

MOSÈ GIORDANO, ORLANDO IOVINO, MATTEO LECCARDI

GUIDA PRATICA ALL'USO
DI
GNU EMACS E AUCTIONTEX



GUIDAEMACSAUCTIONTEX.TEX V.1.0 DEL 2013/03/12

LICENZA D'USO

Quest'opera è soggetta alla Creative Commons Public License versione 3.0 o posteriore: *Attribuzione, Non Commerciale, Condividi allo stesso modo*. Un riassunto della licenza in linguaggio accessibile a tutti è reperibile sul sito ufficiale <http://creativecommons.org/licenses/by-nc-sa/3.0/deed.it>.

TU SEI LIBERO:

- Di riprodurre, distribuire, comunicare al pubblico, esporre in pubblico, rappresentare, eseguire e recitare quest'opera.
- Di modificare quest'opera.

ALLE SEGUENTI CONDIZIONI:

- ⓘ Devi attribuire la paternità dell'opera nei modi indicati dall'autore o da chi ti ha dato l'opera in licenza e in modo tale da non suggerire che essi avallino te o il modo in cui tu usi l'opera.
- Ⓢ Non puoi usare quest'opera per fini commerciali.
- Ⓒ Se alteri o trasformi quest'opera, o se la usi per crearne un'altra, puoi distribuire l'opera risultante solo con una licenza identica o equivalente a questa.

PRESENTAZIONE

Emacs è uno dei più vecchi e potenti editor di testi in circolazione e può vantare fra i suoi utenti Donald Knuth, l'inventore di T_EX, e Leslie Lamport, l'autore di L^AT_EX.

AUCT_EX, invece, è un pacchetto per Emacs scritto interamente in linguaggio Emacs Lisp, che estende notevolmente le funzionalità di Emacs per produrre documenti in L^AT_EX e in altri formati legati al programma di tipocomposizione T_EX.

In questa guida tematica parleremo del programma GNU Emacs (ovvero del programma Emacs del progetto GNU, mentre non si accennerà alle sue diverse varianti) e del pacchetto AUCT_EX. Si spiegherà come installarli e configurarli sui tre principali sistemi operativi in circolazione (Windows, GNU/Linux e Mac OS), e come usarli per scrivere documenti L^AT_EX. Per finire si darà qualche indicazione per personalizzare e rendere più efficiente Emacs.

Si tenga presente che, trattando di un programma così potente come Emacs, la guida è ben lontana dall'essere esauriente. Per ogni altro approfondimento non riportato in queste righe rimandiamo il lettore ai manuali ufficiali, in particolare a [STALLMAN \(2012\)](#) per Emacs e [THORUP *et al.* \(2012\)](#) per AUCT_EX. Speriamo, con queste brevi note, di convincere il lettore che Emacs non è poi così complicato come spesso viene considerato.

COLOPHON & RINGRAZIAMENTI

Questa guida tematica è stata scritta a *sei mani* grazie al sistema di controllo di versione git, che si scarica dal sito <http://git-scm.com>. Chi vuole collaborare può “clonare” il progetto dal deposito ufficiale presente sul sito <https://github.com/> alla pagina <https://github.com/GuITeX/guidaemacsauctex>. Per compilare il codice sorgente si consiglia di munirsi

di una distribuzione L^AT_EX aggiornata. È inoltre necessario scaricare e installare nel proprio albero personale la classe *guidatematica* disponibile alla pagina <https://github.com/GuITeX/guidatematica>.

Ringraziamo qui Tommaso Gordini per aver, da letterato qual egli è, reso il testo più leggibile, oltre ad averci dato suggerimenti importanti per la comprensione della guida.

È gradita ogni forma di collaborazione. Chiunque voglia segnalare errori, refusi, commenti e suggerimenti può farlo agli indirizzi di posta elettronica qui riportati.

GLI AUTORI

MOSÈ GIORDANO

giordano dot mose at libero dot it

ORLANDO IOVINO

orlando dot iovino at yahoo dot it

MATTEO LECCARDI

matteo dot leccardi at gmail dot com

INDICE

LICENZA D'USO	II
PRESENTAZIONE	III
1 INTRODUZIONE	1
2 Emacs E AUCTION SU WINDOWS	3
2.1 Installare Emacs	3
2.2 Installare AUCTION	3
2.3 Correttore ortografico	4
2.4 Sincronizzare sorgente e anteprima	4
3 Emacs E AUCTION SU GNU/LINUX	7
3.1 Installare Emacs	7
3.2 Installare AUCTION	8
3.3 Correttore ortografico	8
4 Emacs E AUCTION SU MAC OS	9
4.1 Installare Emacs	9
4.2 Installare AUCTION	9
4.3 Correttore ortografico	11
4.4 Sincronizzare sorgente e anteprima	11
5 PRIMI PASSI IN Emacs	13
5.1 Ciao mondo	13
5.2 Correzione ortografica	16
6 USO AVANZATO DI Emacs	17
6.1 Un documento più complesso	17

6.2	Le variabili locali	18
6.3	Sinossi delle principali scorciatoie	20
7	PERSONALIZZARE Emacs E AUCTION	23
	BIBLIOGRAFIA	26

Lo sviluppo di Emacs è cominciato negli anni '70 del Novecento ai laboratori di Intelligenza Artificiale del MIT, cioè molto prima che si diffondesse l'informatica che conosciamo oggi. Ciò comporta che Emacs utilizza una terminologia spesso diversa da quella utilizzata dalla stragrande maggioranza degli altri programmi che usiamo quotidianamente. Perciò non bisogna sorprendersi se l'operazione di *tagliare* il testo (generalmente indicata in inglese con *cutting*) in Emacs viene chiamata *killing* e quella di *incollare* il testo copiato (in inglese *pasting*) viene chiamata *yanking*.

Emacs è famoso perché rende possibile eseguire *qualsiasi* operazione usando solo la tastiera, cioè senza l'ausilio del mouse. A ogni comando può essere associata una combinazione di tasti. Sebbene questo fatto possa inizialmente spaventare, è un vero punto di forza di Emacs, perché permette di eseguire le operazioni in maniera rapida, invece di dover spostare il mouse alla ricerca di appositi pulsanti. In questa guida seguiremo la notazione diffusa nel mondo di Emacs per indicare i tasti. Quindi il tasto `Ctrl` verrà indicato con `C`, l'`Invio` con `RET`, il tasto `←→` con `TAB`, mentre `M` indica il tasto `Meta` `◆` (sulle moderne tastiere quest'ultimo tasto è scomparso, ma si può ottenere lo stesso risultato con `Alt` oppure premendo e rilasciando `Esc`). Nella notazione di queste scorciatoie, il trattino che separa due tasti in una combinazione di tasti indica che i tasti vanno premuti contemporaneamente. Così, quando si dirà che per eseguire un comando bisogna usare la combinazione `M-x`, si dovranno premere *contemporaneamente* `Alt` e `x`, oppure premere `Esc`, rilasciarlo e premere *successivamente* `x`.

Il manuale di Emacs (STALLMAN, 2012) può essere consultato dall'interno dello stesso programma con le combinazioni `C-h r` (premere contemporaneamente `Ctrl` e `h`), rilasciarli e premere successivamente `r`, oppure `C-h i d m Emacs RET` (premere contemporaneamente `Ctrl` e `h`),

rilasciarli e premere poi in successione `i`, `d`, `m`, scrivere successivamente Emacs, quindi premere `Invio`). Se si preferisce, si può consultare il manuale anche tramite il menu `Help` `»` `Read the Emacs Manual`.

Una delle caratteristiche principali di Emacs è il fatto di essere un editor di testi completamente personalizzabile ed espandibile, in modo da risultare sempre adatto alle esigenze dell'utente. Il linguaggio utilizzato per le espansioni è l'Elisp, un dialetto del Lisp sviluppato appositamente per Emacs. Esistono migliaia di pacchetti scritti in Emacs Lisp che permettono di estenderne ulteriormente le funzionalità.

La versione standard di Emacs, senza pacchetti aggiuntivi, fornisce già un discreto supporto alla creazione di documenti L^AT_EX, ma il pacchetto AU_CT_EX mette a disposizione numerosi strumenti che rendono senza alcun dubbio Emacs uno degli editor di testo più potenti in questo senso. In questa guida vedremo alcune delle funzioni principali di AU_CT_EX.

Per finire, prima di addentrarci nell'installazione e nell'utilizzo di Emacs, AU_CT_EX e di altri programmi accessori, è utile riportare qui qualche indicazione sul file di inizializzazione di Emacs, scritto in Elisp e di fondamentale importanza per modificare il comportamento delle funzioni del programma. Questo file, che sarà richiamato spesso in questo documento, presenta diversi nomi e localizzazioni a seconda del sistema operativo usato. Deve essere creato perché inizialmente non esiste ed in particolare:

- su Windows si chiama `init.el` e va messo in
 - `C:\Documents and Settings\nome utente\Application` (fino a XP);
 - `C:\Users\nome utente\AppData\.emacs.d\init.el` (da Vista in poi);
- su GNU/Linux e Mac OS si chiama `.emacs` o `init.el` e va messo in `~/.emacs`, `~/.emacs.el` oppure in `~/.emacs.d/init.el`.

Nelle prossime pagine ci riferiremo spesso al file di inizializzazione chiamandolo semplicemente `.emacs`, indipendentemente dalla posizione in cui si trova e dall'effettivo nome utilizzato nel proprio sistema (`.emacs`, `.emacs.el` oppure `init.el` che sia).

Si faccia attenzione quando si copia e incolla il codice da questo documento: potrebbe essere necessario riscrivere i caratteri " e ' in Emacs.

Emacs E AUCT_EX SU WINDOWS

2

(a cura di Orlando Iovino)

2.1 INSTALLARE Emacs

Per sistemi operativi Microsoft Windows non esiste una vera e propria procedura di installazione. Emacs si può utilizzare semplicemente procurandosi l'archivio che ne contiene gli eseguibili.

All'indirizzo <http://ftp.gnu.org/pub/gnu/emacs/windows/> si scarichi l'archivio compresso `emacs-24.2-bin-i386.zip` e lo si decomprima in una posizione di comodo; una buona soluzione, per esempio, è decomprimerlo in `C:\Programmi\Emacs`, che assumeremo da questo momento come cartella d'installazione.

Se si vuole creare un collegamento di Emacs nel menù Start, è sufficiente eseguire il programma `addpm.exe` contenuto nella cartella `C:\Programmi\Emacs\bin`.

A questo punto Emacs è pronto per scrivere documenti in L^AT_EX: l'utente infatti non deve specificare nessun percorso degli eseguibili della distribuzione L^AT_EX installata perché Emacs li trova da solo.

2.2 INSTALLARE AUCT_EX

Si scarichi l'archivio compresso `auctex-11.87-e24.2-msw.zip` dall'indirizzo <http://www.gnu.org/software/auctex/download-for-windows> e lo si decomprima in `C:\Programmi\Emacs`. Si noti che alcuni file sono gli stessi, perciò basta unire il contenuto dell'archivio di AUCT_EX con quello di Emacs.

Dopo aver installato AUCT_EX, Emacs caricherà da solo i file necessari per attivare la modalità `latex-mode`, propria di AUCT_EX, quando si apre o si crea un file con estensione `.tex`.

2.3 CORRETTORE ORTOGRAFICO

Per usare il correttore ortografico in Emacs si deve installare il programma GNU Aspell disponibile all'indirizzo <http://aspell.net/win32/>. L'ultima versione stabile è la 0.50-3 del dicembre 2002.

Si devono installare, nell'ordine:

1. il programma `Aspell-0-50-3-3-Setup.exe`;
2. il dizionario precompilato per l'italiano `Aspell-it-0.50-2-3.exe`.

Infine rimane da eseguire qualche altra operazione:

1. Si deve copiare il percorso della cartella dove sono contenuti gli eseguibili di Aspell nella variabile d'ambiente `PATH`.
2. Si deve aggiungere al proprio file `.init` le seguenti righe:

```
(setq-default ispell-program-name "aspell")  
(setq-default ispell-extra-args '("--reverse"))  
(setq ispell-dictionary "italiano")
```

dove la prima e la terza sono quelle strettamente necessarie, mentre la seconda serve a risolvere gli eventuali problemi presentati dalle versioni precedenti del programma.

2.4 SINCRONIZZARE SORGENTE E ANTEPRIMA

Prima di attivare la sincronizzazione tra il testo sorgente e l'anteprima del documento, si spiega come risolvere il problema dei diritti di amministratore – qualora l'account utente ne sia fornito – per i sistemi operativi Vista e 7 (su XP non dovrebbero esserci problemi, mentre su Windows 8 non è stato possibile provarlo). Infatti, in tale situazione si potrebbero avere problemi con la cartella `server` del tipo

```
error: The directory '~/.emacs.d/server' is unsafe
```

Una possibile soluzione, riportata in SCHLOSSER (2012), è la seguente. Si apra il prompt dei comandi e ci si porti nella cartella che contiene a propria volta la cartella `server`, scrivendo nella riga di comando un'istruzione del tipo

```
cd C:/Users/<nome utente>/Roaming/.emacs.d
```

oppure

```
cd %APPDATA%/.emacs.d
```

e infine

```
takeown /f server
```

Il successo dell'operazione verrà segnalato con qualcosa come

```
OPERAZIONE RIUSCITA: il file o la cartella ...
```

Si può ora configurare Emacs per la sincronizzazione con il visualizzatore Sumatra PDF – l'unico in grado di supportarla nei sistemi Windows – scaricabile dal sito <http://blog.kowalczyk.info/software/sumatrapdf/free-pdf-reader.html>.


Per la ricerca diretta basta scrivere nel file d'inizializzazione .init

```
(setq TeX-source-correlate-method (quote syntex))
(setq TeX-source-correlate-mode t)
(setq TeX-source-correlate-start-server t)

(setq TeX-view-program-list
      '(("Sumatra PDF" ("\"C:/Program Files (x86)/\
SumatraPDF/SumatraPDF.exe\" -reuse-instance"
      (mode-io-correlate " -forward-search %b %n"
        " %o")))))

(setq TeX-view-program-selection
      '(((output-dvi style-pstricks) "dvips and start")
        (output-dvi "Yap")
        (output-pdf "Sumatra PDF")
        (output-html "start")))
```

Per la ricerca inversa, invece, si apra Sumatra PDF, si raggiunga il menù

 e si scriva

```
C:/Programmi/Emacs/bin/emacsclientw.exe --no-wait +%l "%f"
```

in Imposta la ricerca inversa via riga di comando, come si vede dalla figura 2.1.

Con queste impostazioni il visualizzatore predefinito per Emacs – ma non del sistema operativo – sarà Sumatra PDF. Per la ricerca diretta

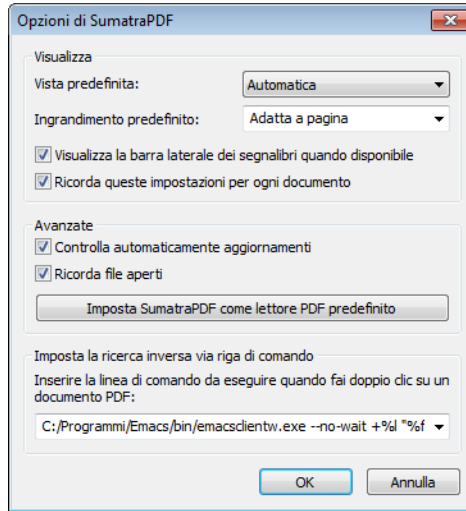


FIGURA 2.1 Impostazioni per il visualizzatore Sumatra PDF.

vanno premuti i tasti `C-c C-v` e il visualizzatore si aprirà (o porterà) in corrispondenza della riga del testo sorgente in cui si trova il cursore. Per la ricerca inversa, invece, basta un doppio clic nel visualizzatore.

Emacs E AUCTEX SU GNU/LINUX

3

(a cura di Mosè Giordano)

3.1 INSTALLARE Emacs

Emacs fa parte del progetto GNU, quindi è presente nei depositi di tutti i sistemi operativi della famiglia GNU/Linux. Se non fosse già installato, il metodo più semplice per ottenerlo è tramite il gestore di pacchetti della propria distribuzione (naturalmente, si possono utilizzare anche i gestori di pacchetti a interfaccia grafica). Riportiamo qui i comandi da terminale utili per installare il programma in alcune delle principali distribuzioni.

Su Debian e Ubuntu:

```
$ sudo apt-get install emacs
```

Su Fedora:

```
$ sudo yum install emacs
```

Su OpenSUSE:

```
$ sudo zypper install emacs
```

Il metodo più difficile, per i non avvezzi al terminale, consiste nel compilare Emacs a partire dal codice sorgente, ma non è intenzione di questa guida spiegare come farlo.

Dopo aver installato Emacs, lo si potrà avviare facendo clic sul suo *launcher* oppure eseguendo da terminale il comando

```
$ emacs
```

Se si desidera utilizzare Emacs con interfaccia testuale bisogna aggiungere al comando appena visto l'opzione `-nw` oppure `-no-window-system`:

```
$ emacs -nw
```

Come argomento da linea di comando si può aggiungere il nome del file o dei file che si vogliono modificare:

```
$ emacs file1.tex file2.tex
```

3.2 INSTALLARE AUCTION

Di solito anche il pacchetto AUCTION è presente nei depositi dei sistemi GNU/Linux. Come già detto per Emacs si consiglia di installarlo con il gestore di pacchetti della propria distribuzione. Ecco i comandi da usare nelle principali distribuzioni.

Su Debian e Ubuntu:

```
$ sudo apt-get install auctex
```

Su Fedora:

```
$ sudo yum install emacs-auctex
```

Su OpenSUSE:

```
$ sudo zypper install emacs-auctex
```

Sulle prime due il gestore potrebbe raccomandare di installare la TeX Live presente nei depositi ufficiali del sistema operativo. Per evitare che questo avvenga, per esempio perché si utilizza una TeX Live installata in altro modo, installando AUCTION da terminale è sufficiente aggiungere l'opzione `--no-install-recommends`:

```
$ sudo apt-get install --no-install-recommends auctex
```

3.3 CORRETTORE ORTOGRAFICO

Il correttore ortografico GNU Aspell può essere installato facilmente su GNU/Linux utilizzando, come al solito, il gestore di pacchetti della propria distribuzione. Il dizionario italiano di GNU Aspell per GNU/Linux si chiama `aspell-it`, quindi in Debian si installa da terminale con il comando

```
$ sudo apt-get install aspell-it
```

in Fedora con

```
$ sudo yum install aspell-it
```

mentre in openSUSE si può dare

```
$ sudo zypper install aspell-it
```

Emacs E AUCTION SU MAC OS

4

(a cura di Matteo Leccardi)




4.1 INSTALLARE Emacs

Sul mirror ufficiale <http://ftp.gnu.org/pub/gnu/emacs/> non è presente una versione di Emacs già compilata per Mac OS X, perché queste versioni sono rese disponibili da vari volontari come per esempio il gestore del sito <http://emacsformacosx.com/>. La procedura d'installazione è quella consueta: dopo aver scaricato la versione più recente, si apra il file `.dmg` (se non è già stato aperto automaticamente al termine del download) e si trascini l'icona di Emacs nella cartella **Applicazioni**.

In Mac OS le applicazioni avviate dall'interfaccia grafica non hanno accesso ai valori delle variabili d'ambiente, per cui bisogna rendere disponibile a Emacs la variabile `PATH` dando da terminale il comando

```
$ defaults write ~/.MacOSX/environment PATH "$PATH"
```

Si noti che le nuove impostazioni saranno effettive solo dopo aver eseguito un `logout` e un successivo `login`. Il comando va ripetuto ogni volta che si installa un programma che modifica il valore di `PATH`. (il caso che riguarda più da vicino i lettori di questa guida è la distribuzione `TeX`).

Per gli utenti che usano la tastiera italiana è utile impostare il tasto  come  e il tasto  per scrivere i caratteri speciali (in particolare parentesi quadre, graffe e tilde) scrivendo queste righe nel file `.emacs`:

```
(setq ns-command-modifier 'meta)
(setq ns-alternate-modifier nil)
```

4.2 INSTALLARE AUCTION

Come per Emacs, non è disponibile una versione di AUCTION già compilata per Mac OS, ma purtroppo in questo caso non ne esistono neppure versioni

non ufficiali. È quindi necessario utilizzare il programma `make`, che viene installato insieme all'ambiente di sviluppo `Xcode`. Chi non intende dedicare svariati GiB di disco a un software utile soltanto a chi sviluppa software può sfruttare il fatto che `make` viene installato anche dal correttore ortografico `cocoAspell`, descritto nel paragrafo 4.3. Ecco le istruzioni.

Dalla pagina relativa a Mac OS X, <http://www.gnu.org/software/auctex/download-for-macosx.html>, bisogna scaricare l'archivio compresso `auctex-11.87.tar.gz`. Se ne estragga poi il contenuto in una cartella temporanea, si apra una sessione di terminale e ci si porti nella cartella appena creata. Bisogna poi configurare AUCTION con il comando

```
$ ./configure \
--prefix=/Applications/Emacs.app/Contents/Resources/ \
--with-emacs=/Applications/Emacs.app/Contents/MacOS/Emacs \
--with-lispdir=/Applications/Emacs.app/Contents/Resources \
    /site-lisp/ \
--without-texmf-dir
```

sostituendo se necessario a `Applications`¹ la cartella in cui si trova Emacs. In questo modo tutti i file saranno contenuti nel bundle `Emacs.app`.

Infine si compili e si installi `make` dando

```
$ make && make install
```

Dopo l'installazione, bisogna impostare i visualizzatori per i vari file creati con L^AT_EX scrivendo nel file `.emacs` queste righe

```
(setq TeX-view-program-list
  '(("dvips and Skim" "%(o?)dvips %d -o &&\
  /Applications/Skim.app/Contents/SharedSupport/\
displayline %n %f %b")
    ("Skim" "/Applications/Skim.app/Contents/\
SharedSupport/displayline %n %o %b")
    ("open" "open %o")))
(setq TeX-view-program-selection
  '((output-dvi style-pstricks) "dvips and Skim")
    (output-dvi "Skim")
    (output-pdf "Skim")
    (output-html "open")))
```

¹Utilizzando il terminale non bisogna utilizzare i nomi localizzati delle cartelle ma quelli reali, quindi `Applications` è corretto anche se si utilizza la versione italiana di Mac OS.

Come visualizzatore si usa Skim, <http://skim-app.sourceforge.net/>, perché supporta la sincronizzazione tra sorgente e anteprima. La procedura di configurazione di Emacs e Skim è descritta nel paragrafo 4.4.

4.3 CORRETTORE ORTOGRAFICO

Anche per Mac OS X è disponibile una versione di GNU Aspell, chiamata cocoAspell, e disponibile all'indirizzo <http://cocoaspell.leuski.net/>.

Dopo aver installato la versione 2.1 (a oggi l'ultima versione) si possono scaricare i dizionari delle lingue che interessano direttamente dal sito di GNU Aspell, all'indirizzo <ftp://ftp.gnu.org/gnu/aspell/dict/>. Quello italiano, per esempio, si chiama [aspell6-it-2.2_20050523-0.tar.bz2](#).


A questo punto bisogna estrarre il contenuto degli archivi in una cartella temporanea, aprire una sessione di terminale e portarsi nella cartella appena creata. I dizionari si installano dando i comandi

```
$ ./configure
$ make
$ sudo make install
```

Infine si deve configurare Emacs aggiungendo al file `.emacs` le righe

```
(setq-default ispell-program-name "aspell")
(setq ispell-dictionary "italiano")
```

4.4 SINCRONIZZARE SORGENTE E ANTEPRIMA

L'unico visualizzatore di file PDF per Mac OS X associato a Emacs che supporti la sincronizzazione tra sorgente e anteprima è Skim (si veda il paragrafo 4.2) che una volta installato va configurato nel pannello  > Preferenze... > Sincronizza come segue e come mostra la figura 4.1:

- si spunti la casellina Controlla cambiamenti del file;
- si scriva nel campo Comando:
`/Applications/Emacs.app/Contents/MacOS/bin/emacsclient`
- si scriva nel campo Argomenti: `--no-wait +%line "%file"`

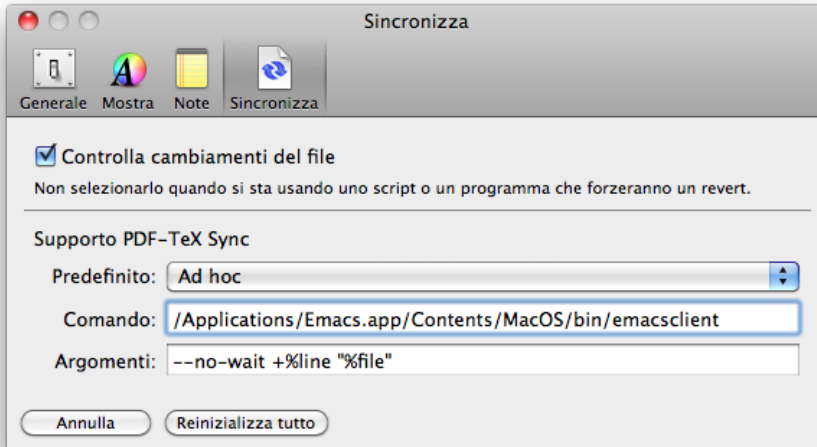


FIGURA 4.1 Preferenze di Skim per sincronizzare sorgente e anteprima.

La prima opzione fa in modo che quando un file aperto in Skim viene modificato da un altro processo, il programma chieda se il file modificato debba essere ricaricato: scegliendo **Automatico**, i successivi aggiornamenti verranno visualizzati senza chiedere ulteriori conferme.

Emacs si configura scrivendo nel file `.emacs` le righe

```
(setq TeX-source-correlate-method 'synctex)
(setq TeX-source-correlate-mode t)
(setq TeX-source-correlate-start-server t)
```

Si passa dal sorgente al PDF con la combinazione `C-c C-v`; si ottiene il contrario facendo clic sulla parola che interessa premendo contemporaneamente `⌘` e `⌘`.

Se alcune versioni di Emacs precedenti alla 23.3 lasciassero in primo piano la finestra del visualizzatore, si aggiungano queste righe al file `.emacs`

```
(defun ns-raise-emacs ()
  (ns-do-applescript "tell application \"Emacs\" to
    activate"))
(add-hook 'server-switch-hook 'ns-raise-emacs)
```

Nei precedenti capitoli si è spiegato come installare e configurare i principali componenti per usare \LaTeX con Emacs e \AUCTeX . In questo capitolo vedremo come si scrive in Emacs un semplice documento `.tex` e come usare il correttore ortografico.

5.1 CIAO MONDO

Ogni sistema operativo permette di aprire Emacs tramite un collegamento al programma principale, dopo di che sullo schermo dovrebbe comparire la finestra principale mostrata nella figura 5.1. Nel gergo di Emacs una finestra è chiamata *frame*. Ciascun *frame* può essere suddiviso in una o più *windows*, cioè lo spazio dove l'utente scrive, che è la zona di principale interazione con il programma. Un'altra componente fondamentale di un *frame* di Emacs, mostrata nella figura, è il *minibuffer*. Questo è il posto dove Emacs dialoga con l'utente mostrando messaggi di notifica oppure permettendo l'inserimento dell'input dei comandi eseguiti di volta in volta (DE BARI, 2006). Il testo che si sta modificando in Emacs è contenuto in un oggetto chiamato *buffer*. Quando si apre un file, il suo testo viene memorizzato in un buffer. In prima approssimazione possiamo dire che il *buffer* è il contenuto di un file, già esistente oppure che si sta creando.

Terminata questa breve carrellata terminologica, possiamo passare alla pratica. Per creare un nuovo documento ci sono due modi: raggiungendo il menù `File` `»` `Visit New File...` oppure premendo `C-x C-f`, dopo di che Emacs chiederà di aprire/trovare un file. Scrivendo un nome che nella cartella di lavoro non esiste questo file sarà creato. Dopo averlo nominato, per esempio `primo.tex`, si può cominciare a scrivere in \LaTeX . Per gli amanti delle scorciatoie da tastiera (autentico punto di forza di Emacs), scrivendo

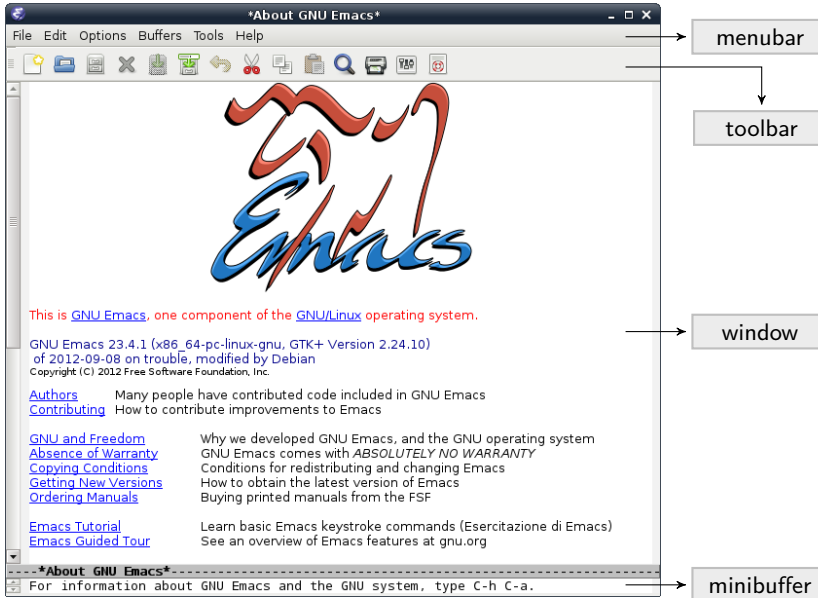


FIGURA 5.1 Un *frame* di GNU Emacs che contiene la classica schermata iniziale.

C-c C-e Emacs chiederà quale ambiente inserire. Il cursore si sposterà nel *minibuffer* e lì si potrà scrivere il nome dell'ambiente da inserire. Questo è il messaggio visualizzato:

```
Environment type: (default document)
```

In questo esempio, dato che non c'è ancora alcun documento attivo, l'ambiente *document* sarà proposto come quello predefinito. Dopo il primo RET apparirà nel minibuffer la scritta

```
Document class: (default article)
```

che permette di scegliere la classe che, per impostazione predefinita, è *article*. Si dia ancora RET e si vedrà

```
Options:
```

che permette di scegliere le opzioni della classe. Si faccia la scelta, si dia un terzo RET e si avrà infine il classico

```
\documentclass{article}

\begin{document}
< cursore >
\end{document}
```

con il cursore correttamente posizionato nell'ambiente *document*.

Come è naturale che sia, le macro \LaTeX possono essere inserite manualmente, ma \AUCTeX facilita notevolmente il loro inserimento. \AUCTeX , infatti, conosce la sintassi delle macro e degli ambienti di un centinaio dei pacchetti \LaTeX più diffusi. Per poter inserire una macro nel documento bisogna premere C-c RET e nel *minibuffer* sarà possibile inserire il nome della macro. Si può utilizzare TAB per l'autocompletamento. Inoltre conosce gli argomenti e le opzioni delle macro: infatti, dando C-c RET `frac` RET nel documento si otterrà `\frac{}{}` e il cursore si posizionerà all'interno del primo paio di parentesi graffe. Analogamente, con C-c RET `sqrt` RET verrà richiesto l'ordine *n* della radice da inserire (premere direttamente RET per non inserire nulla) e si otterrà `\sqrt[n]{}`. Un altro esempio è la macro `\verb`: \AUCTeX chiederà di scegliere il carattere da utilizzare come delimitatore e poi di inserire il testo di argomento.

Per comporre il documento, si può

- * premere sul pulsante presente nella *toolbar*;
- * scegliere `Command` \gg `LaTeX`;
- * scrivere C-c C-c. In quest'ultimo caso nel minibuffer si vedrà

```
Command: (default LaTeX)
```

e si dovrà premere RET per lanciare effettivamente \LaTeX .

Durante la composizione del documento Emacs restituirà

```
Type 'C-c C-l' to display results of compilation.
```

e alla fine, se tutto è filato liscio senza errori, si avrà

```
LaTeX: successfully formatted {1} page
```

Con questa procedura si scrive un nuovo documento \LaTeX .

Senza nessuna impostazione personale, il procedimento appena descritto crea un file `.dvi`. Se con gli stessi comandi si vuole invece un file `.pdf`, si deve scrivere

```
(add-hook 'LaTeX-mode-hook 'TeX-PDF-mode)
```

nel file `.emacs`.

Per visualizzare il documento prodotto, sia esso un file `.dvi` o un `.pdf`, si può

- * premere sul pulsante presente nella *toolbar*;
- * raggiungere `Command` `View` e dare poi RET;
- * scrivere `C-c C-C RET` o `C-c C-v`.

Se Emacs e AUCTEX sono configurati per lavorare con la sincronizzazione, in tutti e tre i casi il visualizzatore dovrebbe aprirsi sulla riga corrispondente del file sorgente.

5.2 CORREZIONE ORTOGRAFICA

Dopo aver descritto le modalità di installazione dei correttori ortografici, in questo paragrafo specifichiamo come usarli, indipendentemente dal sistema operativo presente sulla propria macchina.

Per fare una revisione del documento che si sta scrivendo, bisogna dare `M-x ispell RET`: in questo modo il dizionario proporrà, in sequenza, la correzione dei termini che trova errati.

Un secondo modo per scrivere correttamente è attivare la modalità *flyspell* ('correzione al volo'). Se la si vuole solo sul documento corrente, basta andare in `Tools` `Spell Checking` `Automatic spell checking (Flyspell)`; se invece la si vuole sempre abilitata si deve aggiungere la riga

```
(add-hook 'LaTeX-mode-hook 'flyspell-mode)
```

al proprio file `.emacs`.

Quando il correttore rileva una parola potenzialmente non corretta, la colora (in genere) in rosso. Alcuni termini, però, vengono segnalati come scorretti anche quando non lo sono, ma solo perché non sono presenti nel dizionario: vi possono essere aggiunti portando il puntatore del mouse sulla parola in questione, premendo il tasto centrale e scegliendo `Save word`.

USO AVANZATO DI Emacs

6.1 UN DOCUMENTO PIÙ COMPLESSO

Per scrivere documenti piuttosto corposi come libri, manuali, tesi e così via, conviene suddividere il documento in un file principale, che chiameremo per esempio `master.tex`, e altri file nei quali scrivere il vero e proprio contenuto del lavoro, per esempio `capitolouno.tex` e `capitolodue.tex`.

Emacs, come altri editor di testi, permette di lanciare la composizione del documento direttamente dal file su cui si sta lavorando anche se questo non è il file principale, cioè il file `master.tex` che dovrà contenere, per definizione, qualcosa come

```
\documentclass{book}

\begin{document}

\input{capitolouno}
\input{capitolodue}

\end{document}

%%% Local Variables:
%%% mode: latex
%%% TeX-master: t
%%% End:
```

Si notano subito alcune righe commentate non presenti nel codice descritto nel paragrafo 5.1: sono le *variabili locali*. Per capire subito di che cosa si tratti, si pensi alle famose ‘righe magiche’ che si usano per esempio con T_EXworks (GREGORIO, 2011).

Le variabili locali sono metacommenti ignorati da $\text{L}^{\text{A}}\text{T}_{\text{E}}\text{X}$ – in quanto commenti – ma non da Emacs e servono per impostare alcune preferenze del documento in lavorazione. Esse si possono indicare alla fine del file tra le righe `%%% Local Variables:` e `%%% End:` oppure in un'unica riga tra `% -*-` e `-*-` all'inizio del file. Ci riferiremo al primo caso.

Prima di proseguire è bene ricordare che le due variabili di cui parleremo sono messe a disposizione da $\text{AUC}^{\text{T}}\text{E}^{\text{X}}$. La riga `%%% mode: latex` indica che stiamo scrivendo un documento $\text{L}^{\text{A}}\text{T}_{\text{E}}\text{X}$, mentre `%%% TeX-master: t` indica che il file in questione è il file principale su cui lanciare le composizioni.

I file `capitolouno.tex` e `capitolodue.tex`, invece, oltre al proprio contenuto conterranno alla fine le righe

```
%%% Local Variables:
%%% mode: latex
%%% TeX-master: "master"
%%% End:
```

in cui mediante la riga `%%% TeX-master: "master"` si specifica che la composizione deve essere lanciata sul file `master.tex`. La figura 6.1 mostra un documento complesso come quello appena descritto così come appare sullo schermo.

Per fare in modo che all'apertura di un nuovo documento ci venga chiesto se il file debba essere o no un file *master*, si deve scrivere

```
(setq-default TeX-master nil)
```

nel file `.emacs` (si veda il capitolo 7).

6.2 LE VARIABILI LOCALI

Abbiamo già accennato alle variabili locali nel paragrafo precedente. Ora vediamo la questione un po' più a fondo. Va detto subito che ogni volta che si aggiunge una variabile locale si deve ricaricare il file perché questa sia riconosciuta. Questo può essere fatto aprendo e chiudendo Emacs, oppure, più semplicemente, eseguendo `M-x revert-buffer RET yes RET`.

Si consiglia di impostare le variabili locali con la modalità di inserimento tra `%%% Local Variables:` e `%%% End:` alla fine del file. Alle già citate `mode: latex` e `TeX-master: t` se ne aggiungono diverse altre molto utili in un documento `.tex`. Alcune di queste sono messe a disposizione da Emacs e altre da $\text{AUC}^{\text{T}}\text{E}^{\text{X}}$.

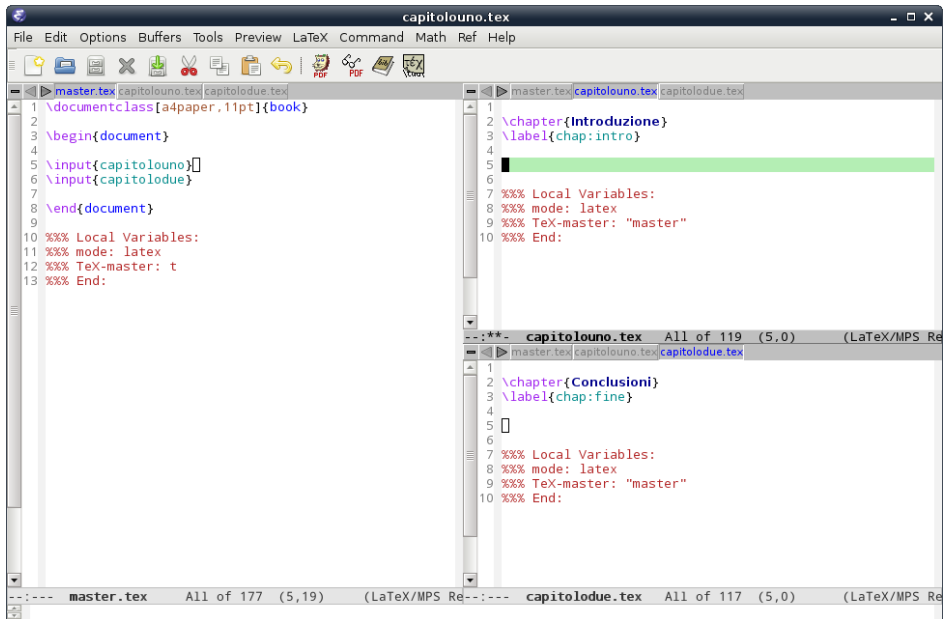



FIGURA 6.1 Lo schema di un documento complesso suddiviso in più file. Il *frame* è diviso in tre *windows*.

Se per esempio si vuole impostare la codifica del file sorgente a un particolare valore, Emacs mette a disposizione la variabile `coding`, che in questo documento è impostata come `coding: utf-8` per scrivere con la codifica Unicode a 8 bit. Un altro valore comune è per esempio `coding: latin-1`. Per vedere tutte le codifiche supportate da Emacs basta dare `M-x list-coding-systems`.

Tra le varie variabili locali di AUCTEX invece, segnaliamo qui quella che riguarda il motore di tipocomposizione. La variabile è `TeX-engine:`, che può essere impostata, per esempio, come `TeX-engine: xetex` per far sì che la sequenza `C-c C-c` lanci direttamente la composizione con X_YL^AT_EX.

Infine, se si è incerti sul nome della variabile da aggiungere e si ha timore di sbagliare, si può scrivere `M-x add-file-local-variable`, poi `RET` e quindi la variabile, aiutandosi magari con l'autocompletamento (tasto ).

6.3 SINOSI DELLE PRINCIPALI SCORCIATOIE

Emacs offre un'infinità di combinazioni di tasti per eseguire operazioni banali e meno banali, tanto che scriverle tutte sarebbe impossibile. Va da sé, dunque, che per usare il programma non serve ricordarle tutte, ma solo le principali, mostrate nella tabella 6.1, che con l'uso quotidiano saranno automaticamente memorizzate. Per approfondire il discorso si possono consultare anche le *Reference Card*, disponibili in genere nella propria cartella di installazione.

Tabella 6.1: *Scorciatoia* è la combinazione di tasti da premere per ottenere il risultato descritto nell'ultima colonna. La seconda colonna riporta la *funzione*, da eseguire con M-x, a cui è legata quella particolare combinazione di tasti.

SCORCIATOIA	FUNZIONE	EFFETTO
C-x C-f	find-file	Apri il file da specificare nel minibuffer
C-x C-s	save-buffer	Salva il buffer attuale
C-x C-w	write-file	Salva il file attuale nel file indicato nel minibuffer
C-x C-c	-	Chiede se salvare i buffer aperti e chiude Emacs
C-g	keyboard-quit	Interrompe il comando in esecuzione
C-x u	undo	Annulla l'ultima operazione
Ricerca e sostituisci		
C-s	isearch-forward	Ricerca incrementale in avanti
C-r	isearch-backward	Ricerca incrementale indietro
M-g g	goto-line	Sposta il cursore alla riga specificata nel minibuffer
M-%	query-replace	Ricerca e sostituzione

Continua nella prossima pagina

Continua dalla pagina precedente

SCORCIATOIA	FUNZIONE	EFFETTO
Copia e incolla		
M-w	kill-ring-save	Copia la regione selezionata
C-y	yank	Incolla l'ultima selezione
C-w	kill-region	Taglia la regione selezionata
C-k	kill-line	Taglia il resto della riga attuale
Aiuto		
C-h ?	help-for-help	Apre una finestra di aiuto
C-h k	describe-key	Mostra la documentazione della combinazione di tasti che viene successivamente premuta
C-h f	describe-function	Mostra la documentazione della funzione specificata nel minibuffer
Navigazione fra i buffer		
C-x b	switch-to-buffer	Cambia buffer visualizzato
C-x C-b	list-buffers	Mostra l'elenco dei buffer
C-x k	kill-buffer	Chiude il buffer specificato nel minibuffer
"Window"		
C-x 2	-	Divide la "window" attuale in due, verticalmente
C-x 3	-	Divide la "window" attuale in due, orizzontalmente
C-x o	other-window	Sposta il cursore nella "window" successiva

Continua nella prossima pagina

Continua dalla pagina precedente

SCORCIATOIA	FUNZIONE	EFFETTO
C-x 0	<code>delete-window</code>	Chiude la “window” attuale
C-x 1	-	Chiude tutte le “windows” tranne quella attuale

Si conclude dalla pagina precedente

PERSONALIZZARE Emacs E AUCTIONTEX

7

In questo capitolo vedremo alcune personalizzazioni utili quando si lavora con documenti L^AT_EX. Per modificare le impostazioni relative ad AUCTIONTEX si può dare `M-x customize-group RET AUCTIONTEX RET`. In questo modo si aprirà un buffer in cui modificare le opzioni desiderate tramite l'interfaccia.

Di seguito si suggeriscono alcune porzioni di codice Elisp per personalizzare Emacs, da inserire nel file di inizializzazione `.emacs`. Per rendere effettive le modifiche si ricordi di riavviare il programma.

Per fare in modo che AUCTIONTEX conosca i pacchetti L^AT_EX caricati, e di conseguenza fornire le funzioni di autocompletamento per le macro, bisogna permettergli di effettuare il parsing del documento aggiungendo nel proprio file `.emacs` il seguente codice:

```
(setq TeX-parse-self t) ; Attiva parsing al caricamento
(setq TeX-auto-save t) ; Attiva parsing al salvataggio
```

In questo modo verrà creata una cartella `auto` nella cartella di lavoro, in cui verranno registrate le informazioni sul proprio documento L^AT_EX (se invece non si desidera affollare inutilmente le proprie cartelle, non si attivino le due opzioni appena descritte).

AUCTIONTEX conosce le macro dei pacchetti principali, ma naturalmente non conosce *tutti* i pacchetti. Però gli si può far analizzare tutti i file di stile della propria distribuzione L^AT_EX dando `M-x TeX-auto-generate-global`. In questo modo verrà automaticamente aggiunto il supporto a tutti i pacchetti (tranne quelli troppo complicati e quelli basati su L^AT_EX₃). L'operazione può richiedere alcuni minuti, soprattutto alla prima esecuzione. Se un pacchetto non è stato correttamente analizzato da AUCTIONTEX, si possono aggiungere a mano le relative macro al momento di caricare il pacchetto in questione. Per esempio, uno dei pacchetti basati su L^AT_EX₃ è *siunitx*. Il seguente codice aggiunto al file `.emacs` informerà AUCTIONTEX che questo

pacchetto definisce i comandi `\SI`, `\si`, `\ang` e `\num`, dei quali il primo accetta due argomenti e gli ultimi tre solo uno:

```
(eval-after-load "tex"
  '(TeX-add-style-hook
    "siunitx"
    (lambda ()
      (TeX-add-symbols
        '("SI" 2)
        '("si" 1)
        '("ang" 1)
        '("num" 1))))))
```

Talvolta, in un documento L^AT_EX non si carica un certo pacchetto *B* poiché è automaticamente caricato da un altro pacchetto *A*, effettivamente presente nel preambolo. In questi casi, però, AUCTIONTEX potrebbe non rendersene conto e dunque non fornire l'autocompletamento delle macro del pacchetto *B*, anche se correttamente supportato. Supponiamo inoltre che insieme a *B*, il pacchetto *A* carichi anche i pacchetti *C* e *D*. Per ovviare a questo problema, si può inserire nel file d'inizializzazione il seguente codice:

```
(eval-after-load "tex"
  '(TeX-add-style-hook
    "A"
    (lambda ()
      (TeX-run-style-hooks "B" "C" "D"))))
```

Per esempio, il pacchetto *siunitx* carica automaticamente i pacchetti *array* e *amstext* e quindi non serve caricarli ulteriormente. Per farlo sapere ad AUCTIONTEX, possiamo completare il codice precedente come segue:

```
(eval-after-load "tex"
  '(TeX-add-style-hook
    "siunitx"
    (lambda ()
      (TeX-add-symbols
        '("SI" 2)
        '("si" 1)
        '("ang" 1)
        '("num" 1)
        (TeX-run-style-hooks "array" "amstext")))))
```

Se si è soliti suddividere i propri documenti L^AT_EX in più file da includere nel file principale con i comandi `\input` o `\include`, può essere utile aggiungere al file `.emacs`

```
(setq-default TeX-master nil)
```

In questo modo, all'apertura del documento Emacs chiederà qual è il file principale associato sul quale saranno eseguiti i comandi di composizione.

Se si usa spesso L^AT_EX per comporre documenti matematici, potrebbe essere utile attivare la modalità `LaTeX-math-mode` con `C-c ~` oppure con il comando `M-x LaTeX-math-mode`. In questo modo, per esempio, per inserire il simbolo `\alpha` si potrà usare la scorciatoia `' a`. L'elenco delle altre scorciatoie per i simboli può essere consultato dal menù `Math` nella barra dei menù. Per attivare automaticamente questa modalità ogni volta che si apre un documento L^AT_EX, bisogna aggiungere

```
(add-hook 'TeX-mode-hook 'LaTeX-math-mode)
```

al file `.emacs`.

Il codice

```
(setq TeX-electric-sub-and-superscript t)
```

aggiunto al file `.emacs`, permette di aggiungere automaticamente una coppia di parentesi graffe `{}` scrivendo i simboli `_` e `^` in modalità matematica.

Normalmente, la pressione di `RET` in un sorgente L^AT_EX aggiunge semplicemente una nuova riga. Se si vuole che la nuova riga sia anche automaticamente rientrata, bisogna premere `C-j`. Per ottenere il rientro con il semplice `RET`, basta aggiungere il seguente codice al file d'inizializzazione

```
(setq TeX-newline-function 'newline-and-indent)
```

Se si utilizza la funzione `reindent-then-newline-and-indent` al posto di `newline-and-indent` si avrà, in più, il rientro sulla riga corrente.

BIBLIOGRAFIA

- DE BARI, O. (2006). «GNU Emacs e AUCT_EX per L^AT_EX». *ArsT_EXnica*, anno n^o 2, pp. 60–64. <http://www.guitex.org/home/it/numero-2>.
- GREGORIO, E. (2011). «Introduzione a T_EXworks». <http://profs.sci.univr.it/~gregorio/introtexworks.pdf>.
- SCHLOSSER, J. (2012). «L^AT_EX – a complete setup for Windows». <http://www.latexbuch.de/files/latexsystem-en.pdf>.
- STALLMAN, R. (2012). *GNU Emacs Manual*. Free Software Foundation, Inc., 16^a edizione. <http://www.gnu.org/software/emacs/manual/emacs.pdf>.
- THORUP, K. K., ABRAHAMSEN, P., KASTRUP, D. *et al.* (2012). *AUCT_EX, A sophisticated T_EX environment for Emacs*. Free Software Foundation, Inc. <http://www.gnu.org/software/auctex/manual/auctex.pdf>.