e Æ si ottengono mediante `S-F5 s`, `S-F5 S`, `S-F5 o e`, `S-F5 A E`, `S-F5 a` e `S-F5 A` rispettivamente.

In francese, si possono utilizzare anche `< <` e `> >` per ottenere le *guillemets* tipografiche francesi. In spagnolo, i punti esclamativi e i punti di domanda di apertura si ottengono mediante `! *` o `! ‘` e `? *` o `? ‘` rispettivamente.

2.8.4. Comandi ibridi e simulazione LaTeX

TeXmacs permette di attivare comandi LaTeX direttamente dalla tastiera, come viene spiegato qui di seguito. Si preme innanzitutto il taso `\` per attivare la modalità di comando ibrida LaTeX/TeXmacs. Poi si preme il comando che si desidera eseguire. Appena si finisce di scrivere il comando, a sinistra della barra a piè di pagina verrà visualizzato qualcosa del genere

```
<return>: comando da eseguire
```

Quando si batte il tasto `invio` il comando verrà eseguito. Per esempio, in modalità matematica, si può creare una frazione digitando `\ f r a c invio`.

Se il comando richiesto non è (riconosciuto come) un comando LaTeX, il programma controllerà innanzitutto se il comando esiste come macro, funzione o ambiente di TeXmacs (forniti dal file di stile). Se è così, la macro, la funzione o l'ambiente corrispondente vengono applicati (con il numero corretto di argomenti). Altrimenti, il programma assume che il comando corrisponda ad una variabile d'ambiente e ne richiede il valore. Il tasto `\` è sempre equivalente a uno dei comandi `M-i l`, `M-i e`, `M-i a`, `M-i #` o `M-i v`.

2.8.5. Oggetti dinamici

Alcuni oggetti più complessi possono avere molti *stati* durante il processo di redazione. Esempi di tali *oggetti dinamici* sono le etichette e i riferimenti, perché l'aspetto di un riferimetno dipende da un numero determinato dinamicamente. Molti altri esempi di marcatori dinamici si trovano nella documentazione relativa allo scrivere file di stile.

Quando si inserisce un oggetto dinamico come un'etichetta usando `M-!`, lo stato predefinito è *inattivo*. Questo stato inattivo consente di scrivere le informazioni rilevanti per l'oggetto dinamico, come il nome dell'etichetta nel nostro caso. Alcuni oggetti dinamici ammettono un numero arbitrario di parametri e uno nuovo di essi può essere inserito usando il tasto `tab`.

Quando si termina di scrivere le informazioni rilevanti per il proprio oggetto dinamico, si può battere il tasto `invio` per *attivare* l'oggetto. Un oggetto dinamico attivo può essere disattivato ponendo il cursore appena dietro l'oggetto e premendo il tasto `backspace`.

2.8.6. Personalizzazione della tastiera

L'utilizzatore può modificare il comportamento della tastiera. Per fare questo, suggeriamo di guardare innanzitutto i file nella directory `$TEXMACS_PATH/progs/keyboard`, dove è definito il comportamento standard della tastiera. Si potrà quindi ridefinire il comportamento della tastiera in un proprio file di inizializzazione.

Capitolo 3

Formule matematiche

Per poter comporre delle formule matematiche, si deve innanzitutto attivare la "modalità matematica" premendo il tasto `$` o inserendo un'equazione (utilizzando Testo → Matematica → Equazione). In modalità matematica, si hanno a disposizione dei comandi specifici e delle combinazioni di tasti per scrivere simboli matematici e formule. Per esempio, il prefisso `H-` può essere utilizzato per scrivere le lettere greche (si ricordi che `H-` è equivalente a `F5`, `escape escape escape` o `A-C-`).

L'editor agevola la scrittura di formule matematiche seguendo certe regole. Questa proprietà, che sarà ulteriormente sviluppata nelle versioni future, è utile quando si interagisce con un pacchetto di computer algebra. Attualmente, si deve, per esempio, scrivere esplicitamente il simbolo della moltiplicazione `*` tra i simboli a e b. Per default, scivendo `a b` si otterrà ab e non $a\,b$.

3.1. Oggetti matematici principali

Gli oggetti matematici principali vengono creati utilizzando il prefisso `A-` come segue:

Abbreviazione	Scopo	Esempio
`A-$`	Testo	$L=\{x\|x \text{ sia sufficientemente grande}\}$
`A-f`	Frazioni	$\frac{a}{b+c}$
`A-s`	Radici quadrate	$\sqrt{x+y}$
`A-S`	Radici n-esime	$\sqrt[3]{x^3+y^3}$
`A-n`	Negazioni	$\frac{\cancel{a}}{\cancel{b+c}}$

Tabella 3.1. Creazione dei principali oggetti matematici.

Apici, apici a sinistra, pedici e indici posti in alto sono creati come mostrato qui di seguito:

Abbreviazione	Scopo	Esempio
`'`	Apici	f' o $(g+h)'''$
`` ` ``	Apici a sinistra	${}^{\backprime}f$
`_`	Indici	x_n o x_{i_3}
`^`	Indici posti in alto	x^2, x_n^2 o $\mathrm{e}^{\mathrm{e}^x}$
`A-l _`	Indici a sinistra	${}_2x$
`A-l ^`	Indici in alto a sinistra	${}^{\pi}x$ o ${}^*_*\mathrm{He}^*_*$

Tabella 3.2. Creazione di apici, indici e indici posti in alto

3.2. Simboli matematici

In $\mathrm{T_EX_{MACS}}$, i caratteri greci si ottengono combinando il tasto modificatore `H-` con una lettera. Per esempio, `H-a` genera α e `H-G` genera Γ. Ricordiamo che il tasto `F5` è equivalente a `H-`, quindi ρ può anche essere ottenuto battendo i tasti `F5 r`. Analogamente, `F6`, `F7`, `F8` e `S-F6` possono essere utilizzati per scrivere caratteri in grassetto, calligrafici, fraktur e lavagna grassetto. Per esempio, `F8 m` produce $\mathfrak{m}$, `S-F6 R` produce $\mathbb{R}$ e `F6 F7 Z` produce $\boldsymbol{\mathcal{Z}}$.

I caratteri greci si possono ottenere anche come "varianti" dei caratteri latini utilizzando il tasto `tab`. Per esempio, `p tab` produce π. Il tasto `tab` viene utilizzato anche per ottenere delle varianti degli stessi caratteri greci. Per esempio, sia `H-p tab tab` che `p tab tab tab` generano ϖ.

Molti altri simboli matematici si ottengono con "naturali" combinazioni di tasti. Per esempio, `- >` genera $\rightarrow$, `- - >` produce $\longrightarrow$ e `> =` $\geqslant$. Analogamente, `| -` genera $\vdash$, `| - >` genera $\mapsto$ e `- > < -` genera $\rightleftarrows$. Qui di seguito riportiamo alcune regole generali che permettono di ottenere delle varianti dei simboli:

`tab`. è il tasto principale per ottenere delle varianti. Per esempio, `> =` produce $\geqslant$, ma `> = tab` produce $\geq$. Analogamente, `< tab` produce $\prec$, `< tab =` produce $\preccurlyeq$ e `< tab = tab` genera $\preceq$. Inoltre, `P tab tab` produce $\wp$ e `e tab tab` genera la costante $\mathrm{e} = \exp(1)$. Si può "tornare indietro" utilizzando `S-tab`.

`@`. viene utilizzato per inserire i simboli in cerchietti o in quadratini. Per esempio, `@ +` produce $\oplus$ e `@ x` produce $\otimes$. Analogamente, `@ tab +` genera $\boxplus$.

`/`. permette di ottenere le negazioni. Per esempio, `= /` produce $\neq$ e `< = /` produce $\nleqslant$. Si noti che `< = tab tab /` genera $\nleqq$, mentre `< = tab tab / tab` produce $\lneqq$.

`!`. si utilizza dopo le frecce per forzare il posizionamento dei caratteri seguenti al di sopra o al di sotto delle frecce stesse. Per esempio, `- - > ^ x` produce $\longrightarrow^x$, ma `- - > ! ^ x` produce $\xrightarrow{x}$.

Molti altri simboli, che non possono essere inseriti in modo naturale come nei casi descritti qui sopra, si possono ottenere utilizzando il prefisso `S-F5`. Qui di seguito riportiamo una piccola tabella con alcuni di tali simboli:

Abbreviazione	Simbolo	Abbreviazione	Simbolo
`S-F5 a`	$\amalg$		
`S-F5 n`	$\bigcap$	`S-F5 u`	$\bigcup$
`S-F5 v`	$\bigvee$	`S-F5 w`	$\bigwedge$

Tabella 3.3. Alcuni simboli che non possono essere ottenuti utilizzando in modo naturale le regole generali.

3.3. Operatori grandi

Le seguenti combinazioni di tasti vengono usate per creare dei simboli grandi:

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

Tabella 3.4. Operatori matematici grandi.

I segni grandi di integrale ammettono due varianti, a seconda di dove si desidera porre gli indici in alto e in basso. Per default, gli indici sono posti come nell'esempio seguente:

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

Per ottenere l'alternativa "con i limiti"

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

si deve utilizzare S-F5 L I. Analogamente, si può utilizzare la sequenza S-F5 L O per ottenere il simbolo $\oint$.

NdT : Per ottenere i simboli che non si riescono ad ottenere con la combinazione S-F5, si può utilizzare il menu matematico disponibile nella barra dei menu quando si lavora in modalità matematica.

3.4. Delimitatori grandi

I delimitatori grandi vengono creati utilizzando delle combinazioni di tasti come mostrato qui di seguito:

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

Tabella 3.5. Combinazioni di tasti per ottenere i delimitatori grandi.

In $\mathrm{T_EX_{MACS}}$, i delimitatori grandi possono essere di tre tipi: "delimitatori a sinistra", "delimitatori a destra" o "delimitatori interni". In generale, le parentesi aperte (, [, { e ⟨ sono i delimitatori a sinistra, le parentesi chiuse),], } e ⟩ sono i delimitatori a destra mentre |, / e \ sono i delimitatori interni. Ma questo stato può essere modificato utilizzando le combinazioni di tasti `A-l`, `A-r` e `A-m`. Per esempio, `A-l )` produce) considerata come delimitatore a sinistra.

In TeX e LaTeX, i "delimitatori interni" o "separatori" non esistono; in $\mathrm{T_EX_{MACS}}$ sono utilizzati per produrre delle barre verticali in alcune formule simili alla seguente

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

Tra un delimitatore a sinistra e uno a destra ci possono essere tanti delimitatori interni quanti se ne vogliono.

3.5. Accenti matematici larghi

La tabella seguente mostra come si possono porre dei larghi accenti matematici sopra i simboli o sopra intere formule. Infatti, alcuni di questi accenti diventano automaticamente larghi quanto la formula sotto di essi.

Abbreviazione	Esempio	Variante ampia	Abbreviazione	Risultato
`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`	$\breve{x}$	$\overbrace{x+y}$		

Tabella 3.6. Abbreviazioni da tastiera per ottenere accenti matematici ampi.

Capitolo 4

Tabelle

4.1. Creare tabelle

Per poter creare una tabella, si può utilizzare il menu Inserisci → Tabella o una delle seguenti combinazioni di tasti:

`M-t N t`. Per creare una tabella normale.

`M-t N T`. Per creare una tabella le cui celle sono centrate.

`M-t N b`. Per creare un "blocco" normale, le cui celle sono separate da linee.

`M-t N B`. Per creare un blocco le cui celle sono centrate.

In modalità matematica, sono disponibili altre strutture di tipo tabella:

`M-t N m`. Per creare una matrice.

`M-t N d`. Per creare un determinante.

`M-t N c`. Per creare una lista di scelta.

Anche l'ambiente `\eqnarray*` è una speciale struttura di tipo tabella, che si estende su più righe. Si può creare una lista di equazioni utilizzando Testo → Matematica → Equazioni.

Appena si crea una tabella, le sue dimensioni sono minime (solitamente 1×1) e le sue celle sono vuote. Nuove righe e colonne vengono inserite utilizzando le combinazioni di tasti `A-sinistra`, `A-destra`, `A-su` e `A-sotto`. Per esempio, `A-destra` crea una nuova colonna a destra della posizione corrente del cursore. Si può anche inserire una nuova riga sotto la posizione corrente del cursore premendo il tasto `invio`.

4.2. Modalità di formattazione

In TeXmacs, blocchi arbitrari di celle nella tabella possono essere formattati in modi particolari. Per esempio, si può assegnare un colore di sfondo a celle individuali, si può anche decidere che un'intera colonna sia centrata orizzontalmente. Per default, i comandi di formattazione operano su celle individuali, ma ciò può essere modificato tramite Tabella → Modo operativo della cella. Sono disponibili le seguenti modalità per le operazioni di formattazione:

`M-t m c`. Opera su celle individuali.

`M-t m h`. Opera sulle righe.

`M-t m v`. Opera sulle colonne.

`M-t m t`. Opera sull'intera tabella.

E' possibile anche selezionare un blocco di celle utilizzando il mouse e eseguire una singola operazione su quel rettangolo.

4.3. Allineamento delle celle e delle tabelle

L'operazione di formattazione più frequente è l'allineamento orizzontale o verticale di un blocco di celle. Si possono utilizzare le combinazioni di tasti `M-←`, `M-→`, `M-↑` e `M-↓` per allineare rapidamente più a sinistra, a destra, in alto o in basso.

Un allineamento specifico può anche essere selezionato nei menu Tabella → Allineamento orizzontale della cella e Tabella → Allineamento verticale della cella. In alternativa, si possono utilizzare delle scorciatoie da tastiera come `M-t h` x e `M-t v` x rispettivamente per l'allineamento orizzontale e verticale.

In maniera analoga, si può specificare come la tabella stessa deve essere allineata rispetto al testo circostante. Ciò si può fare tramite i sottomenu Tabella → Allineamento orizzontale della tabella e Tabella → Alineamento verticale della tabella, o utilizzando scorciatoie da tastiera della forma `M-t H` x o `M-t V` x. Qui x rappresenta `l` per "sinistra", `c` per "centrato", `r` per "destra", `b` per "in basso" e `t` per "in alto".

4.4. Specificare le dimensioni della tabella e delle celle

Utilizzando Tabella → Larghezza della cella → Imposta larghezza e Tabella → Altezza della cella → Imposta altezza si possono specificare la larghezza e l'altezza di una cella. La larghezza (o altezza) specificata può essere considerata in tre modi diversi:

Modo minimo. La larghezza attuale della cella sarà la minima tra la larghezza specificata e la larghezza dello spazio interno alla cella.

Modo esatto. La larghezza della cella sarà esattamente quella specificata.

Modo massimo. La larghezza attuale della cella sarà la massima tra la larghezza specificata e la larghezza dello spazio interno alla cella.

La larghezza del bordo e lo spazio tra le celle, che vedremo in seguito, vengono considerati nel calcolo delle dimenioni dello spazio interno alla cella.

Si possono inoltre specificare la larghezza e l'altezza dell'intera tabella in Tabella → Proprietà speciali della tabella. In particolare, si può specificare che la tabella occupi l'intera larghezza del paragrafo. Quando si specificano una larghezza o un'altezza, si può decidere come distribuire tra le celle lo spazio inutilizzato attraverso il menu Tabella → Proprietà speciali della cella → Distribuisci spazio inutilizzato. Per default, lo spazio inutilizzato viene distribuito equamente.

4.5. Bordi, spaziature e colori di sfondo

Si possono specificare le larghezze dei bordi e le spaziature di una cella nelle quattro direzioni possibili: a sinistra, a destra, in basso e in alto (si veda il menu Tabella → Bordo della cella). Per specificare le dimensioni dei bordi e le spaziature di una cella si possono anche utilizzare delle scorciatoie da tastiera della forma `M-t b` x e `M-t p` x.

Per default, la larghezza dei bordi delle celle nell'ambiente blocco è `1ln`, cioè la larghezza standard di una linea nel font corrente (come la larghezza di una barra di frazione). Questa larghezza viene utilizzata a destra e in basso di ogni cella (tranne quando la cella è nella prima riga o nella prima colonna). La spaziatura orizzontale di default per le celle è `1spc`: l'ampiezza di uno spazio bianco nel font corrente. La spaziatura verticale è di `1sep`: la separazione minima standard tra due rettangoli consecutivi.

Attraverso Tabella → Colore di sfondo della cella si può assegnare un colore di sfondo alle celle.

Si può dare un bordo e una spaziatura per l'intera tabella in Tabella → Proprietà speciali della tabella → Bordo. In questo caso, la spaziatura è applicata all'esterno del bordo.

4.6. Caratteristiche avanzate

Nei menu, si possono trovare molte altre caratteristiche speciali per le tabelle. Molto brevemente, queste includono le seguenti:

- Modifiche dello "span" di una cella e per consentirle di sovrapporsi alle celle vicine a destra e in basso.
- Creazione di intere sottotabelle all'interno delle celle.
- Correzione di fondo e altezza del testo, per poter far coincidere le linee base.
- Sillabazione orizzontale del contenuto delle celle e sillabazione verticale dell'intera tabella.
- Incollamento di più righe e/o colonne, in modo che le celle incollate diventino "parte dei bordi" delle celle rimanenti.
- Disattivazione della tabella, per poter vedere il suo "codice sorgente"
- Impostazione del "centro dell'estensione" di una tabella. In seguito a questa impostazione, le proprietà di formattazione di questa cella saranno utilizzate per le nuove celle create intorno a questo centro.
- Specificazione delle dimensioni massime e minime della tabella, che verranno rispettate durante le successive fasi di composizione del documento (ciò è particolarmente utile nella creazione di macro di tabelle).

Attualmente, tutte le tabelle vengono create in un ambiente come tabella, blocco, matrice, ecc. Quando ci si crea una propria macro di tipo tabella, si può utilizzare Tabella → Proprietà speciali della tabella → Estrai formato per estrarre il formato di una data tabella.

Capitolo 5

Collegamenti e contenuti generati automaticamente

5.1. Creare etichette, collegamenti e riferimenti

Si può creare una nuova etichetta inattiva utilizzando `M-!` o Inserisci → Collegamento → Etichetta e un riferimento a questa etichetta utilizzando `M-?` o Inserisci → Collegamento → Referenza. Si faccia attenzione a porre l'etichetta in un punto in cui il suo numero sarà corretto. Quando si assegnano etichette alle sezioni, la posizione raccomandata è quella subito dopo il nome della sezione. Quando si etichettano le equazioni, la posizione raccomandata è all'inizio dell'equazione.

È possibile creare degli hyperlink ad altri documenti utilizzando `M-i >` o Inserisci → Collegamento → Collegamento ipertestuale. Il primo campo dell'hyperlink è associato al testo, che risulta visualizzato in blu quando viene attivato. Il secondo campo contiene il nome di un documento, che può torvarsi sul web. Come usualmente accade per gli hyperlink, un collegamento della forma *`#etichetta`* punta ad una etichetta nello stesso documento e un collegamento della forma *`url#etichetta`* punta ad una etichetta che si trova nel documento all'indirizzo *`url`*.

In maniera simile, si può associare un'azione ad una parte di testo o a degli elementi grafici utilizzando `M-i *` o Inserisci → Collegamento → Azione. In questo caso, il secondo campo contiene uno script Guile/Scheme, che viene eseguito ogni volta che si fa doppio click due volte sul testo, dopo la sua attivazione. Per ragioni di sicurezza, tali script non sono sempre accettati. Per default, viene chiesto se si vuole accettare lo script; questo comportamento di default può essre modificato in Modifica → Preferenze → Sicurezza. Si noti che il comando Guile/Scheme:

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

valuta `shell-command` come comando shell.

Infine, si possono includere direttamente altri documenti in un dato documento utilizzando `M-i i` o Inserisci → Collegamento → Includi. Ciò consente per esempio di includere il listato di un programma nel proprio testo, in modo che le modifiche che verranno compiute sul programma si riflettano automaticamente nel testo in cui è esso stato incluso.

5.2. Inserire immagini

Si possono includere delle immagini nel testo utilizzando il menu Inserisci → Immagine. Attualmente, $\text{T}_\text{E}\text{X}_\text{MACS}$ riconosce i seguenti formati di file: `ps`, `eps`, `tif`, `pdf`, `pdm`, `gif`, `ppm`, `xpm` e `fig`. Qui, `gs` (cioè ghostscript) viene utilizzato per rendere immagini postscript. Se ghostscript non è istallato sul proprio sistema, lo si può scaricare da

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

Attualmente, gli altri formati di file vengono convertiti in file postscript utilizzando gli script `tiff2ps`, `pdf2ps`, `pnmtops`, `giftopnm`, `ppmtogif`, `xpmtoppm`. Se questi script non sono disponibili sul proprio sistema, si consiglia di contattare il proprio amministratore di sistema.

Per default, le immagini vengono visualizzate nelle loro dimensioni originali. Sono supportate le seguenti operazioni sulle immagini:

- Ritagliare l'immagine seguendo un rettangolo. Il punto in basso a sinistra dell'immagine originale è considerato come origine per specificare il rettangolo di ritaglio.
- Ridimensionamento di un'immagine. Quando si specifica una nuova larghezza, ma non l'altezza (o viceversa), l'immagine viene ridimensionata in modo da mantenere le proporzioni originali.
- Ingrandimento dell'immagine. Un modo alternativo per ridimensionare un'immagine, moltiplicandone la larghezza e l'altezza per una costante.

Si è incluso anche uno script per convertire in formato encapsulated postscript figure che contengono formule in LaTeX. Per includere una formula LaTeX in una figura `xfig`, si deve inserire la formula come testo, selezionare un font LaTeX e inserire un flag speciale nel testo.

5.3. Generare un sommario

Generare un sommario per il proprio documento è molto facile. E' sufficiente posizionare il cursore nella posizione in cui si vuole il sommario e cliccare su Testo → Automatico → Indice.

Per generare il sommario, si dovrebbe essere in una modalità in cui le interruzioni di pagina siano visibili (si selezioni il foglio in Documento → Pagina → Tipo), in modo che vengano calcolati i riferimenti appropriati ai numeri delle pagine. Poi, si utilizza Documento → Aggiorna → Indice o Documento → Aggiorna → Tutto per generare il sommario. Si dovrà compiere questa operazione diverse volte, fino a che il documento non subirà più cambiamenti. Infatti, la numerazione delle pagine può cambiare come conseguenza di modifiche nel sommario!

5.4. Compilare una bibliografia

Per il momento, TeXmacs usa `bibtex` per compilare le bibliografie. Il meccanismo per compilare automaticamente una bibliografia è il seguente:

- Si scrive un file `.bib` con tutti i riferimenti bibliografici. Questo file deve avere il formato di un file bibliografia standard per LaTeX.
- Usare Inserisci → Collegamento → Citazione e Inserisci → Collegamento → Citazione invisible per inserire le citazioni, che corrispondono agli elementi del file `.bib`.
- Nella posizione in cui si desidera che venga compilata la bibliografia, si clicchi su Testo → Automatico → Bibliografia. Alla richiesta, si inserisca uno stile `bibtex` (come `plain`, `alpha`, `abbrv`, ecc.) e il proprio file `.bib`.
- Usare Documento → Aggiorna → Bibliografia per compilare la bibliografia.

5.5. GENERARE UN INDICE ANALITICO

Per generare un indice, innanzitutto devono essere inserite le voci dell'indice nel proprio documento usando Inserisci → Collegamento → Voce dell'indice analitico. Al passo successivo, si deve porre il cursore nella posizione in cui si desidera che venga generato l'indice e di deve cliccare su Testo → Automatico → Indice analitico. L'indice verrà così generato in maniera simile al sommario.

Nel menu Inserisci → Collegamento → Voce dell'indice analitico, si trovano diversi tipi voci dell'indice. I più semplici sono "pricipale", "secondo livello", "terzo livello", che sono macro con uno, due e tre argomenti rispettivamente. Le voci della forma "secondo livello" e "terzo livello" possono essere utilizzate per voci di indici subordinati rispetto ad altri.

Le voci di un indice complesso ammettono quattro argomenti. Il primo è una chiave secondo la quale la voce sarà ordinata e dovrà essere una "-upla" (creata utilizzando `M-i <`) la cui prima componente è la categoria principale, la seconda una sottocategoria, ecc. Il secondo argomento delle voci di un indice complesso è bianca o "strong", nel qual caso il numero della pagina della voce corrispondente apparirà in grassetto. Il terzo argomento solitamente è vuoto, ma se si creano due voci dell'indice con il medesimo terzo argomento non vuoto, allora questo creerà un intervallo di numeri di pagine. Il quarto argomento, che pure è una -upla, è la voce stessa.

È anche possibile creare nell'indice una riga senza un numero di pagina utilizzando "interiezione" in Inserisci → Collegamento → Voce dell'indice analitico. Il primo argomento di questa macro è una chiave che indica come ordinare la riga dell'indice. Il secondo argomento contiene il testo effettivo. Questo costrutto può essere utile per creare diverse sezioni "A", "B", ecc. nel proprio indice.

5.6. COMPILARE UN GLOSSARIO

I glossari vengono compilati in maniera analoga agli indici, ma le voci non sono ordinate. Una voce "regolare' di un glossario contiene solamente un testo e per essa verrà generato un numero di pagina. Una voce "spiegata" di un glossario contiene un secondo argomento esplicativo. Una voce "duplica" può essere usata per creare un numero di pagina per la seconda ripetizione di una voce. Una riga di un glossario crea una voce senza un numero di pagina.

5.7. LIBRI E DOCUMENTI COMPOSTI DI MOLTI FILE

Quando un documento assume dimensioni molto grandi, si può desiderare di suddividerlo in parti più piccole. Questo permette sia di riutilizzare più facilmente i pezzi individuali in altri lavori, sia di migliorare i tempi di risposta dell'editor. Si può inserire un intero file in un altro utilizzando Inserisci → Collegamento → Includi. Per poter rendere più rapido il trattamento dei documenti inclusi, questi vengono messi in un buffer. Per aggiornare tutti i documenti inclusi si utilizza Strumenti → Aggiorna → Inclusioni.

Quando si scrive un libro, generalmente si mettono i singoli capitoli in file `c1.tm`, `c2.tm` until `cn.tm`. Poi si crea un file `book.tm` per l'intero libro, nel quale i file `c1.tm`, `c2.tm` until `cn.tm` sono inclusi utilizzando il meccanismo ora spiegato. Il sommario, la bibliografia, ecc. vengono generalmente messi nel file `book.tm`.

Per poter vedere i riferimenti incrociati agli altri capitoli quando si compone un particolare capitolo `ci.tm`, si può specificare `book.tm` come "file principale" per i file da `c1.tm` a `cn.tm` utilizzando Documento → Master → Collega. Attualmente, la numerazione dei capitoli non è gestita mediante questo meccanismo. Per assegnare una numerazione corretta, si deve assegnare manualmente la variabile d'ambiente `chapternr` all'inizio di ciascun file contenente un capitolo.

Capitolo 6

Strumenti per la redazione

6.1. Copia e incolla

Si possono selezionare parti di testo e formule tenendo premuto il tasto sinistro del mouse. Per cancellare la regione selezionata si utilizza Modifica → Taglia. Per copiare la regione selezionata si clicchi su Modifica → Copia. Poi, la si può copiare, quante volte si vuole, nella posizione del cursore utilizzando Modifica → Incolla. In alternativa, si può copiare una regione selezionata utilizzando il tasto centrale del mouse.

È anche possibile modificare le proprietà del testo di una regione selezionata. Per esempio, per trasformare del testo scritto in nero in rosso, lo si seleziona usando il tasto sinistro del mouse e poi si clicca su Formato → Colore → Rosso. Analogamente, se si seleziona una formula e si clicca su Inserisci → Matematica → Frazione, allora la formula diventa il numeratore di una frazione.

Quando si utilizza il meccanismo copia-incolla per comunicare con altre applicazioni, il testo viene copiato e incollato utilizzando il formato di dati TeXmacs. Si possono specificare altri formati per importare ed esportare mediante Strumenti → Selezioni → Importa e Strumenti → Selezioni → Esporta. Per default, le azioni per copiare e incollare sfruttano il buffer di testo primario. Utilizzando Modifica → Copia in e Modifica → Incolla da, si possono specificare tanti buffer diversi quanti se ne desiderano.

6.2. Cerca e sostituisci

Si può cominciare la ricerca in un testo premendo `C-s` o utilizzando il menu Modifica → Cerca. Durante una ricerca, la stringa cercata viene visualizzata a sinistra nella barra a piè di pagina. Ciascun carattere che si scrive viene aggiunto a questa stringa di ricerca e ogni sua ripetizione successiva viene racchiusa in un rettangolo rosso. Premendo `C-s` una seconda volta durante una ricerca, viene cercata la ripetizione successiva. Un suono indica che nel documento non sono state trovate altre ripetizioni; premendo ancora `C-s` la ricerca continuerà dall'inizio del documento. Si può premere il tasto `backspace` per annullare l'azione attivata dopo aver premuto un tasto durante la ricerca.

Generalmente, un testo viene cercato nel documento dalla posizione corrente del cursore in avanti. Si può eseguire una ricerca all'indietro utilizzando `C-r`. In una ricerca, verrà trovato solamente del testo nella stessa modalità e nello stesso linguaggio attivi nella posizione da dove comincia la ricerca. In altre parole, quando si cerca una x in modalità matematica, non si troverà alcuna x nel testo ordinario. Per il momento, la stringa di ricerca può contenere solamente testo ordinario e non simboli matematici o altre strutture di testo più complicate.

Per effettuare una sostituzione si utilizza `C-=` o Modifica → Sostituisci. Il programma richiede una stringa che dovrà essere messa al posto di quella che si vuole sostituire. Ad ogni ripetizione della stringa che si vuole sostituire il programma richiederà di scegliere tra sostituire (y) o no (n) la stringa, oppure sostituire questa e tutte le ripetizione successiva (a). Come nel caso della ricerca, anche il comando per la sostituzione e' sensibile alla modalità e alla lingua delle stringhe in gioco.

6.3. Controllo ortografico

Se sul proprio sistema è stato installato il programma `ispell`, allora lo si può utilizzare per controllare la presenza di errori ortografici tramite `M-$` o Modifica → Ortografía. Si noti che si deve verificare che i dizionari corrispondenti alle lingue utilizzate nel proprio testo siano stati installati nel sistema; cosa che generalmente accade per la lingua inglese.

Quando si attiva il controllo ortografico, sia sull'intero testo che su una sua regione selezionata, ad ogni parola che presenta un errore il programma porrà una richiesta all'utilizzatore e nella barra a piè di pagina compariranno le opzioni disponibili:

a). Accettare la parola e tutte le sue ripetizioni successive nel testo.

r). Sostituire la parola con una correzione che si inserirà.

i). Indicare che la parola "scorretta" è in realtà corretta e che deve essere inserita nel proprio dizionario.

1-9). Diversi suggerimenti di correzione per la parola.

Si noti che `ispell` controlla solamente le parole scorrette. Ma non verrà segnalato alcun errore grammaticale.

Il controllore ortografico utilizzerà il dizionario della lingua attiva nella posizione corrente del cursore o all'inizio di una parte selezionata. Solamente il testo in quella lingua verrà controllato. Se il proprio documento contiene parti di testo in lingue diverse, allora si dovrà attivare un controllo ortografico per ciascuna lingua utilizzata.

6.4. Annullare e ripetere

È possibile annullare gradualmente le modifiche eseguite in un documento dal momento in cui è stato lanciato TeXmacs. Ciò può essre fatto tramite Modifica → Annulla o utilizzando le combinazioni di tasti `M-[` o `C-/`. Le modifiche annullate possono essere ripristinate mediante Modifica → Rifai o `M-]`.

Per limitare l'uso della memoria, il numero di azioni successive che possono essere annullate è per default limitato a 100. È comunque possibile aumentare questo numero aggiungendo una linea di comando come la seguente

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

nel proprio file di inizializzazione (si veda Aiuto → Manuale → Personalizzare TeXmacs). Quando si specifica un numero negativo come dimensione massima di annullamenti, qualunque numero di azioni può essere annullato.

Capitolo 7
Proprietà avanzate di impaginazione

7.1. Flussi

I documenti complessi spesso contengono note a piè di pagina o oggetti fluttuanti, che sulle pagine hanno un aspetto diverso dal testo principale. Infatti, il contenuto di tali documenti complessi utilizza vari *flussi*, uno per il testo principale, uno per le note, uno per gli oggetti fluttuanti, un'altro per il testo su due colonne. I diversi flussi vengono spezzati tra le pagine in maniera indipendente l'uno dall'altro.

Per poter inserire note a piè di pagin, si può utilizzare Inserisci → Inserimento di pagina → Nota a piè di pagina. Il numero di colonne del testo può essere modificato in Documento → Paragrafo → Numero di colonne.

7.2. Oggetti fluttuanti

Gli oggetti fluttuanti si possono muovere nella pagina indipendentemente dal testo principale. Generalmente essi contengono delle figure o delle tabelle che sono troppo grandi per adattarsi bene nel testo principale. Un oggetto fluttuante può essere inserito tramite Inserisci → Inserimento di pagina → Oggetto fluttuante.

Si può anche creare un oggetto fluttuante e inserire direttamente in esso una figura o una tabella utilizzando Inserisci → Inserimento di pagina → Figura fluttuante e Inserisci → Inserimento di pagina → Tabella fluttuante. A volte si può desiderare di inserire figure molte piccole o tabelle in un oggetto fluttuante. Si può fare ciò usando Inserisci → Immagine → Figura piccola e Inserisci → Tabella → Tabella piccola.

Dopo aver creato un oggetto fluttuante, si può controllare la sua posizione mediante Inserisci → Inserimento di pagina → Posiziona fluttuante (quando ci si trova all'interno dell'oggetto fluttuante). Si può così specificare se si vuole che l'oggetto compaia all'inizio della pagina, in basso, direttamente nel testo o nella pagina successiva. Per default, l'oggetto fluttuante può comparire ovunque. Comunque, un oggetto fluttuante non potrà mai apparire nel testo principale a meno di tre righe dalla fine e dall'inizio della pagina.

7.3. Interruzioni di pagina

Le interruzioni di pagina possono essere controllate dall'utilizzatore in maniera molto precisa in Documento → Pagina → Salti. Nel sottomenu Algoritmo, si può specificare l'algoritmo da usare. L'interruzione di pagina professionale dà i migliori risultati durante la stampa, ma può rallentare la composizione del documento quando viene utilizzata interattivamente in modalità pagina. I salti di pagina sloppy sono i più rapidi, i salti medi sono come i professionali tranne per documenti su più colonne, per i quali l'algoritmo professionale è molto più lento.

Si può permettere che in casi eccezionali l'algoritmo di interruzione di pagina aumenti o riduca la lunghezza delle pagine nel sottomenu Limiti. La variazione della spaziatura tra paragrafi può essere specificata in Flessibilità. Il fattore di default è 1; un fattore più piccolo forza una spaziatura più rigida, ma la qualità delle interruzioni potrebbe risentirne.

Capitolo 8
I plugin di TeXmacs

Esistono diversi modi per personalizzare TeXmacs o estenderne le funzionalità: l'utilizzatore può definire file di stile personalizzati, personalizzare l'interfaccia o scrivere collegamenti con altri programmi esterni. Il sistema di plugin è un meccanismo universale per mettere insieme, all'interno di un unico pacchetto, una o più delle estensioni precedenti. I plugin sono allo stesso tempo facili da installare e semplici da scrivere e da mantenere.

8.1. Installare e usare un plugin

Di solito un plugin, che chiamiamo genericamente ***mioplugin***, viene reso disponibile in alcuni siti web in formato tarball binario con il nome

`mioplugin-version-architecture.tar.gz`

Se si installa TeXmacs nella directory `$TEXMACS_PATH`, allora si dovrà scompattare questo tarball nella directory `$TEXMACS_PATH/plugins`, utilizzando il comando

`tar -zxvf mioplugin-version-architecture.tar.gz`

In tal modo verrà creata la sottodirectory *mioplugin* in `$TEXMACS_PATH/plugins`. Dopo aver fatto ripartire TeXmacs il plugin dovrebbe essere automaticamente riconosciuto. Siete comunque pregati di leggere la documentazione relativa al plugin che volete utilizzare per capirne bene il funzionamento.

Importante 8.1. Se non avete eseguito personalmente l'installazione di TeXmacs o se non avete accesso diretto alla directory `$TEXMACS_PATH`, potete scompattare il tarball direttamente nella directory `$TEXMACS_HOME_PATH/plugins`. Infatti ricordiamo che `$TEXMACS_HOME_PATH` è per default `$HOME/.TeXmacs`. Dopo aver fatto ripartire TeXmacs il plugin dovrebbe, anche questa volta, essere automaticamente riconosciuto.

Importante 8.2. Se un plugin viene distribuito come tarball sorgente del tipo ***mioplugin-version***`-src.tar.gz` allora è necessario, prima di rilanciare TeXmacs, compilare il codice sorgente del plugin. In relazione ai diversi plugin (leggere le istruzioni) ciò di norma viene fatto attraverso il comando

`cd mioplugin; make`

o

`cd mioplugin; ./configure; make`

Importante 8.3. Per l'aggiornamento di un plugin è sufficiente rimuovere la versione vecchia contenuta in `$TEXMACS_PATH/plugins` o in `$TEXMACS_HOME_PATH/plugins`, utilizzando il comando

`rm -rf mioplugin`

e quindi reinstallare la versione aggiornata.

8.2. Scrivere un proprio plugin

Per scrivere un proprio plugin, ad esempio ***mioplugin***, si inizia creando la directory

`$TEXMACS_HOME_PATH/plugins/`*`mioplugin`*

in cui sistemare tutti i file (si ricorda che `$TEXMACS_HOME_PATH` per default è `$HOME/.TeXmacs`). Opzionalmente è possibile creare le seguenti sottodirectory:

`bin` — per file binari

`doc` — per la documentazione (non ancora supportata).

`langs` — per il supporto linguistico, come ad esempio il dizionario (non ancora supportata).

`lib` — per le librerie

`packages` — per i pacchetti di stile

`progs` — per i programmi in Scheme

`src` — per i file sorgente

`styles` — per i file di stile

L'idea è che i file presenti in queste sottodirectory vengano automaticamente riconosciuti all'avvio di TeXmacs. Ad esempio se è prevista una sottodirectory `bin` allora la stringa

`$TEXMACS_HOME_PATH/plugins/`*`mioplugin`*`/bin`

verrà automaticamente aggiunta, all'avvio, alla variabile ambientale `PATH`. Osserviamo come la struttura della sottodirectory di un plugin sia molto simile alla struttura della sottodirectory di `$TEXMACS_PATH`.

Esempio 8.4. Il tipo più semplice di plugin consiste solo di file di dati, ossia di una collezione di file di stile e di pacchetti. Per creare un plugin di questo genere è sufficiente creare le sottodirectory:

`$TEXMACS_HOME_PATH/plugins/`*`mioplugin`*

`$TEXMACS_HOME_PATH/plugins/`*`mioplugin`*`/styles`

`$TEXMACS_HOME_PATH/plugins/`*`mioplugin`*`/packages`

e sistemare i file di stile e i pacchetti nelle ultime due sottodirectory. Dopo aver rilanciato TeXmacs i file di stile e i pacchetti appariranno automaticamente nei menu Documento → Stile e Documento → Usa pacchetto.

Per plugin più complessi, che richiedono l'aggiunta di parti di codice in Scheme o in C++, è necessario scrivere un file di configurazione in Scheme

`$TEXMACS_HOME_PATH/plugins/`*`mioplugin`*`/progs/init-`*`mioplugin`*`.scm`

che dovrebbe contenere istruzioni del tipo

```
(plugin-configure myplugin
  configuration-options
```

dove le opzioni ***configuration-options*** descrivono le principali azioni da eseguire all'inizio, incluso il controllo del funzionamento del plugin stesso. Nella sezione seguente descriveremo alcuni semplici plugin e la loro configurazione. Molti altri esempi sono contenuti nella directory

    $TEXMACS_PATH/examples/plugins

Alcuni di questi verranno comunque descritti in dettaglio nel capitolo relativo alla stesura di nuove interfacce.

8.3. Un esempio di plugin in Scheme

Il plugin `world`.

Consideriamo il plugin `world` nella directory

    $TEXMACS_PATH/examples/plugins

Questo plugin mostra come estendere $\mathrm{T_EX_{MACS}}$ aggiungendo parti di codice scritte in linguaggio Scheme al file

    world/progs/init-world.scm

Per testare il plugin `world` è necessario copiare ricorsivamente la directory

    $TEXMACS_PATH/examples/plugins/world

in `$TEXMACS_PATH/plugins` o in `$TEXMACS_HOME_PATH/plugins`. Dopo aver rilanciato $\mathrm{T_EX_{MACS}}$ il plugin dovrebbe essere automaticamente riconosciuto (un menu World dovrebbe comparire nella barra dei menu).

Come funziona.

Questo file contiene essenzialmente questa parte di codice:

```
(define (world-initialize)
  (menu-extend texmacs-extra-menu
    (=> "World"
        ("Hello world" (insert-string "Hello world")))))

(plugin-configure world
  (:require #t)
  (:initialize (world-initialize)))
```

Le opzioni di configurazione `:require` specificano le condizioni che devono essere soddisfatte per permettere a $\mathrm{T_EX_{MACS}}$ di riconoscere il plugin (ad esempio controllando se alcuni programmi sono o meno disponibili nel sistema). La configurazione fallisce se le richieste formulate non vengono soddisfatte.

L'opzione `:initialize` specifica quale istruzione deve essere eseguita in fase di inizializzazione (modulo soddisfazione delle richieste precedenti). Nel nostro esempio abbiamo solamente creato un nuovo livello di menu World con la voce World → Hello world, che permette di inserire il testo "Hello world". In generale la routine di inizializzazione dovrebbe essere molto breve e, se necessario, richiamare un ulteriore modulo per l'effettiva inizializzazione del plugin. Mantenendo semplice la struttura del file `init-`***`mioplugin`***`.scm` verranno ridotti i tempi di avvio di $\mathrm{T_EX_{MACS}}$.

8.4. Un esempio di plugin in C++

Il plugin `minimal`.

Consideriamo l'esempio del plugin `minimal` nella directory

`$TEXMACS_PATH/examples/plugins`

Il plugin è costituito dai seguenti file:

`minimal/Makefile`

`minimal/progs/init-minimal.scm`

`minimal/src/minimal.cpp`

Per testare il plugin è necessario copiare ricorsivamente la directory

`$TEXMACS_PATH/examples/plugins/minimal`

in `$TEXMACS_PATH/progs` o in `$TEXMACS_HOME_PATH/progs`. Quindi si esegue `Makefile` utilizzando

`make`

in modo da compilare il file `minimal.cpp` e creare il file binario

`minimal/bin/minimal.bin`

Dopo aver rilanciato TeXmacs il plugin dovrebbe essere automaticamente riconosciuto.

Come funziona.

Il plugin `minimal` è un esempio di interfaccia minimale tra TeXmacs e un programma esterno; il programma esterno `minimal.cpp` viene spiegato in dettaglio nel capitolo relativo alla scrittura di interfacce. Il file di inizializzazione `init-minimal.scm` contiene essenzialmente il seguente codice:

```
(plugin-configure minimal
  (:require (url-exists-in-path? "minimal.bin"))
  (:launch "minimal.bin")
  (:session "Minimal"))
```

L'opzione `:require` controlla se `minimal.bin` esiste nel path, percui fallirà se si dimentica di eseguire `Makefile`. L'opzione `:launch` specifica come lanciare il programma esterno. L'opzione `:session` indica che sarà possibile creare una sessione per il plugin `minimal` utilizzando il menu Testo → Sessione → Minimal.

8.5. Opzioni di configurazione per i plugin

Come spiegato in precedenza, il file Scheme di configurazione `*mioplugin*/progs/init-*mioplugin*.scm`, relativo a un plugin con nome ***mioplugin***, contiene istruzioni del tipo

```
(plugin-configure mioplugin
  configuration-options)
```

Di seguito sono elencate le opzioni possibili per ***configuration-options***:

(:require *condition*) — questa opzione serve per specificare una ***condizione*** che deve essere soddisfatta per garantire il funzionamento corretto del plugin. Tipicamente si va a verificare se nel sistema sono presenti alcuni file binari o alcune librerie necessarie per il funzionamento. Se questa condizione fallisce TeXmacs continuerà a lavorare come se il plugin non esistesse. In questo caso la parte di configurazione successiva del plugin non verrà letta. Per questa ragione l'opzione `:require` viene di norma scritta per prima nella lista delle opzioni di configurazione;

(:version *version-cmd*) — questa opzione serve per specificare un'espressione Scheme ***version-cmd*** che valuta la versione del plugin;

(:setup *cmd*) — questo comando viene eseguito solo quando la versione del plugin cambia da una esecuzione di TeXmacs all'altra. Ciò capita principalmente quando viene installata una nuova versione di TeXmacs o viene aggiunta qualche altra applicazione accessoria;

(:initialize *cmd*) — questa espressione esegue l'espressione Scheme ***cmd***. Tipicamente viene inserita subito dopo l'opzione `:require` in modo che il plugin venga configurato solamente una volta appurata la sua esistenza. Per plugin che richiedono numerose istruzioni per essere programmati è importante che il file ***mioplugin***`/progs/init-`***mioplugin***`.scm` sia piccolo in quanto esso viene eseguito ad ogni avvio di TeXmacs. Per ridurre il tempo di caricamento di TeXmacs la maggior parte dei comandi Scheme possono essere inseriti in moduli separati, alcuni dei quali possono essere caricati attraverso comandi di inizializzazione;

(:launch *shell-cmd*) — questa opzione specifica che il plugin riesce a valutare espressioni su una pipe; ciò avviene utilizzando un'applicazione accessoria lanciata da riga di comando con il comando ***shell-cmd***;

(:link *lib-name export-struct options*) — questa opzione è simile a `:launch`, salvo che ora l'applicazione esterna viene collegata dinamicamente. Per ulteriori informazioni si rimanda alla sezione relativa ai link dinamici;

(:session *menu-nome*) — questa opzione indica che il plugin supporta una valutazione per sessioni interattive da shell. Un oggetto de tipo ***menu-oggetto*** verrà inserito nel menu Testo → Sessione per lanciare questo tipo di sessioni;

(:serializer ,*fun-nome*) — se il plugin si può utilizzare in modalità di valutazione allora questa opzione specifica la funzione Scheme ***fun-nome*** utilizzata per trasformare gli alberi TeXmacs in stringhe;

(:commander ,*fun-nome*) — questo comando è simile all'opzione `:serializer` con la differenza che esso viene usato per trasformare comandi speciali in stringhe;

(:tab-completion ,*flag*) — questo comando indica se il plugin supporta o meno i completamenti tramite tabulatore.

Capitolo 9

Utilizzare GNU $\text{T}_{\text{E}}\text{X}_{\text{MACS}}$ come un'interfaccia

Una caratteristica importante di $\text{T}_{\text{E}}\text{X}_{\text{MACS}}$ è la sua capacità di comunicare con sistemi esterni in sessioni di tipo terminale. Tipicamente, è possibile valutare i comandi di un sistema esterno di computer algebra in una sessione e visualizzare i risultati in modo grafico opportuno. In tali sessioni è anche possibile valutare comandi di shell e programmi in Scheme.

9.1. Utilizzo base

Si può dare inizio ad una sessione dal menu Testo → Sessione. Una sessione consiste di una sequenza di ambienti input e output, e tra questi, eventualmente, delle parti di testo. Quando si preme il tasto `invio` all'interno di un ambiente di input di una sessione, il testo nell'ambiente stesso viene valutato e il risultato viene visualizzato in un ambiente di output.

Quando si inserisce un comando in una sessione, l'applicazione tenta di eseguirlo. Molti comandi possono essere lanciati conteporaneamente in uno stesso documento, ma l'output sarà attivo solamente nella sessione dove c'è il cursore e nella posizione del cursore stesso. Per questo motivo, si raccomanda di utilizzare buffer diversi per esecuzioni parallele. Le esecuzioni possono essere interrotte dalla barra delle icone. Inoltre è possibile disconnettere (chiudere) l'applicazione; in questo caso non potranno essere eseguiti altri comandi nella sessione corrispondente.

Nella barra delle icone propria della sessione sono a disposizione alcuni bottoni per selezionare l'input in modalità matematica e per interrompere l'esecuzione. Se implementata per il sistema dato, la modalità matematica per l'input consente di inserire i dati in entrata in formato grafico bidimensionale. Altri due bottoni consentono di interrompere l'esecuzione di un particolare comando (ma ciò non funziona bene per alcuni sistemi) o di disconnettere l'esecuzione del sistema esterno. Quando si preme il tasto `invio` nell'input di un sistema non connesso, questo verrà automaticamente riconnesso.

9.2. Sistemi supportati

Se si guarda quanto contenuto nel menu Inserisci → Sessione si noterà che compaiono solo i sistemi che sono attualmente installati sulla vostra macchina. Fanno eccezione le sessioni scheme e da terminale che sono sempre disponibili.

Troverete sotto una breve lista di sistemi di computer algebra che possono essere interfacciati con $\text{T}_{\text{E}}\text{X}_{\text{MACS}}$. Esistono interfacce anche per alcuni sistemi di computer algebra proprietari. Le informazioni relative a questi ultimi sono disponibili nella documentazione ad essi allegata.

9.2.1. Sessioni scheme e sessioni da terminale

In una "sessione da terminale" è possibile valutare comandi dati da un terminale. Ogni input o output è verbatim. Nessuna particolare utilità per istruzioni da linea di comando (tipo il completamento automatico) è stata ancora implementata. L'output di un comando da terminale viene visualizzato gradualmente via via che il programma esegue le istruzioni.

In una "sessione SCHEME" si possono valutare programmi GUILE/SCHEME. L'input è un testo verbatim che viene valutato e, successivamente, viene mostrato il risultato. Nessun meccanismo di visualizzazione graduale è ancora stato implementato per le sessioni SCHEME.

9.2.2. Giac

GIAC è un sistema di computer algebra che può essere scaricato dal sito

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

9.2.3. GTybalt

GTYBALT è un sistema di computer algebra costruito su GINAC, CLN e basato su un programma per interpretare comandi C e C++. Per ulteriori informazioni si veda il sito

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

9.2.4. Macaulay 2

MACAULAY 2 è un nuovo software sviluppato come supporto alle ricerche in geometria algebrica. Il software è disponibile come codice sorgente per un eventuale porting, e in forma compilata per LINUX, SUN OS, SOLARIS, WINDOWS e qualche altra macchina unix. Si può scaricare questo programma dal sito

`http://www.math.uiuc.edu/Macaulay2`

9.2.5. Maxima

MAXIMA non è solamente uno dei più vecchi e migliori sistemi di computer algebra in circolazione, ma è anche un sistema dedicato per il quale esiste una libera implementazione. Per ottenere questo software potete consultare il sito

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

La versione supportata è MAXIMA 5.6 basata su GCL. Per la corrispondente versione basata su CLISP è necessario editare il file `tm_maxima` e rimpiazzare `-load` con `-i`. Per MAXIMA 5.9-pre, replace `-load` con `-p`. Problemi noti:

- se si preme `invio` quando un comando è incompleto (tipicamente non concluso con `;` o `$`), l'interfaccia si blocca;
- se si causa la comparsa del Lisp break prompt l'interfaccia si blocca;
- il comando `info` non è supportato (è definito nel Lisp soggiacente e difficile da supportare in modo portabile);

- alcuni comandi nel debugger funzionano, altri (incluso `:c`) misteriosamente non funzionano;
- il comando `load` si comporta talvolta in modo strano.

9.2.6. Pari

Pari è un pacchetto software utilizzabile nell'ambito della teoria dei numeri. Esso consiste in una libreria C, `libpari` (con un nucleo opzionale assembler per alcune architetture popolari) e del calcolatore interattivo programmabile `gp`. Il software Pari si può scaricare dal sito

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

Per poter usare questo software in TeXmacs è necessario disporre di una versione superiore a Pari-2.1.0 (per controllare la versione di Pari digitare `gp --version`).

9.2.7. Qcl

QCL è un linguaggio di programmazione di alto livello indipendente dal particolare tipo di architettura per computer quantistici; la sua sintassi deriva da linguaggi procedurali classici come il C o il Pascal. Attraverso QCL è possibile implementare e simulare algoritmi quantistici (che includano anche componenti classiche) in un formalismo consistente. L'interfaccia TeXmacs è particolarmente utile per visualizzare gli stati quantistici in modo leggibile. Ulteriori informazioni si hanno al sito

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

Dalla versione 1.0.0.8, TeXmacs supporta QCL 0.4.3 o versioni successive. Gli utilizzatori di versioni precedenti devono aggiornarle se desiderano utilizzarlo in TeXmacs.

9.2.8. Yacas

Yacas, come suggerisce il nome, è un altro sistema di computer algebra. In esso sono implementati: calcolo con precisione arbitraria, manipolazione di numeri razionali, vettori, numeri complessi e matrici (incluse inversioni, calcolo di determinanti e soluzione di equazioni matriciali), calcolo di derivate, serie di Taylor, metodi numerici (metodo di Newton) e molti altri algoritmi. Il linguaggio supporta funzioni e variabili definite dall'utente. Esiste un supporto per il trattamento di polinomi in una indeterminata, per il calcolo integrale e per il calcolo tensoriale. Potete scaricare Yacas al sito

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

Capitolo 10

Sommario dei principali tag di TeXmacs

10.1. Base comune alla maggior parte degli stili

Il d.t.d. `common-base` contiene le basi virtualmente comuni a tutti gli stili. Esso è suddiviso nelle seguenti parti:

10.1.1. Markup standard

Vari markup standard sono definiti in `std-markup`. I seguenti tag contestuali prendono tutti un solo argomento. La maggior parte può essere trovata nel menu Testo → Content tag.

strong. Indica una regione **importante** del testo. Si può attivare questo tag attraverso Testo → Content tag → Enfatizzato.

em. Enfatizza una regione del testo come in "la cosa *vera*". Questo tag corrisponde alla voce di menu Testo → Content tag → Enfatizza.

dfn. Per le definizioni come "uno *gnu* è una bestia con le corna". Questo tag corrisponde a Testo → Content tag → Definizione.

samp. Una sequenza di caratteri letterali, come i caratteri ae nella legatura æ. Si può attivare questo tag attraverso Testo → Content tag → Esempio.

name. Il nome di una cosa particolare o un concetto come il sistema Linux. Questo ta si ottiene usando Testo → Content tag → Nome.

person. Il nome di una persona come Joris. Questo tag corrisponde a Testo → Content tag → Persona.

cite*. Una citazione bibliografica come un libro o una rivista. Per esempio, *Moby Dick* di Melville. Questo tag, che si ottieneusando Testo → Content tag → Citazione, non deve essere confuso con cite. Quest'ultimo tag viene usato anche per le citazioni, ma dove l'argomento si riferisce a una voce in un database con riferimenti bibliografici.

abbr. Un'abbreviazione. Esempio: io lavoro al C.N.R.S. Un'abbreviazione viene creata usando Testo → Content tag → Abbreviazione o la scorciatoia da tastiera `A-a`.

acronym. Un acronimo è un'abbreviazione formata dalla prima lettera di ciascuna parola in un nome o una frase, come HTML o IBM. In particolare, le lettere non sono separate da punti. Si può inserire un acronimo usando Testo → Content tag → Acronimo.

verbatim. Testo verbatim come l'output di un programma informatico. Esempio: il programma restituisce `hello`. Si può inserire del testo verbatim attraverso Testo → Content tag → Testo semplice. Il tag può anche essere usato come un ambiente per un testo con molti paragrafi.

kbd. Testo che deve essere inserito da tastiera. Esempio: per favore premere `return`. Questo tag corrisponde alla voce di menu Testo → Content tag → Tastiera.

code*. Codice di un programma come in "`cout << 1+1;` yields 2". Questo si attiva usando Testo → Content tag → Codice. Per pezzi di codice più lunghi, si dovrebbe usare l'ambiente code.

var. Variabili in un programma informatico come in `cp` *`src-file dest-file`*. Questo tag corrisponde alla voce di menu Testo → Content tag → Variabile.

math. Questo è un tag che verrà usato in futuro per inserire parti di carattere matematico nel testo regolare. Esempio: la formula $\sin^2 x + \cos^2 x = 1$ è ben nota.

op. Questo tag può essere utilizzato nelle parti matematiche per specificare che un operatore deve essere considerato per se stesso, senza alcun argomento. Esempio: l'operazione $+$ è una funzione da $\mathbb{R}^2$ in $\mathbb{R}$. Questo tag potrebbe diventare obsoleto.

tt. Questo è un tag fisico per per la fase di composizione tipografica. Esso è utilizzato per assicurare la compatibilità con il formato HTML, ma non ne raccomandiamo l'uso.

I seguenti sono ambienti standard:

verbatim. Descritto sopra.

code. Analogo a code*, ma per parti di codice con molte righe.

quote. Ambiente per brevi (un paragrafo) citazioni.

quotation. Ambiente per citazioni lunghe (molti paragrafi).

verse. Ambiente per la poesia.

center. Questo è un tag fisico per centrare una o più righe di testo. E' utilizzato per la compatibilità con il formato HTML, ma non ne raccomandiamo l'uso.

Alcuni ambienti standard per le tabelle:

tabular*. Tabelle centrate.

block. Tabelle allineate a sinistra con un bordo standard di larghezza `1ln`.

block*. Tabelle centrate con un bordo di larghezza standard di `1ln`.

I seguenti tag non hanno alcun argomento:

TeXmacs. Il logo $\mathrm{T_EX_{MACS}}$.

TeX. Il logo $\mathrm{T_EX}$.

LaTeX. Il logo LaTeX.

hflush. Utilizzato dagli sviluppatori per il flushing a destra nella definizione di ambienti.

hrule. Una riga orizzontale come quella qui sotto:

I tag seguenti ammettono uno o più argomenti:

overline. Per $\overline{\text{disegnare una linea sopra}}$ il testo, che può estendersi su più righe.

underline. Per <u>sottolineare</u> il testo, che può estendersi su più righe.

fold. Macro con due argomenti. Il primo argomento viene visualizzato e il secondo è ignorato: la macro corrisponde alla presentazione piegata di un pezzo del contenuto associato a un breve titolo o un abstract. Il secondo argomento si può rendere visibile usando Inserisci → Switch → Unfold.

unfold. Macro con due argomenti `x` e `y`, che genera la presentazione non piegata di un pezzo di contenuto `y` associato a un breve titolo o abstract `x`. Il secondo argomento può essere reso invisibile usando Inserisci → Switch → Arrotola.

switch. Macro con due argomenti `x` e `y`, dove `y` è un insieme di possibili rappresentazioni dello switch e `x` la rappresentazione attuale. I tasti funzione F9, F10, F11 e F12 possono essere usati per lo switch tra rappresentazioni diverse.

phantom. Funzione con un argomento `x`. Questo tag permette di ottenere tanto spazio quanto ne occuperebbe l'argomento `x`, ma `x` non viene visualizzato. Per esempio, il testo "phantom" come argomento di phantom produce "   ".

set-header. Funzione con un argomento per modificare in modo permanente l'intestazione. Si noti che alcuni tag nel file di stile, come i tag di sezione, possono sovrascrivere queste modifiche manuali.

set-footer. Funzione con un argomento per modificare in maniera permanente il piè di pagina.

10.1.2. Simboli standard

Il d.t.d. `std-symbol` definisce i simboli speciali ¢, ¤, ¥, ©, ℗, ®, °, ², ³, ¹, µ, ¶, ¼, ½, ¾, € e ™. Non appena il supporto ai font sarà ulteriormente migliorato, questo d.t.d. diventerà obsoleto.

10.1.3. Markup matematici standard

I markup matematici standard sono definiti in `std-math`.

binom. Per i coefficienti binomiali $\binom{n}{m}$.

choose. Nome alternativo per binom (obsoleto).

shrink-inline. Macro che permette di passare alla dimensione del testo quando non si è nello stile di visualizzazione. Questa macro viene usata principalmente dagli sviluppatori. Per esempio, la macro binom la utilizza.

I seguenti sono ambienti matematici standard per le tabelle:

matrix. Per le matrici $M = \begin{pmatrix} 1 & 2 \\ 3 & 4 \end{pmatrix}$.

det. Per i determinanti $\Delta = \begin{vmatrix} 1 & 2 \\ 3 & 4 \end{vmatrix}$.

choice. Per le liste di scelta $|x| = \begin{cases} -x, & \text{if } x \leqslant 0 \\ x, & \text{if } x \geqslant 0 \end{cases}$

10.1.4. Liste standard

Le liste standard in $\TeX_{\text{MACS}}$ sono definite in `std-list`. Le liste non numerate sono:

itemize. Il tag prima di ciascun elemento dipende dalla profondità di annidamento

itemize-minus. Usa $-$ per il tag.

itemize-dot. Usa $\bullet$ per il tag.

itemize-arrow. Usa $\rightarrow$ per il tag.

Le liste numerate corrispondono ai seguenti ambienti:

enumerate. Il tipo di numero prima di ciascun elemento dipende dalla profondità di annidamento.

enumerate-numeric. Numera gli elementi con 1, 2, 3, ecc.

enumerate-roman. Numera gli elementi con i, ii, iii, ecc.

enumerate-Roman. Numera gli elementi conI, II, III, ecc.

enumerate-alpha. Numera gli elementi con a), b), c), ecc.

enumerate-Alpha. Numera gli elementi con A, B, C, ecc.

I seguenti ambienti possono essere utilizzati per le liste descrittive.

description. L'ambiente per le liste descrittive di default (solitamente description-compact).

description-compact. Allinea a sinistra gli elementi e pone le loro descrizioni immediatamente dopo di essi.

description-dash. Analogo a description-compact, ma usa — per separare ciascun elemento dalla relativa descrizione.

description-align. Allinea a sinistra le descrizioni, mentre allinea a destra gli elementi.

description-long. Pone gli elementi e le loro descrizioni su linee distinte.

I nuovi elementi in una lista sono indicati con il item, o con il tag unario item* nel caso di descrizioni. Gli sviluppatori troveranno qualche macro addizionale, ma instabile, in `std-list` per definire altre strutture di liste.

10.1.5. Generazione automatica del contenuto

Il d.t.d. `std-automatic` consente la generazione automatica di contenuti ausiliari come indici e bibliografie, e inoltre la presentazione dei contenuti stessi. I tag seguenti vengono utilizzati per le bibliografie:

cite. Una funzione con un numero arbitrario di argomenti. Ciascun argomento è una citazione corrispondente ad un elemento in un file BiB-TEX. Le citazioni sono visualizzate nella stessa maniera in cui sono riportate nella bibliografia e forniscono anche degli hyperlink ai riferimenti corrispondenti. Se non si genera la bibliografia, le citazioni appaiono come punti di domanda.

nocite*. Analogo a cite, ma le citazioni non sono visualizzate nel testo principale.

bibitem*. Funzione che specifica come visualizzare un elemento nella bibliografia.

I tag seguenti sono utilizzati per compilare gli indici:

toc-main-1. Funzione con un argomento per creare voci primordiali nell'indice. Questa funzione può, per esempio, essere usata quando un libro è costituito di molte parti.

toc-main-2. Funzione con un argomento per creare una voce principale nell'indice. Generalmente si usa questa funzione per i capitoli.

toc-normal-1. Funzione con un argomento per creare una voce normale nell'indice. Questa funzione è usata spesso per le sezioni.

toc-normal-2. Analogo a toc-normal-2 per le voci meno importanti come le sottosezioni.

toc-normal-3. Analogo a toc-normal-3 per le voci assai meno importanti come sotto-sottosezioni.

toc-small-1. Usata per le voci non molto importanti come i paragrafi (può essere ignorata).

toc-small-2. Usata per le voci assai meno importanti come i sottoparagrafi

toc-dots. Il separatore tra una voce nell'indice e il numero di pagina corrispondente. Per default, usiamo i punti orizzontali.

I tag seguenti sono utilizzati per gli indici analitici:

index. Funzione con un argomento x, che inserisce x nell'indice come una voce principale.

subindex. Funzione con due argomenti x e y, che inserisce y nell'indice come sottovoce di x.

subsubindex. Funzione con tre argomenti *x*, *y* e *z*, che inserisce *z* nell'indice come sottovoce di *y*, la quale è a sua volta sottovoce di *x*.

index-complex. Funzione con quattro argomenti *key*, *how*, *range*, *entry*, che è documentata nella sezione riguardante la generazione degli indici.

index-line. Questa funzione prende un argomento *key*, che indica come ordinare le voci, e la voce attuale *entry*. Non viene generato alcun numero di pagina.

index-1. Macro con una voce dell'indice e un numero di pagina, che viene usata per inserire nell'indice una voce principale.

index-1*. Analogo a index-1, ma senza il numero di pagina.

index-n. (con n tra 1 e 5): macro con una voce dell'indice e un numero di pagina, che è utilizzata per inserire nell'indice una voce di livello n.

index-n*. Analogo a index-n, ma senza il numero di pagina.

index-dots. Macro che produce i punti tra una voce dell'indice e il numero (o i numeri) di pagina corrispondente.

I tag seguenti sono utilizzati per i glossari:

glossary. Funzione che inserisce il suo unico argomento nel glossario.

glossary-dup. Per creare un numero di pagina addizionale per una voce che è già stata inserita.

glossary-explain. Funzione per inserire una voce del glossario con la relativa spiegazione.

glossary-line. Per inserire una voce del glossario snza alcun numero di pagina.

glossary-1. Macro per inserire una voce del glossario e il corrispondente numero di pagina.

glossary-2. Macro inserire una voce del glossario, la sua spiegazione e il suo numero di pagina.

glossary-dots. Macro che produce i punti tra una voce del glossario e i corrispondenti numeri di pagina.

10.1.6. Markup speciali per i programmi

Il d.t.d. `program` fornisce dei markup per l'impaginazione dei programmi informatici. Tuttavia, questi tag sono da considerarsi molto instabili, e prevediamo che verranno sostituiti con un insieme di tag più dettagliati:

algorithm. Macro con due argomenti: il nome dell'algoritmo e l'algoritmo stesso, insieme con eventuali specificazioni.

body. Il corpo effettivo dell'algoritmo.

indent. Per indentare parte di un algoritmo.

10.1.7. Markup speciali per le sessioni

Il d.t.d. session fornisce i seguenti ambienti per le sessioni di computer algebra:

session. Macro con un argomento: il corpo della sessione.

input. Macro con due argomenti: un prompt e i dati in entrata (input) stessi.

output. Macro con il corpo dei dati in uscita (output) come argomento.

Infatti, questi ambienti sono basati sugli ambienti della forma *lan*-session, *lan*-input e *lan*-output per ogni lingua *lan*.

Se gli ambienti specifici per la lingua non esistono, allora si utilizzano al loro posto generic-session, generic-input e generic-output. Si raccomanda di basare gli ambienti specifici per la lingua sugli ambienti generici, che possono avere diverse implementazioni a seconda dello stile (per esempio, il pacchetto varsession). A questo scopo, forniamo anche l'ambiente generic-output*, che è simile a generic-output, tranne per il fatto che i margini rimangono inalterati.

10.2. Ambienti standard in un testo

Il d.t.d. env contiene gli ambienti standard che sono disponibili nella maggior parte degli stili. Esso è suddiviso nelle seguenti parti:

10.2.1. Definire nuovi ambienti

env-manage contiene dei markup di alto livello che possono essere utilizzati dall'utente per definire nuovi ambienti per i teoremi, le note, gli esercizie e le figure:

new-theorem. Definisce un ambiente di tipo teorema. Si deve specificare un nome per l'ambiente (come "esperimento") e il testo corrispondente (come "Esperimento").

new-remark. Analogo a new-theorem, ma per le note.

new-exercise. Analogo a new-theorem, ma per gli esercizi.

new-figure. Analogo a new-theorem, ma per le figure (per le coppie: grande e piccola).

Il d.t.d. contiene anche dei murkup di basso livello per le definizioni attuali degli ambienti. Infatti, la definizione di nuovi teoremi avviene in due passi. Al primo passo, il tag new-theorem è utilizzato per specificare quale ambiente di tipo teorema dovrà essere definito. Al secondo passo (proprio prima che il documento dell'utente venga processato) gli ambienti di tipo teorema vengono effettivamente definiti. Questo meccanismo rende possibile personalizzare gli ambienti in pacchetti che sono processati tra i due passi. Per esempio, la numerazione dei teoremi viene personalizzata in questo modo.

AVVISO 10.1. Attualmente, si dovrebbe usare new-theorem e tag simili in un file di stile personale o in un pacchetto. Se si utilizza new-theorem direttamente in un documento, allora la numerazione può risultare scorretta a causa dello schema a due passi spiegato sopra. Questo inconveniente scomparirà appena sarà possibile specificare dei preamboli corretti per i documenti TeXmacs.

10.2.2. Ambienti matematici

Il d.t.d. `env-math` specifica quali ambienti matematici possono essere usati all'interno della modalità testo. In altre parole, gli ambienti dovrebbero essere utilizzati all'interno della modalità testo, ma i loro corpi contengono formule matematiche o tabelle di formule matematiche

equation. Equazione numerata.

equation*. Equazione non numerata.

eqnarray. Lista di equazioni numerate (non ancora attiva).

eqnarray*. Lista di equazioni non numerate.

Nell'ambiente eqnarray*, si può utilizzare il tag eq-number per numerare l'equazione.

AVVISO 10.2. La numerazione delle equazioni nelle tabelle non è ancora come dovrebbe essere. In particulare, attualmente il tag eqnarray è equivalente a eqnarray*. Quando il tag eqnarray sarà implementato correttamente, sarà disponibile anche un tag no-number per sopprimere il numero di un'equazione, e un pacchetto di stile per porre la numerazione a sinistra delle equazioni.

AVVISO 10.3. Non è ancora disponibile l'opzione per porre la numerazione a sinistra delle equazioni. Tuttavia, si può utilizzare il tag manuale leq-number per fare ciò. Si ha anche a disposizione un next-number che visualizza direttamente il numero e aumenta il contatore dell'equazione.

AVVISO 10.4. Noi non incoraggiamo l'uso degli ambienti AMS-TeX `align`, `gather` e `split`. Comunque, essi sono disponibili sotto i nomi align, gather, eqsplit insieme con le loro varianti align*, gather* e eqsplit*. In futuro, prevediamo di implementare ambienti più potenti.

10.2.3. Ambienti di tipo teorema

Il d.t.d. `env-theorem` fornisce i tag per la disposizione tipografica degli ambienti di tipo teorema. I tag più importanti sono

render-theorem. Macro per visualizzare gli ambienti di tipo teorema. Il primo argomento specifica il nome del teorema, come "Teorema 1.2" e il secondo argomento contiene il corpo del teorema. Questo ambiente e usato per gli ambienti definiti con new-theorem.

render-remark. Analogo a render-theorem, ma per gli ambienti di tipo nota.

render-exercise. Analogo render-theorem, ma per gli ambienti di tipo esercizio.

render-proof. Analogo a render-theorem, ma per le dimostrazioni. Questo ambiente è usato soprattutto per personalizzare il nome di una dimostrazione, come in "Fine della dimostrazione del teorema 1.2".

dueto. Un ambiente che può essere usato per specificare gli autori di un teorema.

corollary*. Per corollari non numerati. Questo ambiente è basato su render-theorem.

proof. Per le dimostrazioni dei teoremi. Questo ambiente è basato su render-proof.

I tag seguenti possono essere utilizzati per ulteriori personalizzazioni degli ambienti.

theorem-name. Una macro che controlla l'aspetto dei nomi degli ambienti di tipo teorema *e* nota. La maggior parte degli stili utilizza il grassetto o lettere maiuscole piccole.

exercise-name. Analogo a theorem-name, ma per gli esercizi.

theorem-sep. Il separatore tra il nome di un ambiente di tipo teorema o di tipo nota e il il suo corpo principale. Per default, questo è un punto seguito da uno spazio.

exercise-sep. Analogo a theorem-sep, ma per gli esercizi.

10.2.4. Ambienti per oggetti fluttuanti

Il d.t.d. `env-float` fornisce i tag per gli oggetti fluttuanti. Il tag seguente è l'unico di alto livello:

footnote. crea una nota a piè di pagina.

I seguenti tag di basso livello possono essere utilizzati per definire ambienti di alto livello per le figure e le tabelle: big-figure, small-figure, big-table e small-table:

render-small-figure. Una macro per visualizzare una figura piccola. Gli argomenti sono un nome breve (come "figura" o "tabella") per la lista di figure, il suo nome reale (come "Figura 2.3" o "Tabella 5"), la figura stessa e una didascalia.

render-big-figure. Una variante di render-small-figure per visualizzare una figura grande.

I tag seguenti possono essere utilizzati per personalizzare l'aspetto del testo attorno alle figure, alle tabelle e alle note a piè di pagina:

figure-name. Una macro che controlla l'aspetto del testo "Figura". Per default, viene utilizzato il grassetto.

figure-sep. Il separatore tra la figura , il suo numero e la didascalia. Per default, questo è un punto seguito da uno spazio.

footnote-sep. Il separatore tra il numero della nota e il testo. Per default, questo è un punto seguito da uno spazio.

10.2.5. Ambienti predefiniti

Il d.t.d. `env-default` contiene gli ambienti testuali di default. Essi sono suddivisi nei gruppi seguenti:

Varianti dei teoremi. Il corpo degli ambienti di tipo teorema solitamente viene enfatizzato. Per default, sono disponibili attraverso il menu Testo → Ambiente, i seguenti ambienti: theorem, proposition, lemma, corollary, axiom, definition, notation, conjecture.

Varianti delle note. Le seguenti sono disponibili attraverso il menu Testo → Ambiente: remark, example, note, warning, convention.

Varianti degli esercizi. Due di questi ambienti sono noti per default e sono disponibili attraverso il menu Testo → Ambiente: exercise e problem.

Variants delle figure. Questi ambienti esistono sempre in coppia: i grandi e i piccoli. Per default, sono disponibili big-figure, small-figure, big-table e small-table. Si può accedere ad essi attraverso i menu Inserisci → Immagine e Inserisci → Tabella.

Altri ambienti utili. Sono disponibili keywords e AMS-class (per la classificazione A.M.S.). Questi ambienti dovrebbero essere inseriti in un abstract.

10.3. INTESTAZIONI E NOTE A PIÈ DI PAGINA

10.3.1. Intestazioni standard

Il d.t.d. `header` fornisce i tag per personalizzare le intestazioni e le note a piè di pagina. La personalizzazione è basata sull'idea che possiamo specificare una *testo di pagina* per ogni pagina. Questo testo di pagina può essere, per esempio, un titolo corrente o il nome della sezione attuale. Il testo di pagina può dipendere dalla parità di una pagina e apparire in modo differente per pagine speciali come la pagina iniziale di un nuovo capitolo. I tag seguenti controllano l'impaginazione fisica dei tipi diversi di pagina:

start-page. Questo tag, con testo di pagina come suo unico argomento, specifica l'impaginazione della prima pagina di un nuovo capitolo o di una sezione.

odd-page-text. Analogo a start-page, ma per l'impaginazione delle pagine dispari ordinarie.

even-page-text. Analogo a start-page, ma per l'impaginazione delle pagine pari ordinarie.

I tag seguenti controllano le azioni logiche relative alle intestazioni da eseguire quando si specfica un titolo, un autore, o quando si inizia una nuova sezione.

header-title. Tag con un "argomento titolo" che viene utilizzato per specificare il titolo del documento.

header-author. Tag con un "argomento autore" che viene utilizzato per specificare l'autore del documento.

header-primary. Tag con un "argomento nome della sezione" che viene utilizzato all'inizio di ciascuna sezione di base (cioè chapter per lo stile libro, o section per lo stile articolo).

header-secondary. Tag con un "argomento nome della sezione" che è utilizzato all'inizio sezione secondaria (cioè section per lo stile libro, o subsection per lo stile articolo).

10.3.2. Titoli standard

Il d.t.d. `header-title` fornisce i tag per i titoli. I seguenti tag di alto livello possono essere usati solamente quando sono inseriti in un tag make-title:

title. Specifica un titolo per il documento.

author. Specifica uno o più autori per il documento.

address. Specifica l'indirizzo dell'autore.

address-block. Specifica un indirizzo per un autore (nel caso di indirizzi multipli).

title-email. Specifica l'indirizzo di posta elettronica dell'autore.

title-date. Specifica la data in cui è stato scritto l'articolo.

title e author usano i tag header-title e header-author per specificare il titolo e l'intestazione correnti. Questi si possono sovrascrivere riutilizzando rispettivamente header-title e header-author. I tag descritti qui sopra dipendono per la loro disposizione fisica anche dai seguenti tag di basso livello:

title*. Macro con un argomento che specifica il layout fisico dei titoli.

author*. Macro con un argomento che specifica il layout degli autori.

address*. Macro con un argomento che specifica il layout fisico degli indirizzi.

title-email*. Macro con un argomento che specifica il layout fisico degli indirizzi email.

title-date*. Macro con un argomento che specifica il layout fisico delle date di creazione.

Il d.t.d. `header-title` definisce anche il tag abstract per il sommario dei documenti.

10.4. Sezioni in stile LaTeX

Il d.t.d. `section-latex` fornisce i tag standard per le sezioni, che sono uguali a quelle del LaTeX. La maggior parte dei tag di sezione prendono un solo argomento: il nome della sezione. In futuro, verranno creati dei tag alternativi con due argomenti, che consentiranno di vedere il corpo di una sezione come parte della struttura. I tag seguenti solitamente generano sezioni numerate, che sono riportate nell'indice:

chapter. Macro per produrre il titolo di un capitolo che può essere numerato.

section. Macro per produrre il titolo di una sezione che può essere numerato.

subsection. Macro per produrre il titolo di una sottosezione che può essere numerata.

subsubsection. Macro per produrre il titolo di una sotto-sottosezione che può essere numerata.

paragraph. Macro per produrre il titolo di un paragrafo che può essere numerato.

subparagraph. Macro per produrre il titolo di un sottoparagrafo che può essere numerato.

I tag chapter*, section*, subsection*, subsubsection*, paragraph* e subparagraph* possono essere utilizzati per le varianti non numerate dei tag appena descritti, che non vengono riportate nell'indice. Il d.t.d. `section-latex` fornisce anche i tag seguenti:

chapter.** Macro con due argomenti: un tipo speciale di capitolo (come "Epilogo") e il nome del capitolo.

appendix. Una variante di chapter o section per produrre le appendici.

section-sep. Macro per personalizzare il separatore tra il numero di una sezione e il suo titolo. Per default, usiamo due spazi.

10.5. Markup per la numerazione automatica

10.5.1. Numerazione degli ambienti

Il d.t.d. `number-env` fornisce i tag di basso livello per la numerazione degli ambienti standard. Uno dei tag più importanti è init-stdenv che viene utilizzato per azzerare tutti i contatori degli ambienti. Solitamente questo viene fatto all'inizio di ciascun capitolo o sezione, o una volta per l'intero documento.

Il d.t.d. esporta anche i tag di livello molto basso newliststdenv, newlistfigure e newliststdenv-counter, che controllano la numerazione in collaborazione con `env-manage`. I pacchetti `number-us` e `number-europe` sono a disposizione per la numerazione in stile americano e europeo rispettivamente.

10.5.2. Numerazione delle sezioni

Il d.t.d. `number-section` fornisce i tag di basso livello per numerare le sezioni. Esso definisce i tag reset-chapter, reset-section, reset-subsection e reset-subsubsection, per azzerare i contatori appropriati all'inizio di capitoli, sezioni, sottosezioni o sotto-sottosezioni. Esso definisce anche un reset-top per azzerare tutti i contatori di alto livello.

Il d.t.d. definisce anche i tag the-chapter, the-section, the-subsection e the-subsubsection per stampare il nome del corrente capitolo, sezione, sottosezione o sotto-sottosezione. Infine, esso mette a disposizione il tag the-prefix che produce il prefisso quando si numerano gli ambienti (equazioni, teoremi, tabelle, ecc.).

Capitolo 11

Personalizzare TeXmacs

Una della maggiori caratteristiche di TeXmacs è che può essere profondamente personalizzato. Innanzitutto, gli aspetti più importanti del programma possono essere configurati in Modifica → Preferenze. Quasi tutte le parti di TeXmacs possono essere interamente adattate o riprogrammate usando il linguaggio di estensione Guile/Scheme. Alcuni semplici esempi si possono trovare nelle sezioni seguenti:

11.1. Introduzione al linguaggio di estensione Guile

Come Emacs, TeXmacs supporta un linguaggio di estensione di tipo Lisp, ovvero il dialetto Guile Scheme del progetto Gnome. Per la documentazione su Guile Scheme, facciamo riferimento a

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

Scheme ha il vantaggio che può essere esteso con tipi e routine esterne del C e C++. Nel nostro caso, abbiamo esteso Scheme con routine che si possono utilizzare per creare dei menu e delle combinazioni di tasti personalizzati, e persino per scrivere delle proprie estensioni per TeXmacs.

Se si sono scaricati i file sorgenti di TeXmacs, potrebbe essere interessante dare un'occhiata ai file

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

Questi tre file "glue" contengono le routine C++ che sono visibili in Scheme. Qui di seguito, discuteremo alcune delle routine più importanti. In futuro, scriveremo una guida di riferimento più completa sull'argomento. Si possono comunque guardare i file scheme `.scm` nella directory `$TEXMACS_PATH/progs`.

11.2. Scrivere file di inizializzazione personalizzati

Quando si apre TeXmacs viene eseguito il file

`$TEXMACS_PATH/progs/init-texmacs.scm`

e, se esiste, viene eseguito anche il file dell'utente

`$TEXMACS_HOME_PATH/progs/my-init-texmacs.scm`

Per default, il path `$TEXMACS_HOME_PATH` coincide con `.TeXmacs`. Analogamente, ogni volta che si crea un nuovo buffer, viene eseguito il file

`$TEXMACS_PATH/progs/init-buffer.scm`

e, se esiste, viene eseguito anche

```
$TEXMACS_HOME_PATH/progs/my-init-buffer.scm
```

11.3. Creare menu dinamici personalizzati

Si può definire (o modificare) un (una parte di un) menu con il nome `name` usando

```
(menu-bind name . prog)
```

e aggiungere nuove voci ad un (ad una parte di un) menu esistente con il nome `name` usando

```
(menu-extend name . prog)
```

Qui `prog` è un programma che rappresenta le voci del menu. In particolare, si possono guardare i file nella directory

```
$TEXMACS_PATH/progs/menu
```

per vedere come sono definiti i menu standard di $\mathrm{T_EX_{MACS}}$.

Più precisamente, il programma `prog` in `menu-set` o `menu-append` è una lista di voci che possono avere una delle seguenti forme:

```
(=> "pulldown menu name" menu-definition)
(-> "pullright menu name" menu-definition)
("entry" action)
---
(if condition menu-definition)
(link variable)
```

I costruttori `=>` e `->` sono utilizzati per creare dei menu che si aprono verso il basso o verso destra e `menu-definition` deve contenere un programma che crea il submenu. Il costruttore `("entry" action)` crea una voce normale, dove `action` sarà compilata ed eseguita quando si clicca su `entry`. Gli elementi del menu possono essere separati usando `---`. Il costruttore `if` è usato per inserire elementi del menu solamente se una certa `condition` è soddisfatta (per esempio, se ci si trova in modalità matematica).

Infine, se si dichiara un menu `name`, allora si può utilizzare questo menu indirettamente usando il costruttore `link`. Questo modo indiretto di dichiarare dei sottomenu ha due vantaggi

- Un sottomenu "indiretto" può essere collegato a tanti menu quanti si vuole.
- Nuovi elementi possono essere aggiunti a dei sottomenu "indirecti" *a posteriori* usando `menu-append`.

I menu principali di $\mathrm{T_EX_{MACS}}$ sono `texmacs-menu`, `texmacs-popup-menu`, `texmacs-main-icons`, `texmacs-context-icons` e `texmacs-extra-icons`. Altri menu standard indiretti sono `file-menu`, `edit-menu`, `insert-menu`, `text-menu`, `paragraph-menu`, `document-menu`, `options-menu` and `help-menu`.

11.4. Creare scorciatoie da tastiera personalizzate

Le corrispondenze dei tasti sono specificate usando il comando

```
(kbd-map predicate . keymaps)
```

La parte `predicate` specifica sotto quali condizioni le corrispondenze dei tasti sono valide. Esempi di predicati sono `always?`, `in-math?` e `in-french?`, ma l'utente può definire dei predicati personalizzati. Ogni elemento in `keymaps` può avere una delle seguenti forme:

```
(key-combination action_1 ... action_n)
(key-combination result)
(key-combination result help-message)
```

Nel primo caso, le `action_i` sono i comandi Scheme associati alla stringa `key-combination`. Nel secondo e nel terzo caso, `result` è una stringa che deve essere inserita nel testo dopo che la combinazione di tasti `key-combination` è stata eseguita. Può essere visualizzato un ulteriore messaggio `help-message` quando viene eseguita `key-combination`.

11.5. Altri file interessanti

Potrebbe essere interessante guardare i file seguenti:

- `$TEXMACS_PATH/fonts/enc` contiene le codifiche di diversi font TeX.
- `$TEXMACS_PATH/fonts/virtual` contiene le definizioni di caratteri virtuali.
- `$TEXMACS_PATH/langs/natural/dic` contiene i dizionari usati da TeXmacs.
- `$TEXMACS_PATH/langs/natural/hyphen` contiene i modelli di sillabazione per varie lingue.
- `$TEXMACS_PATH/progs/fonts` contiene i programmi Scheme per definire i font.

Compatibilità con altri formati

TeXmacs è pienamente compatibile con il formato postscript, che viene utilizzato per stampare i documenti. TeXmacs inoltre fornisce dei convertitori da e in LaTeX e un filtro in input per il formato Html.

11.6. Compatibilità con LaTeX

Nonostante TeXmacs non sia stato creato per essere pienamente compatibile con LaTeX, è possibile convertire documenti da TeXmacs a LaTeX e *viceversa*, sebbene il risultato potrebbe non essere sempre perfetto. Inoltre, le conversioni da TeXmacs a LaTeX generalmente daranno risultati migliori rispetto all'inverso. In particolare, TeXmacs può essere usato ragionevolmente bene per scrivere articoli che devono poi essere convertiti in LaTeX. In questo capitolo, descriveremo più precisamente i meccanismi di conversione, che aiuteranno ad ottenere il risultato più soddisfacente possibile.

11.6.1. Conversione da TEXMACS a LATEX

La situazione più comune è che si voglia convertire un articolo da TEXMACS a LATEX, per poterlo spedire a qualche rivista. Dato un file TEXMACS `name.tm`, lo si può convertire in un file LATEX `name.tex` mediante File → Esporta → Latex. Come primo passo, si può provare ad eseguire LATEX su `name.tex` e vedere se il risultato che si ottiene è già soddisfacente. Se ciò accade, si può spedire alla rivista il file `name.tex` insieme al file di stile `TeXmacs.sty`, che si trova nella `$TEXMACS_PATH/misc/latex`.

Spesso, la rivista utilizza un proprio file di stile, detto `journal.sty`. In tal caso, si dovrebbe copiare anche il file

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

in

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

e usare `journal` come proprio stile di documento in Documento → Stile → Altri stili. Si può anche modificare `journal.ts`, in modo che l'impaginazione dell'articolo diventi più simile allo stile della rivista. In alcuni casi, si deve creare anche un'altra copia di `TeXmacs.sty`, e modificare qualche ambiente per ottenere la compatibilità con il file di stile della rivista `journal.sty`.

Se il primo tentativo di convertire un documento in LATEX non ha prodotto un risultato soddisfacente, generalmente si osserverà che solamente piccole parti di testo non sono state convertite correttamente. Questo può essere dovuto a tre cause principali:

- Nel documento vengono utilizzate caratteristiche specifiche di TEXMACS.
- E' stata utilizzata una caratteristica diTEXMACS che non è ancora stata implementata nell'algoritmo di conversione.
- E' presente un bug nell'algoritmo di conversione.

Questi aspetti verranno discussi più dettagliatamente nella prossima sezione.

In caso di problemi, una stategia naive sarebbe di correggere il file LATEX prodotto e spedirlo alla rivista. Comunque questa strategia ha lo svantaggio che si devono ripetere queste correzioni ogni volta che si converte il file TEXMACS file `name.tm`, dopo avere compiuto qualche altra modifica. Una strategia migliore è di usare Inserisci → Specifico → Latex e Inserisci → Specifico → Texmacs per scrivere del testo visibile solamente nel file convertito e originale rispettivamente.

Per esempio, si assuma che la parola "blauwbilgorgel" sia sillabata correttamente nel sorgente TEXMACS, ma non nella conversione LATEX. Allora si può procedere nel modo seguente:

1. Selezionare "blauwbilgorgel".
2. Cliccare su Inserisci → Specifico → Texmacs per rendere il testo "blauwbilgorgel" come specifico di TEXMACS.
3. Cliccare su Inserisci → Specifico → Latex.
4. Scrivere il codice latex `blauw\-bil\-gor\-gel` con la sillabazione corretta.

5. Premere `invio` per attivare il testo specifico di LaTeX.

In maniera simile, si possono inserire interruzioni di riga, interruzioni di pagina, spaziature verticali, modifiche dei parametri di stile, ecc., specifiche di LaTeX.

11.6.2. Possibili problemi di conversione

11.6.2.1. Caratteristiche specifiche di TeXmacs

Alcune primitive di composizione tipografica di TeXmacs non possiedono un analogo in LaTeX, e l'algoritmo di conversione li trasformerà semplicemente in spazi bianchi. Alcune caratteristiche principali che sono specifiche di TeXmacs sono le seguenti:

- Primi a sinistra.
- Separatori grandi tra le parentesi grandi.
- Mosaici.
- Alberi.
- Macro personalizzate complesse.
- Spazi verticali "prima" e "dopo".
- Flag di indentazione "prima" e "dopo".

Si dovrebbe cercare di evitare di usare queste caratteristiche specifiche di TeXmacs, se si prevede di convertire il proprio documento in LaTeX. Comunque, in futuro, il programma di conversione dovrebbe generare per default un file encapsulated postscript di una conversione più intellegibile.

11.6.2.2. Conversioni non ancora implementate

Nonostante si cerchi di mantenere l'algoritmo di conversione il più completo possibile per soddisfare le necessità degli utenti, alcune cose non sono state ancora implementate. Alcuni esempi sono

- Font non standard.
- Conversioni di tabelle.
- Parametri di stile.

Qualunque suggerimento riguardante le estensioni desiderabili dell'algoritmo di conversione dovrebbe essere segnalato all'indirizzo

`contact@texmacs.org`

Si cercherà di incorporare le estensioni segnalate il più in fretta possibile. Potrebbe servire del tempo per implementare correttamente la conversione dei parametri di stile, poiché questi non sono gli stessi in TeXmacs e LaTeX. Inoltre, le differenze di impaginazione tra TeXmacs e LaTeX non possono essere eliminate completamente.

11.6.2.3. Bug nell'algoritmo di conversione

La situazione più spiacevole è quando un documento TeXmacs convertito produce molti errori in fase di compilazione o se il risultato non ha nulla a che fare con l'originale. In tal caso, probabilmente si è scovato un bug dell'algoritmo di conversione (o nell'installazione di LaTeX sul proprio sistema). Per favore, si cerchi di capire qual è la causa del bug in questo caso e segnalarlo inviando un'email all'indirizzo

`TeXmacs@math.u-psud.fr`

11.6.2.4. Work-around

TeXmacs non è stato progettato per essere completamente compatibile con LaTeX. Per quanto riguarda la conversione da LaTeX a TeXmacs, il nostro scopo principale è di *aiutare* l'utente nella conversione in TeXmacs di vecchi documenti. Se non sono stati definiti ambienti strani e non sono stati utilizzati file di stile e comandi insoliti, si dovrebbe poter convertire correttamente i vecchi documenti in formato LaTeX. Altrimenti, suggeriamo di modificare il vecchio documento in maniera tale che possa essere convertito ragionevolmente bene e poi applicare qualche modifica finale al documento risultante.

11.6.3. Conversione da LaTeX a TeXmacs

Lo scopo attuale del programma di conversione da LaTeX a TeXmacs è di *aiutare* nella traduzione in TeXmacs di vecchi documenti. In genere, le conversioni da LaTeX a TeXmacs sono più problematiche delle conversioni contrarie. Tuttavia, se ci si è limitati ad usare i comandi LaTeX più comuni, dovrebbe essere possibile convertire i propri vecchi documenti ragionevolmente bene. Per esempio, tutti i file di aiuto di TeXmacs sono stati scritti in LaTeX per verificare il programma di conversione da LaTeX a TeXmacs.

Si può convertire un documento LaTeX `name.tex` in TeXmacs usando File → Importa → Latex e salvarlo con il nome `name.tm`. Se il proprio documento LaTeX è stato scritto sufficientemente bene, allora il risultato della conversione dovrebbe essere più o meno accettabile, a parte alcuni comandi che non sono stati riconosciuti che compaiono in rosso. Una buona soluzione sarebbe di scrivere un proprio file di stile per i documenti convertiti, basato sullo stile originale, nel quale siano definiti i comandi sconosciuti.

Tuttavia, in alcuni casi meno fortunati, il documento convertito apparirà assai confuso. Ciò è generalmente causato dal fatto che TeX e LaTeX permettono agli utenti di modificare dinamicamente l'analizzatore sintattico, per esempio, utilizzando il comando `\catcode`. In questo caso, il programma di conversione potrebbe confondersi e compiere delle assunzioni errate sulla modalità o sull'ambiente. Come risultato, il testo potrebbe essere convertito come modalità matematica, le parti matematiche come verbatim, e così via. Comunque, nel file sorgente `name.tex` i comandi che confondono il programma di conversione solitamente si localizzano facilmente confrontando la versione in formato LaTeX con la corrispondente conversione in TeXmacs. Dopo qualche ricerca nel file sorgente e dopo aver rimosso qualche parte del codice che crea problemi, il documento dovrebbe poter essere convertito correttamente.

Nel futuro, il programma di conversione verrà esteso con un convertitore di file di stile e qualche caratteristica aggiuntiva per facilitare la traduzione dei comandi definiti dall'utente, che sono definiti in un documento distinto da quello che si desidera convertire.

11.7. Conversione di documenti TeXmacs in Html

Abbiamo cominciato ad implementare la conversione tra i formati HTML e TeXmacs. Per il momento, è possibile solamente importare documenti HTML utilizzando File → Importa → Html. La maggior parte di HTML 2.0 e alcune parti di HTML 3.0 sono attualmente supportate. Comunque, non è ancora possibile la navigazione. Nel futuro, implementeremo Math-ML.

Quando si importano documenti HTML, i file i cui nomi cominciano con `http:` o `ftp:` saranno scaricati dalla rete usando `wget`. Se ci si compila TeXmacs personalmente, si può scaricare `wget` da

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

Nelle distribuzioni binarie, `wget` è già incluso.

Appendice A
Configurazione di TeXmacs

A.1. Introduzione

Prima di iniziare ad utilizzare TeXmacs può essere conveniente configurare il programma utilizzando il menu Modifica → Preferenze, in modo da adattarlo nel modo migliore possibile alle vostre specifiche esigenze. Oltre a questo è importante scegliere un tipo ben definito di "aspetto" in Modifica → Preferenze → Aspetto. Ciò permette, ad esempio, di rendere le abbreviazioni dei comandi da tastiera di TeXmacs simili a quelle utilizzate in altre applicazioni.

Allo stesso tempo TeXmacs gestisce un potente ambiente di comandi abbreviati da tastiera, in modo da ottimizzare l'uso di tasti modificatori come `shift` e `control`. In alcuni sistemi tuttavia questi tasti non sono ben configurati percui è necessario procedere alla loro riconfigurazione.

A.2. Configurazione dei tasti modificatori

TeXmacs utilizza cinque principali modificatori da tastiera: `shift`, `control`, `alternate`, `meta` e `hyper`, che vengono abbreviati rispettivamente con `S-`, `C-`, `A-`, `M-` e `H-`. I tasti `shift` e `control` si trovano su tutte le tastiere; il tasto `alternate` su quasi tutte. Molte tastiere moderne sono dotate anche del tasto `windows` che, in TeXmacs, è equivalente a `meta`.

Prima di riconfigurare la tastiera è consigliabile verificare se ciò sia veramente necessario. Se sulla vostra tastiera ci sono i tasti corrispondenti a `shift`, `control`, `alternate` e `meta`, probabilmente non saranno necessarie modifiche di configurazione. Una possibile eccezione è che desideriate utilizzare il tasto `caps-lock` per scrivere simboli matematici. In questo caso dovrete far corrispondere il tasto `caps-lock` al tasto `hyper`.

Per riconfigurare la tastiera è sufficiente selezionare, nel menu Modifica → Preferenze → Tastiera, il modificatore logico che desiderate far corrispondere ad un dato tasto fisico. Ad esempio, selezionando Tasto Window → Map to M modifier, il tasto `windows` verrà fatto corrispondere al modificatore `meta`. In modo del tutto analogo, selezionando Caps-lock key → Map to H modifier, il tasto `caps-lock` verrà fatto corrispondere al modificatore `hyper`.

Sfortunatamente il sistema X Window permette di effettuare solo riconfigurazioni globali. Per questo, se in TeXmacs viene riconfigurato il tasto `caps-lock`, il nuovo comportamento di questo tasto interesserà anche tutte le altre applicazioni in cui questo stesso tasto viene utilizzato. Perciò è consigliabile riconfigurare solo i tasti che non vengono utilizzati per scopi diversi in altre applicazioni. Ad esempio, il tasto `windows` non viene solitamente utilizzato in molte altre applicazioni, percui la sua riconfigurazione non ha ripercussioni. Alcuni utilizzatori potrebbero decidere di effettuare una opportuna riconfigurazione globale della tastiera. Ciò può essere fatto ricorrendo al comando `xmodmap` per informazioni sul quale si rimanda alle corrispondenti pagine del manuale.

In alcuni casi può accadere che i tasti corrispondenti a `alternate`, `meta` e `hyper`, siano presenti sulla tastiera ma non funzionino nel modo desiderato. Per farli funzionare adeguatamente è sufficiente rimappare i prefissi `A-`, `M-` e `H-` in altri modificatori logici appartenenti al primo gruppo di sottomenu in Modifica → Preferenze → Tastiera.

Ad esempio, per avere compatibilità con Emacs, potreste decidere di scambiare il tasto `meta` o `windows` con il tasto `alternate` senza tuttavia eseguire una modifica globale. Ciò può essere fatto cercando i modificatori corrispondenti a questo tasto; tipicamente avremo `Mod1` al posto di `alternate` e `Mod4` al posto di `meta` o `windows`. Quindi eseguiremo le dovute permutazioni nel menu Modifica → Preferenze → Tastiera, selezionando A modifier → Equivalent for Mod4 e M modifier → Equivalent for Mod1.

A.3. NOTE PER UTILIZZATORI RUSSI E UCRAINI

Per scrivere testi in russo o in ucraino sono disponibili diverse opzioni:

- scegliere Russo come lingua predefinita in Modifica → Preferenze → Lingua → Russo. Se la lingua russa è selezionata come lingua locale, TeXmacs si avvierà automaticamente con i menu in lingua russa;
- selezionare Russo per scrivere un intero documento in lingua russa utilizzando il menu Documento → Lingua → Russo;
- selezionare Russo per scrivere porzioni di un testo in russo all'interno di un testo in altra lingua utilizzando il menu Formato → Lingua → Russo.

Se il vostro server X utilizza l'estensione xkb ed è istruito per eseguire lo switch tra le modalità tastiera Latina e tastiera Russa, non avete bisogno di introdurre modifiche. Semplicemente eseguite lo switch con la modalità tastiera russa e procedete. Tutto il software necessario a questo scopo viene normalmente incluso nelle moderne distribuzioni di GNU/Linux e l'estensione xkb viene attivata di default in `XF86Config`. Nell'estensione xkb i caratteri russi sono scritti su 2 byte e le lettere russe su 0x6??. La tastiera viene configurata attraverso `setxkbmap`. Quando il server X parte viene lanciato questo comando con il file di sistema globale `Xkbmap` (usualmente questo file si trova in `/etc/X11/xinit`) se esiste; e quindi, sempre se esiste, con il file degli utilizzatori `~/.Xkbmap`. Un file `~/.Xkbmap` può avere un aspetto del tipo

```
ru basic grp:shift_toggle
```

Questo significa che la modalità della tastiera può essere modificata attraverso `l-shift r-shift`. Altre scelte diffuse sono `control shift` o `control alternate`, si veda `/usr/X11R6/lib/X11/xkb/` per ulteriori dettagli. Questo è il settaggio migliore per sistemi Linux moderni e nel caso in cui sia prevedibile un uso intensivo della lingua russa.

In vecchi sistemi Linux l'estensione xkb è spesso disabilitata. I caratteri vengono scritti su 1 byte e configurati in `xmodmap`. Quando il server X parte i comandi vengono lanciati attraverso `Xmodmap` (tipicamente incluso in `/etc/X11/xinit`), se esiste; e, sempre se esiste, con il file degli utilizzatori `~/.Xmodmap`. Potete configurare una combinazione di tasti per la modalità di cambio della tastiera e utilizzare un codice in russo a 1 byte (come koi8-r) per lavorare in modalità Russo. Risulta comunque più agevole scaricare il pacchetto `xruskb` e attivarlo all'inizio di una sesione di X digitando

```
xrus jcuken-koi8.
```

Attraverso questo comando viene selezionato il formato jcuken (vedi oltre) e il codice koi8-r per la tastiera in modalità Russo. Se decidete di utilizzare questo settaggio per la tastiera dovete selezionare Options → international keyboard → russian → koi8-r.

Potete anche utilizzare il codice Windows cp1251 al posto di koi8-r anche se questa scelta è rara in ambiente UNIX. Se utilizzate `xrus jcuken-cp1251`, selezionate cp1251 al posto di koi8-r.

Tutti i metodi appena descritti richiedono alcune azioni specifiche per "rendere russa" la tastiera. Ciò non è difficile e per questo potete consultare il Cyrillic-HOWTO o, meglio, una sua versione aggiornata:

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

Oltre a questo precisiamo che i metodi precedenti modificano globalmente tutte le applicazioni che fanno uso di X: editori di testo (emacs, nedit, kedit...), terminali X, TeXmacs ecc....

Se avete bisogno di utilizzare raramente la lingua russa il settaggio della tastiera potrebbe causare più danni che benefici. Per utilizzatori occasionali della lingua russa TeXmacs implementa un metodo che non richiede azioni preliminari. Ovviamente questo metodo funziona solo in TeXmacs e non in altre applicazioni.

Il modo più semplice per scrivere in russo con una tastiera standard americana, senza tuttavia aver apportato modifiche al software, consiste nel selezionare il menu Modifica → Preferenze → Tastiera → Metodo di input cirillico → translit. In questo modo quando si digita una lettera dell'alfabeto latino verrà prodotta la lettera russa ad esso "più simile". Per ottenere alcune lettere russe è necessario ricorrere a combinazioni di due o tre lettere:

Abbreviazione	per	Abbreviazione	per
`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

Tabella A.1. Scrivere in cirillico su una tastiera occidentale.

Se desiderate "сх" e non "ш" dovete digitare `s / h`. Chiaramente la scelta ottimale della mappatura da lettere latine a russe non è unica. Potete verificare la mappatura supportata da TeXmacs e se non vi è gradita sovrascriverla in `~/.TeXmacs/progs/my-init-texmacs.scm`.

Se viene selezionato jcuken al posto di translit, si ha a disposizione la tastiera russa "ufficiale". Questa tastiera ha questo nome in quanto i tasti "qwerty" producono "йцукен". Questo metodo di input risulta più utile se avete a disposizione una tastiera originale russa, che ha a disposizione ulteriori lettere dell'alfabeto russo scritte in rosso su alcuni tasti. Volendo utilizzare jcuken, un effetto simile si può riprodurre incollando opportunamente degli adesivi con lettere dell'alfabeto russo sulla tastiera americana.

Coloro che non dispongono di lettere russe sulla tastiera forse preferiscono l'impaginazione yawertyin cui il tasto “qwerty” produce “яверты”. Anche qui ogni lettera latina viene mappata in una russa "simile", mentre altre lettere russe vengono prodotte con il tasto `shift`-lettera. In TeXmacs yawerty viene implementato in modo un pò diverso in quanto, per convenienza, non vengono ridefiniti i tasti `$`, `£`, `\`. Le corrispondenti lettere russe vengono prodotte utilizzando una combinazione del tipo `shift`-lettera.

Appendice B

A proposito di GNU TeXmacs

B.1. Sommario

GNU TeXmacs	
Sistemi supportati	Principalmente sistemi GNU/Linux
Copyright	© 1998–2002 by Joris van der Hoeven
Licenza	GNU General Public License
Siti web	`http://www.texmacs.org` `http://www.gnu.org/software/texmacs`
Contatti	`contact@texmacs.org`
Indirizzi	Dr. Joris van der Hoeven Dépt. de Mathématiques (Bât. 425) Université Paris-Sud 91405 Orsay Cedex France

Tabella B.1. Sommario delle principali informazioni su GNU TeXmacs.

B.2. La filosofia di TeXmacs

B.2.1. Una breve descrizione di GNU TeXmacs

GNU TeXmacs è un text editor scientifico libero, che si ispira sia a TeX che a GNU Emacs. L'editor permette di scrivere documenti strutturati attraverso una interfaccia amichevole e wysiwyg (acronimo di "what you see is what you get", ossia ciò che vedi è ciò che scrivi) (what-you-see-is-what-you-get). Inoltre l'utilizzatore può crearsi degli stili personalizzati per la redazione dei documenti. Il programma implementa algoritmi per la composizione ad alta qualità ed utilizza caratteri TeX e in questo modo permette di produrre documenti con un livello di impaginazione professionale.

L'alta qualità sul piano della composizione riguarda anche le formule generate automaticamente e ciò rende TeXmacs adatto come interfaccia per i diversi programmi di computer algebra. TeXmacs supporta anche i linguaggi di estensione Guile/Scheme, attraverso i quali è possibile personalizzzare l'interfaccia o scrivere estensioni dell'editor stesso.

TeXmacs funziona normalmente in sistemi GNU/Linux (sono raccomandati un processore con velocità >200MHz e disponibilità di memoria >32Mb) e su computer sun. Esistono dei convertitori per TeX/LaTeX e sono in fase di sviluppo altri per HTML/MathML/XML. Per il futuro si prevede di far evolvere TeXmacs in modo da trasformarlo in una suite office scientifica completa dotata di un foglio di calcolo, di un ambiente per il disegno tecnico e di strumenti di presentazione.

B.2.2. Perchè la libertà è importante per la scienza

Uno degli obiettivi prioritari di TeXmacs consiste nel promuovere lo sviluppo di software libero per gli scienziati e a cui gli scienziati stessi possano contribuire, in modo da ridurre significativamente i costi di produzione di interfacce utente di alta qualità. A questo proposito, se qualcuno ha in progetto di scrivere un interfaccia tra TeXmacs e altro tipo di software, è invitato a contattarci.

Siamo profondamente convinti che solo i programmi liberi possono essere accettati da un punto di vista scientifico e ciò in ordine a due motivi essenziali:

- Un risultato calcolato da un sistema a codice chiuso non può essere accettato all'interno di una dimostrazione matematica.
- Come un matematico, a partire da teoremi esistenti, ne può costruire potenzialmente dei nuovi, analogamente, sulla base di algoritmi noti, dovrebbe essere possibile rilasciarne altri ottenuti modificando liberamente i precedenti.

Da questo punto di vista appare strano e vergognoso che la maggior parte dei programmi di matematica attualmente in uso siano proprietari. La ragione principale di questo è che spesso i matematici non considerano la programmazione come un'attività scientifica nel vero senso della parola. Di conseguenza, lo sviluppo del software viene delegato agli "ingegneri" e quanto prodotto viene utilizzato a scatola chiusa.

Questa suddivisione dell'attività scientifica appare davvero artificiale; inoltre, dal punto di vista scientifico, è spesso molto importante conoscere il contenuto della scatola chiusa. Inoltre solo attraverso una profonda conoscenza scientifica si può giungere allo sviluppo di software di qualità. Per questo motivo pensiamo che gli scienziati dovrebbero farsi carico dello sviluppo di software considerando a pieno titolo tale attività come scientifica e assimilabile alla stesura di pubblicazioni. In queso senso è anche chiaro che la diffusione del software dovrebbe seguire modalità di sviluppo simili, che garantiscano pubblica disponibilità, riproducibilità e uso libero del software.

B.3. Gli autori di TeXmacs

Il sistema GNU TeXmacs, cha fa parte del progetto GNU, è stato scritto e progettato da Joris van der Hoeven. Il sistema si ispira sia a TeX, scritto da D. Knuth, che a Emacs, scritto da R. Stallman. Ringraziamenti speciali vanno a D. Knuth, a R. Stallmann e al C.N.R.S. (istituto nazionale francese per la ricerca scientifica), di cui l'autore è dipendente, e che ha concesso la libera distribuzione del programma. Ulteriori ringraziamenti sono rivolti a tutti i seguenti contributori:

B.3.1. Sviluppo di TeXmacs

- Andrey Grozin ha contribuito costantemente alla soluzione di molti problemi: interfacciamento con molti programmi di computer algebra, supporto per il Cirillico, tool per l'uso dei dizionari, ecc....
- David Allouche ha rimpiazzato il preprocessore gencc con il più standard sistema di template del C++. Ha anche contribuito con diverse patch, nell'attività di debugging e nell'amministrazione di TeXmacs.

- Dan Grayson ha contribuito all'implementazione della comunicazione via pipes dei programmi di computer algebra. Ha finanziato economicamente TeXmacs e ha offerto numerosi e utili suggerimenti.
- Karim Belabas ha progettatto e sviluppato con l'autore il primo protocollo per l'interfacciamento di TeXmacs con sistemi di calcolo scientifico automatico e con sistemi di computer algebra. Ha implementato l'interfaccia con Pari.
- Stéphane Payrard ha fissato un notevole bug per la distruzione di finestre.
- Michael Graffam per il suo aiuto nella realizzazione dell'interfaccia con GNU Octave.
- Michael Lachmann per il suo lavoro all'imminente rilascio dell'interfaccia GNU R.
- Gwenael Gabard per contributi nella conversione LaTeX to TeXmacs.
- Felix Breuer per l'aiuto nel supporto XML e per una donazione.
- Igor V. Kovalenko per gli aiuti nel debugging di TeXmacs e per alcune patch.
- Gareth McCaughan per numerose patch e per gli utili commenti.
- Jonas Lööf per la procedura accurata di installazione in Cygwin.
- Rob Clark per una patch che ha migliorato il supporto del time system.

B.3.2. Amministrazione di TeXmacs e supporto materiali

- Jean-Claude Fernandez, Fabien Salvi e altri del CRI hosting e amministrazione del sito web di TeXmacs.
- Álvaro Tejero Cantero mantenimento di TeXmacs Wiki.
- Loic Dachary accesso TeXmacs su Savannah.

B.3.3. Porting di TeXmacs su altre piattaforme

- Marciano Siniscalchi porting di TeXmacs su Cygwin.

  Il suo lavoro è stato successivamente perfezionato da Loïc Pottier.
- Martin Costabel porting di TeXmacs su MacOSX.
- Bruno Haible ha contribuito al porting di TeXmacs su sistemi SUN.
- Dan Martens e Stéphane Payrard sono al lavoro per il porting su Windows.

B.3.4. Internazionalizzazione di TeXmacs

Ceco. David Rezac.

Olandese. Joris van der Hoeven.

Finlandese. Teemu Ikonen.

Francese. Michèle Garoche, Joris van der Hoeven.

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

Ungherese. András Kadinger.

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

Polacco. Robert Janusz.

Portoghese. Márcio Laurini and Alexandre Taschetto de Castro.

Rumeno. Dan Ignat.

Russo. Andrey Grozin.

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

Svedese. Harald Ellmann.

Ucraino. Volodymyr Lisivka.

B.3.5. Altri contributori

Infine un ringraziamento a tutti quelli che in qualche forma hanno contribuito TEXMACS, ad esempio riportando bug o offrendo suggerimenti per le versioni future: 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.3.6. Contatti

Potete contattarci via email all'indirizzo

`contact@texmacs.org`

o per posta ordinaria all'indirizzo

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

Sono disponibili numerose mailing list di TeXmacs:

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

B.4. Importanti cambiamenti in TeXmacs

Sotto vengono brevemente riportati i cambiamenti più rilevanti che hanno riguardato TeXmacs a partire dalla versione 0.3.3.15. Per chi desidera maggiori informazioni ricordiamo che viene mantenuto anche un change log più dettagliato.

In generale, se si effettua un aggiornamento ad una nuova versione di TeXmacs, si raccomanda di eseguire un backup dei file precedenti prima di aprirli con la nuova versione. Nel caso sfortunato in cui i vecchi file non venissero aperti correttamente siete pregati di segnalare la presenza di un bug al sito

`bugs@texmacs.org`

spedendo in allegato i file che vengono aperto scorrettamente. Non dimenticate inoltre di menzionare la versione di TeXmacs utilizzata e il sistema su cui il programma lavora.

B.4.1. Tastiera (1.0.0.11 – 1.0.1)

L'uso della tastiera in TeXmacs è stato reso maggiormente razionale. I cambiamenti maggiori sono stati i seguenti:

- Il prefisso `E-` è stato rinominato `M-`.
- `escape` è equivalente a `M-` e `escape`-`escape` a `A-`.
- I comandi dipendenti da Modo hanno come prefisso `A-`. In particolare gli accenti vengono editati utilizzando `A-` al posto di `E-`.
- Le mutazioni di carattere si ottengono utilizzando `tab` al posto di `*` ed è possibile percorrerle all'indietro usando `shift-tab`.
- Le lettere dell'alfabeto greco vengono scritte utilizzando `A-C-`, `F5`, o il tasto modificatore iper configurabile in Edit → Preferences. Le lettere dell'alfabeto greco possono essere ottenute come mutazioni delle lettere dell'alfabeto latino. Ad esempio digitando `p tab` si ottiene π.
- Il significato delle combinazioni dei tasti cursore con control, alt e meta sono stati cambiati.

Puoi scegliere tra numerosi "aspetti" per il comportamento della tastiera in Modifica → Preferenze → Aspetto. L'aspetto di default è Emacs, ma è possibile scegliere Vecchio stile per mantenere il comportamento della tastiera a cui siete abituati.

B.4.2. Menu (1.0.0.7 – 1.0.1)

Parecchi cambiamenti sono intervenuti a questo proposito. I cambiamenti principali sono stati i seguenti:

- Buffer è stato rinominato in Vai.
- Molte voci del menu File sono state spostate in Visualizza.
- I menu Modifica → Importa e Modifica → Esporta sono stati spostati in Tool → Selezioni.

- Il menu Inserisci è stato suddiviso nei menu Inserisci, Testo e Matematica.
- I menu Testo e Paragrafo sono stati inclusi nel menu Formato.
- Opzioni è stato inserito in Documento, Visualizza, Strumenti e Modifica → Preferenze.

B.4.3. Stili (1.0.0.4)

Molti cambiamenti sono intervenuti nell'organizzazione dei file di stile di TeXmacs. File di stile personali, dipendenti da pacchetti TeXmacs intermedi, possono richiedere alcuni piccoli adattamenti.

Stiamo lavorando per rendere stabili i file e i pacchetti dello stile standard. Al termine di questa fase sarà facile adattare i file di stile di LaTeX in TeXmacs. Al più presto sarà resa disponibile la documentazione in linea su questo argomento Aiuto → Documentazione in linea.

B.4.4. Tabelle (0.3.5)

La gestione delle tabelle è completamente cambiata. La composizione di tabelle, matrici, vettori, ecc... è divenuta molto più semplice. Molte caratteristiche sono state aggiunte come il colore di sfondo, i bordi, le sottotavole, ecc.... Tuttavia può accadere che le tabelle redatte con versioni precedenti del programma vengano caricate in modo errato, nel qual caso vi invitiamo a segnalare un bug.

B.4.5. Formato documenti (0.3.4)

Il formato TeXmacs dei documenti è stato profondamente modificato in modo da rendere, in futuro, TeXmacs compatibile con il formato XML. Fatto ancora più importante, i vecchi ambienti di stile come

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

applicati tramite corrispondenza a coppie `<begin|env>text<end|env>`, sono stati rimpiazzati da macro

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

applicate attraverso un'unica espansione `<expand|env|text>`. In modo del tutto analogo, coppie corrispondenti `<set|var|val>text<reset|var>`, relative a variabili ambientali, sono state sostituite da costrutti del tipo `<with|var|val|text>` più prossimi agli attributi XML. Da un punto di vista tecnico queste modifiche comportano complicazioni nel caso in cui il corpo del `testo` sia composto di molti paragrafi. Per questo i documenti strutturati in modo poco accurato possono venire visualizzati in modo scorretto nelle nuove versioni del programa (finora comunque abbiamo rilevato solo una modifica minore nei nostri documenti). Oltre a questo, per mantenere un alto livello di strutturazione del documento il comportamento dell'editor, relativamente ad ambienti che prevedono paragrafi multipli, è stato modificato leggermente.

Appendice C
Contribuire a GNU TeXmacs

C.1. Usare TeXmacs

Uno dei modi migliori per contribuire a GNU TeXmacs consiste nell'utilizzarlo spesso, nel parlarne ad amici e colleghi e nel segnalare all'autore bug o malfunzionamenti in genere. Da questo punto di vista, quando inviate articoli scritti con TeXmacs, siete invitati a segnalarlo. Per fare questo è sufficiente inserire il tag made-by-TeXmacs da qualche parte nel titolo utilizzando il menu Testo → Titolo → Avviso di TeXmacs.

Oltre a questo modo generico (ma molto importante) di contribuire a TeXmacs sono benvenuti tutti i possibili aiuti sui molti aspetti specifici di cui si parla oltre. Quindi, se desiderate contribuire allo sviluppo di qualche aspetto del programma, non esitate a contattarci. Nel menu Aiuto potete trovare la documentazione relativa al codice sorgente di TeXmacs, al suo formato documenti, su come scrivere interfacce in altri formati e così di seguito.

C.2. Fare donazioni al progetto TeXmacs

Fare una donazione al progetto TeXmacs attraverso l'organizzazione SPI.

Un modo importante per supportare TeXmacs consiste nel donare al progetto del denaro. TeXmacs fa attualmente parte dei progetti di pubblico interesse che rientrano in SPI (Software in the Public Interest; vedi `http://www.spi-inc.org`). Potete fare donazioni di denaro a TeXmacs attraverso questa organizzazione, annotando sul vostro check o nella e-mail per effettuare la transazione che il finanziamento è a favore del progetto TeXmacs. Sono accolte anche donazioni di attrezzature e di servizi e donazioni attraverso commercianti. Per ulteriori informazioni potete visitare il sito SPI. Presto verrà mantenuta una pagina web contenente la lista completa delle donazioni; naturalmente resta facoltà del donatore richiedere di non esservi incluso.

Dettagli sulle donazioni.

Per fare una donazione potete indirizzare un check o un vaglia postale a:

> *Software in the Public Interest, Inc.*

al seguente indirizzo:

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

Per una transazione bancaria elettronica (spesso funziona anche in paesi non-US), è necessario consegnare alla banca numero di routing e numero di account come segue:

> Il conto bancario SPI è presso la banca American Express Centurion Bank.

```
Routing Number: 124071889
Account Number: 1296789
```

Non dimenticate di segnalare esplicitamente sul vostro check o nella e-mail che la donazione è rivolta al progetto TeXmacs. In aggiunta è possibile specificare in modo preciso il modo in cui desiderereste venisse impiegato il denaro all'interno del progetto. Per approfondire questo aspetto potete eventualmente contattarci direttamente.

Osservazioni importanti.

Nel caso in cui insorgessero dei problemi contattate il tesoriere di SPI (`treasurer@spi-inc.org`). Una volta completata la transazione elettronica, inviate una copia della ricevuta all'indirizzo sopra menzionato, in modo da garantire l'esistenza di copia della donazione. La copia spedita al tesoriere è importante! Potete anche contattare il team di TeXmacs per essere certi dell'avvenuto deposito del denaro donato sul conto bancario intestato a TeXmacs.

Nota: *l'indirizzo di SPI e le coordinate del relativo conto bancario possono subire nel tempo delle variazioni percui siete invitati a controllare i dati aggiornati all'indirizzo internet di SPI,* `http://www.spi-inc.org/donations`*, prima di fare qualsiasi donazione.*

Donazioni in Europa possono essere effettuate attraverso il nostro partner tedesco, ffis e.V. Se desiderate utilizzare il loro conto bancario (in modo da minimizzare i costi per le transazioni internazionali) seguite le indicazioni disponibili al sito `http://www.ffis.de/Verein/spi-en.html`.

C.3. CONTRIBUIRE ALLA DOCUMENTAZIONE DI GNU $\text{T}_\text{E}\text{X}_\text{MACS}$

Abbiamo grande bisogno sia di documentazione nuova su $\text{T}_\text{E}\text{X}_\text{MACS}$ sia di persone che siano disponibili a tradurre la documentazione esistente in altre lingue. Lo scopo delle seguenti indicazioni è di provvedere alcuni criteri per lo sviluppo di documentazione di qualità. Per questo è necessario leggere attentamente queste linee guida su come redarre la documentazione.

C.3.1. Introduzione a come contribuire

La creazione di documentazione di qualità è questione sia di contenuti che di struttura. I contenuti devono essere quanto più pedagogici possibile in relazione al gruppo di lettori a cui essi sono rivolti. Per questo conviene corredare le esposizioni con esempi e, se necessario, con immagini. Nonostante la documentazione non debba essere necessariamente completa è auspicabile che una sua parte sia stabile e corretta dal punto di vista dell'ortografia. Le parti sperimentali di documentazione verranno inserite nella directory `incoming` o in $\text{T}_\text{E}\text{X}_\text{MACS}$ Wiki.

Un ulteriore importante aspetto riguarda la struttura da conferire ai documenti che deve essere curata e organizzata ricorrendo principalmente all'uso dei marcatori inclusi nello stile `tmdoc`. La strutturazione dei documenti serve per poter compilare automaticamente libri stampabili, per poter utilizzare diverse modalità di visualizzazione degli stessi e per rendere più efficenti le ricerche di informazioni contenute al loro interno. In modo particolare ogni documento deve contenere indicazioni sul copyright e sulla licenza e, se esso fa riferimento a molti file, deve contenere le indicazioni per poter spostarsi automaticamente all'interno della restante documentazione.

AVVISO C.1. Se state traducendo un documento non dimenticate di selezionare il comando Documento → Lingua → Your language. In questo modo alcune parti del documento, come i menu o le istruzioni, verranno automaticamente tradotte. Si raccomanda anche l'utilizzo del correttore ortografico di TeXmacs il cui uso è chiaramente condizionato dalla scelta della lingua in cui è redatto il documento.

C.3.2. Usare CVS

L'attuale documentazione di TeXmacs è mantenuta nel sito `http://savannah.gnu.org` utilizzando il sistema CVS (Concurrent Version System). Per contribuire alla documentazione è necessario prima di tutto crearsi un account personale nel sito sopra indicato e succeessivamente spedire un'email a Joris van der Hoeven (`vdhoeven@texmacs.org`) chiedendo di essere inclusi nella lista degli addetti alla documentazione. Solo dopo aver fatto questo sarete in grado di accedere al CVS per aggiungere o modificare la documentazione. Per informazioni più dettagliate potete consultare il sito `http://savannah.gnu.org/cvs/?group_id=1747`.

Il sistema CVS non appare del tutto conveniente ai nostri propositi per lo sviluppo della documentazione, in quanto risulta poco dinamico. Per questo, in futuro, si pensa di creare un sito web interamente dedicato alle pubblicazioni, che permetta sia di salvare la documentazione direttamente sul web, sia di effettuare automaticamente la conversione in diversi formati e la compilazione di libri, ecc....

C.3.3. Convenzioni e nomi dei file

La documentazione viene organizzata in funzione dell'argomento all'interno di un albero di directory. Le sottodirectory della directory principale sono le seguenti:

devel. documentazione per sviluppatori;

examples. esempi di documenti in TeXmacs;

incoming. documentazione in stato embrionale;

main. documentazione principale;

meta. come scrivere documentazione e come compilarla.

Come indicazione generale si cerca di mantenere il numero di file per directory ragionevolmente ridotto.

I nomi dei file nella directory principale (main) hanno la forma `tipo-nome.lingua.tm`. Nelle altre directory la forma è `nome.lingua.tm`. Con `tipo` si intende un'indicazione di massima riguardante la caratteristica generale della documentazione come:

adv. documentazione per utilizzatori esperti;

man. documentazione per il manuale di TeXmacs;

tut. documentazione per il tutorial di TeXmacs.

La documentazione riferita ad uno stesso argomento viene tenuta unita, indipendentemente dal tipo. In questo modo risulta più semplice trovare parti di documentazione esistente relative ad uno stesso argomento. Può accadere inoltre che parti della documentazione inizialmente scritte per il tutorial vengano inserite nel manuale o viceversa. La `lingua` in cui viene redatta la documentazione viene indicata da un codice di due lettere come, ad esempio, `en`, `fr` e così via. Il `nome` di ciascun documento rimane invariato all'interno delle traduzioni nelle diverse lingue. Così, ad esempio, il documento `man-keyboard.en.tm`, una volta tradotto in francese, non diviene `man-clavier.fr.tm`. ma semplicemente `man-keyboard.fr.tm`.

C.3.4. Copyright information & the Free Documentation License

All documentation on the `texmacs-doc` site falls under the GNU Free Documentation License. If you write documentation for TEXMACS on this site, then you have to agree that it will be distributed under this license too. The copyright notice

```
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".
```

should be specified at the end of *each* file. This should be done inside the tmdoc-license macro, in a similar way as at the end of the present document. When automatically generating a printed book from several documentation files, this will enable us to include the license only once.

You keep (part of) the copyright of all documentation that you will write for TEXMACS on the official `texmacs-doc` site. When you or others make additions to (or modifications in, or translations of) the document, then you should add your own name (at an appropriate place, usually at the end) to the existing copyright information. The copyright notice should be specified using the tmdoc-copyright function just before the license information at the end of the document. The first argument of this function contains a year or a period. Each remaining argument indicates one of the copyright holders. When combining (pieces of) several documents into another one, you should merge the copyright holders. For cover information (on a printed book for instance), you are allowed to list only the principal authors, but a complete list should be given at a clearly indicated place.

C.3.5. Spostamenti automatici nella documentazione

Come regola generale, è preferibile evitare, all'interno della documentazione di TEXMACS, il ricorso a testi lunghi e molto articolati, cercando di scrivere invece documenti brevi e su argomenti ben precisi. In una seconda fase conviene predisporre dei “meta file” di aiuto per agevolare gli spostamenti automatici da una parte all'altra della documentazione. Questa strategia consente, tra l'altro, di riutilizzare una stessa pagina del manuale in ambienti diversi (manuale stampato, tutorial sul web, ecc...).

Lo stile tmdoc prevede la definizione di tre possibili macro per indicare come spostarsi all'interno della documentazione. La macro traverse viene usata per racchiudere regioni di testo che contengono informazioni trasversali. La macro branch indica una pagina di aiuto considerata come una sottosezione e la macro continue indica una pagina che segue. Sia la macro branch che la macro continue hanno per riferimento due argomenti. Il primo argomento descrive il link mentre il secondo fornisce l'indirizzo fisico della pagina a cui il link si riferisce.

Tipicamente al termine di un meta file di aiuto si troveranno numerose macro di tipo branch o continue all'interno di una macro di tipo traverse. All'inizio del documento il titolo viene specificato utilizzando la macro tmdoc-title. Nel momento in cui si desidera generare un manuale stampato la struttura titolo-capitolo-sezione-paragrafo verrà automaticamente generata, a partire da tutte le informazioni contenute in questo ambiente di macro. In alternativa è possibile generare dei bottoni addizionali per consentire la navigazione all'interno del documento, utilizzando un qualsiasi browser.

C.3.6. Usare lo stile tmdoc

Oltre alle macro relative alle informazioni sul copyright e agli spostamenti all'interno della documentazione di cui abbiamo parlato in precedenza, lo stile tmdoc offre un certo numero di macro e di funzioni che, se necessario, possono essere di volta in volta attivate:

key. questa macro viene utilizzata per indicare comandi da tastiera come `C-x C-s`. Le macro specializzate kbd-gen, kbd-text, kbd-math, kbd-symb, kbd-big, kbd-large, kbd-ia, kbd-exec e kbd-table vengono utilizzate per comandi da tastiera corrispondenti ad uno specifico tipo di azione o di modalità. Ad esempio, kbd-math corrisponde al comando da tastiera per l'inserimento di espressioni matematiche come `A-f` che indica l'inizio di una frazione.

menu. questa funzione, che si riferisce ad un arbitrario numero di argomenti, permette di scrivere dei menu come File o Documento → Lingua. La funzione esegue automaticamente la traduzione delle voci del menu che vengono inserite.

markup. questa macro viene utilizzata per indicare macro o funzioni come section.

tmstyle. questa macro indica il nome di un file di stile di $\text{T}_\text{E}\text{X}_\text{MACS}$ come article.

tmpackage. questa macro indica il nome di un pacchetto $\text{T}_\text{E}\text{X}_\text{MACS}$ come ad esempio il pacchetto std-markup.

tmdtd. questa macro indica il nome di un $\text{T}_\text{E}\text{X}_\text{MACS}$ d.t.d. come number-env.

Osserviamo che nessun nome di macro deve essere tradotto in lingua straniera! Oltre a questo si ricorda di non tradurre mai le voci dei menu, le quali vengono tradotte automaticamente in modo da garantire la sincronizzazione tra la traduzione dei manuali e la localizzazione dei menu di $\text{T}_\text{E}\text{X}_\text{MACS}$. Nel caso di marcatori, stili e pacchetti d.t.d., è importante non alterarne il nome originale in quanto, il più delle volte, esso corrisponde al nome di un file.

Le seguenti macro e funzioni, pensate per realizzare link e per gestire gli indici, verranno implementate in futuro:

simple-link. questa macro ha come argomento un hyperlink URL x, dove x rappresenta nome e destinazione dell'hyperlink;

hyper-link. questa macro definisce un normale hyperlink;

concept-link. questa macro ha come argomento un concetto. Successivamente potrebbe essere creato automaticamente un appropriato hyperlink a partire da essa e da altra documentazione;

only-index. indice di una stringa;

def-index. definizione di un nuovo concetto: il testo viene scritto in italico e indicizzato;

re-index. comparsa di un concetto definito in precedenza: il testo viene scritto in roman e inserito nell'indice.

In conclusione elenchiamo i seguenti tag di uso frequente:

icon. link ad una icona in una directory centrale come `$TEXMACS_PATH/doc/images/pixmaps`

screenshot. link a una schermata. Le schermate sono attualmente contenuta nella directory centrale `$TEXMACS_PATH/doc/images/screenshots`

scheme. il linguaggio SCHEME;

cpp. il linguaggio C++;

framed-fragment. per visualizzare una parte di codice in un riquadro;

scheme-fragment. per un codice SCHEME multi-paragrafo;

cpp-fragment. per un codice C++ multi-paragrafo;

tm-fragment. per una parte di codice TEXMACS in formato SCHEME;

scheme-code. per una parte di codice SCHEME;

cpp-code. per una parte di codice C++;

descriptive-table. per tavole descrittive che possono essere utilizzate per realizzare liste di comandi da tastiera, di marcatori, ecc....

Lo stile `tmdoc` eredita numerose macro dallo stile `generic` percui, se necessario, è possibile utilizzare macro come em, verbatim, itemize, e così di seguito.

C.4. INTERNAZIONALIZZAZIONE

Il supporto del maggior numero possibile di lingue straniere rappresenta per TEXMACS una grande sfida in cui il vostro aiuto è estremamente gradito. Il lavoro di traduzione per il supporto in una nuova lingua richiede normalmente molti giorni di lavoro. Per questo si raccomanda di intraprenderlo in compagnia di amici o colleghi, che possano offrirvi un aiuto.

La procedura per l'aggiunta di una nuova lingua è la seguente:

- Si copia il file `english-new.scm` in `english-`*`nuovalingua`*`.dic` tipicamente nella seguente directory `/langs/natural/dic` e se ne esegue la traduzione. Per fare questo potresti aver bisogno del dizionario di Andrey Grozin reperibile all'indirizzo

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

  Prima di utilizzare questo strumento assicurati che Python sia installato nel tuo sistema, quindi scarica il file, scompattalo usando gunzip, rendilo eseguibile e quindi usalo.

- Siete pregati di tenerci al corrente dell'esistenza di eventuali regole tipografiche peculiari della vostra lingua e di comandi da tastiera utili per produrre caratteri speciali.

- Nonostante sia nostro compito occuparci dei problemi tipografici e di sillabazione voi siete coloro che poi verificano sul campo la bontà del lavoro svolto.

- Se avete abbastanza tempo a disposizione potete prendervi carico della traduzione di tutta o parte della documentazione attualmente esistente.

Naturalmente il supporto per le lingue diviene obsoleto ogni volta che vengono aggiunte a TEXMACS nuove funzionalità. Per questo motivo viene mantenuto il file `miss-english-`*`yourlanguage`*`.dic` in cui sono elencati i termini mancanti della traduzione nella vostra lingua eventualmente aggiunti da poco. Non esitate a segnalare l'esistenza di versioni incomplete di `english-`*`yourlanguage`*`.dic` o di `miss-english-`*`yourlanguage`*`.dic` in quanto forse qualcuno desidera completarle.

C.5. Scrivere convertitori di dati

Se avete familiarità con TEX, LATEX, Html, Xml, Sgml, Mathml, Pdf, Rtf, o con qualche altro formato di dati di uso frequente potreste contribuire a TEXMACS scrivendo un buon programma di conversione per uno o più formati.

Scrivere un convertitore Pdf adattando il file `src/Window/PsDevice/printer.cpp` non dovrebbe presentare grosse difficoltà. Convertitori per altri formati potrebbero presentare problemi maggiori di implementazione e forse richiedere una più stretta collaborazione con gli autori principali di TEXMACS. In Aiuto → Codice sorgente → Formato dei dati sono disponibili dettagli sul formato dati di TEXMACS e in Aiuto → Codice sorgente → Conversione di dati vengono forniti suggerimenti utili per questo genere di progetti.

C.6. Portare TEXMACS su altre piattaforme

Dal momento che abbiamo principalmente accesso a sistemi PC/Linux e SUN, siamo interessati a collaborare con persone che desiderano portare TEXMACS su altri sistemi Unix o con X Window, e a mantenere la corrispondente distribuzione. Se desiderate fare questo date un'occhiata ai file:

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

Sono benvenuti gli specialisti di `autoconf`, redhat e rpm, non da ultimo per comunicare suggerimenti, patches, ecc....

Oltre al porting di TeXmacs su altri sistemi Unix, sarebbe interessante avere il porting di TeXmacs sotto Windows e Mac OS. Per dare aiuti in questa direzione conviene prima di tutto iscriversi alla mailing list `texmacs-dev@gnu.org`. Sono in corso discussioni su come eseguire i diversi porting e su quale interfaccia grafica sia più adatta a questo scopo (Gtk, Qt, Wxwindows o GNUstep). La nostra strategia attuale consiste nel sistemare tutte le parti di codice relative alla GUI in una ben specificata TMGUI API e solo successivamente eseguirne il porting. Ciò permette di supportare sistemi multipli di interfacce grafiche. Ulteriori dettagli si possono comunque avere consultando gli archivi della mailing list `texmacs-dev@gnu.org`.

C.7. Interfacciare TeXmacs con altri sistemi

Risulta assai semplice interfacciare TeXmacs con sistemi di computer algebra o con programmi scientifici che producono output strutturati. Tenete in considerazione la possibilità di scrivere un'interfaccia tra TeXmacs e i vostri programmi scientifici preferiti. TeXmacs è già interfacciato con numerosi altri programmi scientifici come Giac, Macaulay 2, Maxima, GNU Octave, Pari, Qcl, gTybalt, Yacas. La documentazione dettagliata su come realizzare nuove interfacceè disponibile nel menu Aiuto → Interfacciare.

C.8. Diventare uno sviluppatore di TeXmacs

Oltre al genere di contributi che è stato descritto dettagliatamente in precedenza, ci sono molti altri aspetti in cui un aiuto sarebbe gradito. Per ulteriori dettagli potete dare un'occhiata ai nostri progetti futuri. Naturalmente sentitevi liberi di proporre idee nuove e di condividerle con noi nella mailing list `texmacs-dev@gnu.org`!

Appendice D

Interfacciare $\TeX_{\text{MACS}}$ con altri programmi

D.1. L'esempio di "mycas"

Il modo migliore per iniziare ad implementare in $\TeX_{\text{MACS}}$ una nuova interfaccia, consiste nello studiare attentamente l'esempio del sistema di computer algebra `mycas`, che è contenuto nella directory `$TEXMACS_PATH/misc/mycas`. Il file `mycas.cpp`, il cui listato è disponibile alla fine di questa sezione, contiene un esempio molto semplice di programma che può essere interfacciato con $\TeX_{\text{MACS}}$. Per testare il programma è necessario compilarlo utilizzando il comando:

```
g++ mycas.cpp -o mycas
```

e spostare il file binario `mycas` in qualche punto del vostro path. All'avvio di $\TeX_{\text{MACS}}$ è quindi necessario avere a disposizione la voce Mycas nel menu Testo → Sessione.

D.2. Studiamo il codice sorgente passo dopo passo

Iniziamo a studiare, passo dopo passo, il codice di `mycas`. In primo luogo osserviamo come ogni comunicazione avvenga utilizzando delle pipes e coinvolga standard input e output. Per permettere a $\TeX_{\text{MACS}}$ di comprendere quando l'output del sistema ha terminato è necessario che ogni output sia incapsulato in blocchi, utilizzando tre speciali caratteri di controllo:

```
#define DATA_BEGIN   ((char) 2)
#define DATA_END     ((char) 5)
#define DATA_ESCAPE  ((char) 27)
```

Il carattere `DATA_ESCAPE` seguito da un qualsiasi altro carattere c può essere usato per produrre c, anche se c è uno dei tre caratteri di controllo. Un'illustrazione di come utilizzare `DATA_BEGIN` e `DATA_END` è data dal seguente messaggio di inizio:

```
int
main () {
  cout << DATA_BEGIN << "verbatim:";
  cout << "------------------------------------------------------\n";
  cout << "Welcome to my test computer algebra system for TeXmacs\n";
  cout << "This software comes with no warranty whatsoever\n";
  cout << "(c) 2001  by Joris van der Hoeven\n";
  cout << "------------------------------------------------------\n";
  next_input ();
  cout << DATA_END;
  fflush (stdout);
```

La prima linea del `main` definisce il formato del carattere del messaggio di inizio che è di tipo "verbatim". La funzione `next_input`, che viene richiamata al termine del messaggio di inizio, serve per stampare un prompt e se ne discuterà in dettaglio nel seguito. In conclusione `DATA_END` chiude il blocco del messaggio di inizio e informa $\TeX_{\text{MACS}}$ che `mycas` è in attesa di un input. Non dimenticate alla fine di svuotare lo standard output, in modo che $\TeX_{\text{MACS}}$ possa ricevere l'intero messaggio.

Il ciclo principale inizia chiedendo l'input dallo standard input:

```
  while (1) {
    char buffer[100];
    cin >> buffer;
    if (strcmp (buffer, "quit") == 0) break;
```

L'output che viene restituito deve essere ancora una volta incluso in un blocco tipo `DATA_BEGIN-DATA_END`.

```
    cout << DATA_BEGIN << "verbatim:";
    cout << "You typed " << buffer << "\n";
```

All'interno di tale blocco è possibile inviare ricorsivamente altri blocchi che possono essere specificati in formati diversi. Ad esempio, il seguente codice spedisce una formula in $\LaTeX$:

```
    cout << "And now a LaTeX formula: ";
    cout << DATA_BEGIN << "latex:" << "$x^2+y^2=z^2$" << DATA_END;
    cout << "\n";
```

Per alcuni scopi particolari può essere utile spedire direttamente l'output in formato $\TeX_{\text{MACS}}$ utilizzando una rappresentazione SCHEME:

```
    cout << "And finally a fraction ";
    cout << DATA_BEGIN << "scheme:" << "(frac \"a\" \"b\")" <<
DATA_END;
    cout << ".\n";
```

Per concludere, dobbiamo nuovamente inviare l'output `DATA_END` e svuotare lo standard output:

```
    next_input ();
    cout << DATA_END;
    fflush (stdout);
  }
  return 0;
}
```

Osserviamo che non è possibile inviare più di un blocco `DATA_BEGIN-DATA_END` alla volta. Appena il primo blocco `DATA_BEGIN-DATA_END` è stato ricevuto da $\TeX_{\text{MACS}}$, si assume che il sistema sia in attesa dell'input. Se si vuole inviare più di un blocco `DATA_BEGIN-DATA_END` tutti i blocchi da inviare devono essere inclusi in un blocco principale.

Un "canale" speciale viene impiegato per inviare il prompt di input. I canali sono specificati come blocchi `DATA_BEGIN-DATA_END` speciali:

```
static int counter= 0;

void
next_input () {
  counter++;
  cout << DATA_BEGIN << "channel:prompt" << DATA_END;
  cout << "Input " << counter << "] ";
}
```

All'interno del canale di prompt è possibile utilizzare ancora blocchi DATA_BEGIN-DATA_END tuttavia in modo annidato. In questo modo è possibile ad esempio utilizzare come prompt una formula. Sono disponibili tre canali standard:

output. canale predefinito per un output normale;

prompt. per inviare prompt di input;

input. per specificare un valore predefinito per l'input successivo.

D.3. Output grafici

È possibile anche inviare output grafici in formato postscript. Immaginiamo ad esempio di avere una figura picture.ps nella home directory. Allora inserendo le linee di codice:

```
cout << "A little picture:\n";
cout << DATA_BEGIN << "ps:";
fflush (stdout);
system ("cat $HOME/picture.ps");
cout << DATA_END;
cout << "\n";
```

nel punto appropriato del ciclo principale si otterrà la visualizzazione della figura nel centro dell'output.

D.4. Il listato completo del programma

```
#include <stdio.h>
#include <stdlib.h>
#include <string.h>
#include <iostream.h>

#define DATA_BEGIN   ((char) 2)
#define DATA_END     ((char) 5)
#define DATA_ESCAPE  ((char) 27)

static int counter= 0;

void
next_input () {
  counter++;
  cout << DATA_BEGIN << "channel:prompt" << DATA_END;
  cout << "Input " << counter << "] ";
}

int
main () {
  cout << DATA_BEGIN << "verbatim:";
  cout << "---------------------------------------------------\n";
  cout << "Welcome to my test computer algebra system for TeXmacs\n";
```

```
  cout << "This software comes with no warranty whatsoever\n";
  cout << "(c) 2001  by Joris van der Hoeven\n";
  cout << "-----------------------------------------------------\n";
  next_input ();
  cout << DATA_END;
  fflush (stdout);

  while (1) {
    char buffer[100];
    cin >> buffer;
    if (strcmp (buffer, "quit") == 0) break;
    cout << DATA_BEGIN << "verbatim:";
    cout << "You typed " << buffer << "\n";

    cout << "And now a LaTeX formula: ";
    cout << DATA_BEGIN << "latex:" << "$x^2+y^2=z^2$" << DATA_END;
    cout << "\n";

    cout << "And finally a fraction ";
    cout << DATA_BEGIN << "scheme:" << "(frac \"a\" \"b\")" << DATA_END;
    cout << ".\n";

    next_input ();
    cout << DATA_END;
    fflush (stdout);
  }
  return 0;
}
```

D.5. Supporting your system TeXmacs

Supponiamo che, seguendo quanto spiegato nella sezione precedente, siate riusciti a scrivere una prima interfaccia per TeXmacs di un certo programma applicativo. Allora è giunto il momento di includere un primo supporto per il vostro sistema in una distribuzione standard di TeXmacs, dopodichè sarà possibile migliorarne le prestazioni.

Dalla versione 1.0.1.5 in poi, è diventato semplice adattare una generica interfaccia in modo da poterla integrare direttamente in TeXmacs. L'idea è di creare una directory:

`$TEXMACS_HOME_PATH/plugins/myplugin`

dove `myplugin` è il nome del corrispondente plugin. Ricordiamo che `$TEXMACS_HOME_PATH` è assimilato per defult a `~/.TeXmacs`. Nella directory `$TEXMACS_PATH/plugins` potete trovare tutti i plugin standard disponibili nella distribuzione di TeXmacs che state usando. I plugin già implementati rappresentano altrettanti buoni esempi da cui prendere spunto.

La directory `myplugin` dovrebbe contenere una struttura di sottodirectory analoga a quella di `$TEXMACS_PATH`, anche se è possibile omettere le directory che non vengono direttamente utilizzate. In ogni caso deve essere fornito un file `progs/init-myplugin.scm` che descriva come inizializzare il vostro plugin. Tipicamente questo file contiene solo una istruzione in SCHEME, con una forma del genere:

```
(plugin-configure myplugin
  (:require (file-in-path "myplugin"))
  (:launch "shell-cmd")
  (:format "input-format" "output-format")
  (:session "Myplugin"))
```

La prima istruzione è un predicato che stabilisce se il vostro plugin può essere utilizzato o meno su un dato sistema. Normalmente si verifica se il programma cui fa riferimento il plugin è disponibile nel path. Le istruzioni seguenti vengono eseguite solo se la richiesta precedente è soddisfatta. L'istruzione `:launch` specifica che il plugin viene lanciato utilizzando `shell-cmd`. Il comando `shell-cmd` normalmente ha la forma `myplugin --texmacs`. L'istruzione `:format` specifica quali sono i formati utilizzati in input e in output. Tipicamente `input-format` è `verbatim` e `output-format` è `generic`. Altri formati disponibili sono `scheme`, `latex`, `html` e `ps`. L'istruzione `:session` rende disponibili le sessioni da shell del plugin dal menu Testo → Sessione → Myplugin.

Se tutto funziona e desiderate rendere possibile ad altri l'utilizzo del plugin inserendolo in una distribuzione ufficiale di TeXmacs contattate Joris van der Hoeven all'indirizzo internet `vdhoeven@texmacs.org`.

D.6. Further customization of the interface

Having written a working interface between your system and TeXmacs, you may want to improve it further. Below, we will discuss a few directions for possible improvement.

First of all, you may want to customize the keyboard behavior inside a `myplugin`-session and add appropriate menus. The mechanisms for doing that are described in the chapter about the GUILE/SCHEME extension language and you may add such support to the file `init-myplugin.scm`. We again recommend you to take a look at the plugins which are shipped with TeXmacs inside the directory `$TEXMACS_HOME_PATH/plugins`.

Certain output from your system might require a special markup. For instance, assume that you want to associate an invisible type to each subexpression in the output. Then you may create a macro `exprtype` with two arguments in `myplugin.ts` and send LaTeX expressions like `\exprtype{1}{Integer}` to TeXmacs during the output.

In the case when you connected your system to TeXmacs using pipes, you may directly execute TeXmacs commands during the output from your system by incorporating pieces of code of the form:

```
[DATA_BEGIN]command:scheme-program[DATA_END]
```

in your output. Inversily, when the cursor is inside a session of your system, you may use the SCHEME command:

```
(extern-exec plugin-command)
```

in order to execute a command of your system.

D.7. Linking your system as a dynamic library

Instead of connecting your system to TeXmacs using a pipe, it is also possible to connect it as a dynamically linked library. Although communication through pipes is usually easier to implement, more robust and compatible with gradual output, the second option is faster.

D.8. Connections via dynamically linked libraries

Let us now describe the steps you have to go through in order to link your system as a dynamic library.

1. Modify the architecture of your system in such a way that the main part of it can be linked as a shared library; your binary should typically become a very small program, which handles verbatim input and output, and which is linked with your shared library at runtime.

2. Copy the include file `$TEXMACS_PATH/include/TeXmacs.h` into the include directory of your system's source and write the input/output routines as required by the last TeXmacs communication protocol as explained below.

3. Include a line of the form:

   ```
   (package-declare "myplugin" "libmyplugin.so" "get_name_package"
   "init")
   ```

   in your file `init-myplugin.scm` which has been described in the case of communication by pipes. Here `libmyplugin.so` is the corresponding shared library, `get_name_package` the function which will be called by TeXmacs in order to link your system to TeXmacs, and `init` some initialization string for your package.

4. Proceed in a similar way as in the case of communication by pipes.

D.9. The TeXmacs communication protocol

The TeXmacs communication protocol is used for linking libraries dynamically to TeXmacs. The file `$TEXMACS_PATH/include/TeXmacs.h` contains the declarations of all data structures and functions used by the protocol. Actually, we foresee a succession of different protocols. Each of these protocols have the abstract data structures `TeXmacs_exports` and `package_exports` in common, with information about the versions of the protocol, TeXmacs and your package.

The n-th concrete version of the communication protocol should provide two data structures `TeXmacs_exports_n` and `package_exports_n`. The first structure contains all routines and data of TeXmacs, which may be necessary for the package. The second structure contains all routines and data of your package, which should be visible inside TeXmacs.

In order to link your system to TeXmacs, you have to implement a function:

```
package_exports* get_my_package (int version);
```

This function takes the highest TeXmacs communication protocol supported by your TeXmacs system on input. It should return a pointer to an instance of a concrete structure `package_exports_n`, where `n` is inferior or equal to `version`.

D.10. Version 1 of the TeXmacs communication protocol

In the first version of the TeXmacs communication protocol, your package should export an instance of the following data structure:

```
typedef struct package_exports_1 {
  char* version_protocol; /* "TeXmacs communication protocol 1" */
  char* version_package;
  char* (*install) (TeXmacs_exports_1* TM, char* options, char**
errors);
  char* (*evaluate) (char* what, char* session, char** errors);
  char* (*execute) (char* what, char* session, char** errors);
} package_exports_1;
```

The string `version_protocol` should contain `"TeXmacs communication protocol 1"` and the string `version_package` the version of your package.

The routine `install` will be called once by TeXmacs in order to initialize your system with options `options`. It communicates the routines exported by TeXmacs to your system in the form of `TM`. The routine should return a status message like:

```
"yourcas-version successfully linked to TeXmacs"
```

If installation failed, then you should return `NULL` and `*errors` should contain an error message. Both `what` and the returned string have a special format, in which it is possible to encode arbitrary TeXmacs documents. This format will be explained in the next section.

The routine `evaluate` is used to evaluate the expression `what` inside a TeXmacs-session with name `session`. It should return the evaluation of `what` or `NULL` if an error occurred. `*errors` either contains one or more warning messages or an error message, if the evaluation failed. The command:

```
(package-format "yourcas" "input-format" "output-format")
```

is used in order to specify the input and output formats for evaluations, in a similar way as in the case of pipes.

The routine `execute` has a similar specification as `evaluate`, except that it is not used for the evaluation of expressions inside a TeXmacs-session, but rather for other communication purposes between TeXmacs and your package.

IMPORTANTE D.1. All strings returned by the routines `install`, `evaluate` and `execute`, as well as all warning and error messages should be allocated using `malloc`. They will be freed by TeXmacs using `free`.

The first version of the TeXmacs communication protocol also requires TeXmacs to export an instance of the data structure:

```
typedef struct TeXmacs_exports_1 {
  char* version_protocol; /* "TeXmacs communication protocol 1" */
  char* version_TeXmacs;
} TeXmacs_exports_1;
```

The string `version_protocol` contains the version `"TeXmacs communication protocol 1"` of the protocol and `version_TeXmacs` the current version of TeXmacs.

D.11. FUTURE PROJECTS

There are many improvements to be made in the TeXmacs interface to computer algebra systems. First of all, the computer algebra sessions have to be improved (better hyphenation, folding, more dynamic subexpressions, etc.). As to the real interface with computer algebra systems, the upcoming changes fall into two categories:

- Changes to the existing interface.

- Support for communication between computer algebra systems.

An example of a change of the first type would be support for the automatic completion of commands. The second project is far more ambitious and targets a semantically safe way to communicate mathematical data. We plan to write a stand-alone program for this, which can be used independently from TeXmacs.

Indice analitico