

Numero 3
Aprile 2007

ArsT_EXnica

Rivista italiana di T_EX e L^AT_EX

GUIT

<http://www.guit.sssup.it/arstexnica>



G_UI_T – Gruppo Utilizzatori Italiani di T_EX

ArsT_EXnica è la pubblicazione ufficiale del G_UI_T

Comitato di Redazione

Massimiliano Dominici – *Direttore*

Claudio Beccari

Fabiano Busdraghi

Riccardo Campana

Massimo Caschili

Gustavo Cevolani

Andrea Fedeli

Enrico Gregorio

Lapo Mori

Gianluca Pignalberi

Ottavio Rizzo

Emiliano Vavassori

Emanuele Vicentini

Raffaele Vitolo

Emanuele Zannarini

ArsT_EXnica è la prima rivista italiana dedicata a T_EX, a L^AT_EX ed alla tipografia digitale. Lo scopo che la rivista si prefigge è quello di diventare uno dei principali canali italiani di diffusione di informazioni e conoscenze sul programma ideato quasi trent'anni fa da Donald Knuth.

Le uscite avranno, almeno inizialmente, cadenza semestrale e verranno pubblicate nei mesi di Aprile e Ottobre. In particolare, la seconda uscita dell'anno conterrà gli Atti del Convegno Annuale del G_UI_T, che si tiene in quel periodo.

La rivista è aperta al contributo di tutti coloro che vogliano partecipare con un proprio articolo. Questo dovrà essere inviato alla redazione di ArsT_EXnica, per essere sottoposto alla valutazione di recensori. È necessario che gli autori utilizzino la classe di documento ufficiale della rivista; l'autore troverà raccomandazioni e istruzioni più dettagliate all'interno del file di esempio (`.tex`). Tutto il materiale è reperibile all'indirizzo web della rivista.

Gli articoli potranno trattare di qualsiasi argomento inerente al mondo di T_EX e L^AT_EX e non dovranno necessariamente essere indirizzati ad un pubblico esperto. In particolare tutorials, rassegne e analisi comparate di pacchetti di uso comune, studi di applicazioni reali, saranno bene accettati, così come articoli riguardanti l'interazione con altre tecnologie correlate.

Di volta in volta verrà fissato, e reso pubblico sulla pagina web, un termine di scadenza per la presentazione degli articoli da pubblicare nel numero in preparazione della rivista. Tuttavia gli

articoli potranno essere inviati in qualsiasi momento e troveranno collocazione, eventualmente, nei numeri seguenti.

Chiunque, poi, volesse collaborare con la rivista a qualsiasi titolo (recensore, revisore di bozze, grafico, etc.) può contattare la redazione all'indirizzo:

arstexnica@sssup.it.

Nota sul Copyright

Il presente documento e il suo contenuto è distribuito con licenza © Creative Commons 2.0 di tipo “Non commerciale, non opere derivate. È possibile, riprodurre, distribuire, comunicare al pubblico, esporre al pubblico, rappresentare, eseguire o recitare il presente documento alle seguenti condizioni:

Ⓒ **Attribuzione:** devi riconoscere il contributo dell'autore originario.

Ⓓ **Non commerciale:** non puoi usare quest'opera per scopi commerciali.

Ⓔ **Non opere derivate:** non puoi alterare, trasformare o sviluppare quest'opera.

In occasione di ogni atto di riutilizzo o distribuzione, devi chiarire agli altri i termini della licenza di quest'opera; se ottieni il permesso dal titolare del diritto d'autore, è possibile rinunciare ad ognuna di queste condizioni.

Per maggiori informazioni:

<http://www.creativecommons.com>

Associarsi a G_UI_T

Fornire il tuo contributo a quest'iniziativa come membro, e non solo come semplice utente, è un presupposto fondamentale per aiutare la diffusione di T_EX e L^AT_EX anche nel nostro paese. L'adesione al Gruppo prevede un quota di iscrizione annuale di 25,00 € nel caso di persone fisiche o di 50,00 € nel caso di Enti o Istituzioni.

Indirizzi

Gruppo Utilizzatori Italiani di T_EX:

c/o Ufficio Statistica

Scuola Superiore Sant'Anna

Piazza Martiri della Libertà 33

56127 Pisa, Italia.

<http://www.guit.sssup.it>

guit@sssup.it

Redazione ArsT_EXnica:

<http://www.guit.sssup.it/arstexnica/>

arstexnica@sssup.it

Codice ISSN 1828-2369

Stampata in Italia

Pisa: 15 Aprile 2007

ArsT_EXnica

Rivista italiana di T_EX e L^AT_EX

Numero 3, Aprile 2007

Editoriale	
<i>Massimiliano Dominici</i>	3
Intervista a Frank Mittelbach	
<i>Frank Mittelbach, Gianluca Pignalberi, Dave Walden</i>	4
L ^A T _E X e la cesura delle parole in fin di riga	
<i>Claudio Beccari</i>	13
L ^A T _E Xpedia: il futuro della documentazione su L ^A T _E X	
<i>Lapo Filippo Mori</i>	21
Scrivere la tesi di laurea con L ^A T _E X 2 _ε	
<i>Lapo Filippo Mori</i>	27
La tipografia nel taschino	
<i>Salvatore Schirone</i>	46

Gruppo Utilizzatori Italiani di T_EX

Editoriale

Massimiliano Dominici

Il nuovo statuto del Gruppo Utilizzatori Italiani di T_EX

Nel corso dell'anno appena trascorso G_UI ha conosciuto una notevole crescita. Il numero degli iscritti si è ormai attestato su una cifra che si aggira attorno al centinaio. Tra i membri sono presenti, oltre a singoli individui, anche alcune istituzioni, per le quali è stato necessario pensare una nuova figura, con obblighi e diritti non necessariamente coincidenti con quelli dei soci tradizionali.

Questi cambiamenti hanno reso necessario adeguare gli strumenti di gestione del gruppo alle nuove esigenze. Il processo di revisione dello statuto, cominciato subito dopo l'ultimo convegno annuale e proseguito soprattutto nei primi mesi di questo anno, ha coinvolto l'intero *staff* del gruppo e si è concluso con l'approvazione della nuova versione.

Le novità sostanziali riguardano soprattutto i rapporti "istituzionali" tra soci *ordinari* e soci *amministrativi*. La nuova situazione, che vede un numero di iscritti più ampio che in passato, ha reso possibile, in particolare, il passaggio ad una forma di amministrazione del gruppo più partecipata.

Con il nuovo statuto i membri del Consiglio Direttivo (il vecchio *staff* amministrativo) vengono, infatti, eletti annualmente dall'Assemblea dei Soci e permangono in carica per tre anni. Il Consiglio Direttivo, elegge, tra i suoi membri aventi almeno un anno di anzianità, un Presidente (con compiti di rappresentanza legale e istituzionale e la facoltà di conferire incarichi speciali sulla base di esigenze che possono rendersi necessarie per il buon funzionamento del gruppo) un Vicepresidente e un Tesoriere. Anch'essi permangono in carica per un periodo di tre anni.

L'Assemblea dei Soci ratifica inoltre il bilancio, che viene presentato annualmente dal Tesoriere.

I soci potranno consultare online, sul sito del gruppo, una copia del nuovo statuto, per avere una visione più dettagliata dei singoli articoli.

Gli articoli di questo numero

Come promesso sul primo numero, in questa terza uscita della rivista proponiamo una nuova intervista di Gianluca Pignalberi. Se la prima era giustamente dedicata al "padre" del T_EX, Donald E. Knuth, in questa seconda intervista Gianluca Pignalberi, insieme all'altro intervistatore, Dave Walden, dialoga con Frank Mittelbach, componente storico del *L^AT_EX Project Team*, che si occupa della manutenzione del *kernel* di L^AT_EX 2_ε e dello sviluppo di L^AT_EX 3.

Il secondo articolo, di Claudio Beccari, è dedicato ad un argomento spesso sottovalutato, ovvero il controllo sulle modalità con le quali L^AT_EX divide le parole in fin di riga. L'articolo, oltre ad indicare come risolvere le più frequenti fonti di errore, è particolarmente indicato a chi volesse approfondire la conoscenza dei meccanismi interni utilizzati da T_EX per decidere in quale punto è possibile spezzare un vocabolo.

Seguono due articoli di Lapo Mori. Nel primo, apparso in traduzione inglese anche sul PracT_EX Journal, l'autore presenta un progetto di documentazione per L^AT_EX, basato su una piattaforma "wiki" e volto a risolvere i problemi che affliggono, da una parte, la tipologia cartacea (rapida obsolescenza, ecc.), dall'altra la tipologia elettronica (scarsa sistematicità, ecc.).

Il secondo articolo è la riproposta, in versione aggiornata e corretta, di un intervento presentato dallo stesso Lapo Mori nel corso del secondo convegno annuale del G_UI, tenutosi nell'ottobre del 2005. Trattandosi di un argomento (la stesura di una tesi di laurea con L^AT_EX) i cui risvolti pratici incontrano un notevole interesse in un utente medio di T_EX, è sembrato opportuno, anche in seguito all'ulteriore esperienza maturata nel periodo trascorso e all'interazione con gli utenti che hanno formulato nuovi interrogativi e suggerito nuove soluzioni, ospitare sulla rivista la nuova versione di questo prezioso tutorial.

Lo sviluppo di T_EX, e la sua diffusione, non dipendono soltanto dalla stesura di *macro* per svolgere nuovi compiti, o gli stessi compiti in maniera più efficiente, ma anche dall'implementazione di distribuzioni pensate per permettere l'uso del programma su un particolare sistema operativo; oppure, come nel caso di *oP_s4G_UI*, la distribuzione creata da Salvatore Schirone, per avere il proprio sistema tipografico sempre a portata di mano. Nell'ultimo articolo l'autore ci illustra questa distribuzione derivata da MiK_T_EX e "portabilizzata", ovvero adeguatamente modificata in modo da poter essere avviata da una *PenDrive* USB su qualsiasi computer avente sistema operativo Windows (9x, ME, XP).

Infine, a partire da questo numero, *A_rsT_EXnica* verrà stampata su carta riciclata. E, da parte nostra, un doveroso contributo al rispetto dell'ambiente.

Buona lettura.

▷ Massimiliano Dominici
mlgdominici@interfree.it

Intervista a Frank Mittelbach

Frank Mittelbach, Gianluca Pignalberi, Dave Walden

Sommario

Sia a Free Software Magazine (FSM, <http://www.freesoftwaremagazine.com>) che al T_EX Users Group (TUG, <http://www.tug.org>) piace pubblicare interviste. Recentemente, Gianluca Pignalberi di FSM e Dave Walden del TUG hanno entrambi contattato Frank Mittelbach per intervistarlo. Anziché fare due interviste separate, Mittelbach, Pignalberi e Walden si sono accordati per un'intervista congiunta, in modo da soddisfare gli interessi comuni già condivisi da FSM e TUG.

Abstract

Free Software Magazine (FSM, <http://www.freesoftwaremagazine.com>) and the T_EX Users Group (TUG (<http://www.tug.org/>)) both like to publish interviews. Recently, Gianluca Pignalberi of FSM and Dave Walden of TUG both approached Frank Mittelbach about interviewing him. Rather than doing two separate interviews, Mittelbach, Pignalberi, and Walden decided on a combined interview in keeping with the mutual interests already shared by FSM and TUG.

Dave Walden, cointervistatore: Frank, per piacere, inizia parlandoci un po' di te e di come sei entrato in contatto con L^AT_EX.

Frank Mittelbach, intervistato: Vivo con la mia famiglia a Magonza (Germania) dai primi anni Ottanta, cioè per la maggior parte della mia vita, finora. Oltre al mio hobby principale (l'arte tipografica), che può veramente essere definito il mio secondo lavoro, mi piace giocare a bei giochi da tavola, ascoltare musica jazz e leggere (principalmente letteratura inglese). Professionalmente lavoro per la Electronic Data Systems, dove attualmente sono responsabile della concezione e dell'implementazione di controlli remoti e gestione di reti e sistemi distribuiti.

Quando ero studente di matematica e informatica all'Università Gutenberg di Magonza sono entrato in contatto prima con T_EX e poi con L^AT_EX; questo ha contribuito al mio interesse per la composizione tipografica ed in particolare alla ricerca sugli algoritmi per la composizione automatica di alta qualità.

Durante i miei giorni da studente, negli anni Ottanta, un amico riportò dall'università di Stanford un nastro di sorgenti contenente qualcosa come T_EX 1.1 e notizie affascinanti sulla qualità del programma (fino ad allora scrivevamo le nostre tesi battendole a macchina, incollando a mano i simbo-

li o, come nel caso di alcune sofisticate macchine da scrivere IBM, cambiando la "palla" ogni due secondi). Provò ad implementare il programma sul sistema Multics che avevamo all'università e ci riuscì: quella era probabilmente la prima, se non l'unica, implementazione di T_EX per questo sistema.

In questo modo fui introdotto a T_EX e $\mathcal{A}\mathcal{M}\mathcal{S}$ -T_EX e scrissi il mio primo articolo, ottenendo uno splendido risultato. L'unico inconveniente fu che

$$\sum_{i=1}^n x_i$$

il nastro di Stanford conteneva solo font Quasi Computer Modern alla risoluzione di 200 dpi (e senza METAFONT), e l'unica stampante grafica che avevamo a disposizione aveva una risoluzione di 72 dpi. Quindi l'output che ottenevamo era grande come la formula al centro di questo capoverso (o più grande): stupendo per tappezzare le pareti, ma non proprio adatto ad una tesi da consegnare. Come risultato, il mio amico ha dovuto scrivere la sua tesi nel modo tradizionale, nonostante i suoi sforzi.

Qualche tempo dopo mi fu chiesto dal mio dipartimento di installare una versione commerciale di T_EX sui nostri PC nuovi fiammanti e di tenere una serie di lezioni a studenti e professori su come usarlo. Questa distribuzione era provvista di L^AT_EX 2.08 e una copia con rilegatura mobile del manuale (che sarebbe diventato il libro di Leslie Lamport su L^AT_EX). L^AT_EX, confrontato a Plain T_EX, mi sembrava molto buono ma, purtroppo,



FIGURA 1: Frank Mittelbach

se usato per produrre un documento, si piantava durante il caricamento dello “stile di documento articolo” a causa di mancanza di memoria su quei PC. Così il mio approccio a $\text{\LaTeX}$ si fermò dopo la lettura del manuale, e fui costretto a sviluppare un mio pacchetto di macro che implementava concetti analoghi, richiedendo al contempo meno memoria. Un anno dopo fu possibile usare realmente $\text{\LaTeX}$ nel dipartimento, ed ho ritirato il mio pacchetto.

Questo esercizio iniziale mi ha dato una buona visione dei concetti e dei meccanismi interni di un sistema come $\text{\LaTeX}$ e mi ha in seguito permesso di criticare costruttivamente certi aspetti di $\text{\LaTeX}$ – cosa che forse ha spinto Leslie a passarmi lo sviluppo e il mantenimento di $\text{\LaTeX}$.

Gianluca Pignalberi, cointervistatore: Molti dei nostri lettori hanno familiarità con $\text{\LaTeX}$ ma, per quelli che non lo fossero, puoi presentarglielo?

FM: $\text{\LaTeX}$ è un sistema di composizione non interattivo che usa $\text{\TeX}$ o una delle sue varianti ($\epsilon\text{-TeX}$, $\text{PDF}\text{\TeX}$, Omega) come motore di composizione.

Il programma $\text{\TeX}$ (sviluppato dal prof. Donald Knuth all’inizio degli anni Ottanta) è un motore di composizione a basso livello programmabile, i cui concetti e algoritmi forniscono una conoscenza di microtipografia¹ della più alta qualità in tempi, come quelli attuali, in cui questa conoscenza è in lento declino per via del fatto che sempre più autori sono costretti a diventare grafici e tipografi di se stessi, pur senza un’adeguata formazione. $\text{\TeX}$ è in special modo famoso per il suo eccellente algoritmo di interruzione di riga [correlato alla sillabazione NdT] e per le sue capacità di composizione di formule matematiche, entrambi insuperati, nonostante la libera disponibilità del programma e dei suoi algoritmi da oltre vent’anni.

$\text{\LaTeX}$ è un pacchetto di macro, scritto per il motore $\text{\TeX}$, che permette all’utente di astrarsi dalle capacità di formattazione a basso livello di $\text{\TeX}$ fornendo interfacce di più alto livello, le quali permettono all’autore di etichettare il testo con etichette logiche invece che procedurali (ad esempio, specificando che qualcosa è una lista o un paragrafo, piuttosto che affermare che qualcosa dovrebbe essere composto con un carattere nero e dello spazio sopra e sotto). La vera trasformazione di un sorgente $\text{\LaTeX}$ in un documento finito viene fatta con l’aiuto di “fogli di stile” e accomodamenti di configurazione che permettono cambiamenti anche radicali dell’aspetto in maniera consistente, e senza toccare o cambiare il sorgente².

1. La microtipografia riguarda l’aspetto dettagliato di caratteri e spaziature, ad esempio la crenatura (allargamento o restrizione dello spazio) tra le lettere, la generazione e il posizionamento delle legature, le interruzioni di riga, ecc. Per contro, la macrotipografia riguarda strutture più grandi, come il disegno di testate, liste o pagine.

2. Be’, idealmente, ma si veda oltre.

Storicamente, $\text{\LaTeX}$ è stato largamente influenzato da un sistema chiamato Scribe (di Brian Reid). A sua volta, il concetto di etichetta logica di $\text{\LaTeX}$ è stato abbastanza influente per i DTD dell’HTML e dei vari SGML/XML, così come lo furono i suoi approcci per convertire queste etichette logiche in una rappresentazione visuale.

Una delle differenze tra $\text{\LaTeX}$ e molti altri approcci simili è che il linguaggio $\text{\LaTeX}$ è effettivamente a sviluppo comunitario: nuovi pacchetti che aumentano (o modificano) le etichette e le funzioni di composizione appaiono continuamente, così che ormai $\text{\LaTeX}$ offre soluzioni di composizione per quasi ogni argomento, pure tanto diversi quanto composizione di giochi (come scacchi, *go* o parole incrociate), formule chimiche o musica. Un’altra differenza importante è che, sebbene $\text{\LaTeX}$ abbia divulgato il concetto di etichetta logica, fornisce anche dei modi per regolare finemente i risultati (essenzialmente fornendo interfacce verso l’etichettatura procedurale), accettando il fatto che nessuna trasformazione automatica di etichette logiche in rappresentazioni visuali può risolvere automaticamente tutti i problemi generati da restrizioni fisiche del formato di output (come la larghezza di una riga di testo o la dimensione della pagina). Mentre in alcune applicazioni questa regolazione fine non aggiunge valore (come nella pubblicazione di contenuti di database, dove è richiesta la totale automazione), è cruciale per comporre libri e articoli di riviste di alta qualità.

GP: Quante persone fanno parte ufficialmente del $\text{\LaTeX}$ Project? E... come definiresti il “ $\text{\LaTeX}$ Project”?

FM: Il team del $\text{\LaTeX}$ Project è un gruppo di persone abbastanza ristretto (in lento cambiamento), che badano al kernel di $\text{\LaTeX}$ e ad un piccolo numero di pacchetti fondamentali che costituiscono una solida base per una grossa quantità di pacchetti e aggiunte in costante evoluzione. Fornire e sorvegliare un nucleo stabile è un compito importante (sebbene non condiviso da molti) per mantenere $\text{\LaTeX}$ vivo come linguaggio di interscambio di documenti. Tra i membri che hanno composto, o attualmente compongono, il team ricordiamo Javier Bezos, Johannes Braams, David Carlisle, Michael Downes, Denys Duchier, Robin Fairbairns, Morten Høgholm, Alan Jeffrey, Thomas Lotze, Chris Rowley, Rainer Schöpf, and Martin Schröder con vari gradi di coinvolgimento.

Storicamente il progetto è subentrato nel 1991 a Leslie Lamport per la manutenzione e lo sviluppo di $\text{\LaTeX}$ 2.09. Fino ad allora il sistema era stato diviso in diverse varianti incompatibili, che spesso proibivano la corretta elaborazione da parte di una variante di $\text{\LaTeX}$ di documenti creati con un altro $\text{\LaTeX}$, sebbene l’indipendenza dai sistemi fosse originariamente uno dei risultati importanti di $\text{\LaTeX}$ come linguaggio di documentazione per la

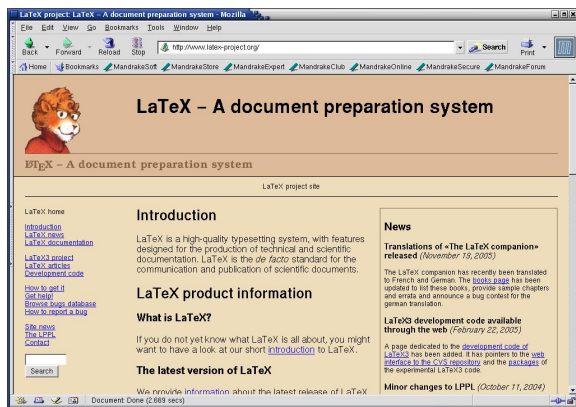


FIGURA 2: La home page del LaTeX Project

comunità scientifica. Un altro risultato per il team fu quello di correggere apparenti deficienze nei concetti di LaTeX 2.09. Il team ha trattato entrambe le questioni all'inizio degli anni Novanta con LaTeX 2_ε, che ha fornito una piattaforma stabile e consolidata in grado di fornire ulteriori possibilità di sviluppo al di fuori del codice del kernel.

Sebbene LaTeX 2_ε curasse la maggior parte, se non tutte, le deficienze trovate nel primo decennio di uso di LaTeX 2.09, si è originariamente pensato che LaTeX 2_ε sarebbe stato solo un passo intermedio verso LaTeX 3. Ma, nel tempo, è diventato sempre più chiaro che le rimanenti questioni aperte non sarebbero potute essere adeguatamente risolte entro i limiti imposti da: a) TeX come motore di composizione, e b) nessuna modifica ai concetti fondamentali realizzati dentro LaTeX.

Come risultato, la maggior parte degli sforzi fatti negli anni recenti dai membri del team di LaTeX è stata volta a ricercare le caratteristiche desiderabili per i motori di formattazione sottostanti e a sviluppare linguaggi e concetti sperimentali per un'interfaccia alla composizione tipografica per i grafici, livello di astrazione largamente mancante dall'attuale LaTeX (che attualmente richiede spesso la programmazione in TeX).

Dunque, attualmente, una definizione del LaTeX Project potrebbe essere che questo lavora per fornire le fondamenta dei concetti e l'implementazione di base di un nuovo sistema di composizione basato sugli aspetti positivi di LaTeX 2_ε (ad esempio, etichette logiche, estensibilità) ma, a livello di codice, non necessariamente orientato a fornire compatibilità. A livello di sintassi dei documenti la situazione è, ovviamente, differente, poiché il riuso dei documenti più vecchi è senz'altro un obiettivo importante. Ma, anche là, la principale concentrazione sarà sulla chiarezza dei concetti; come risultato, la compatibilità potrebbe essere ristretta in determinati casi a fornire un supporto alla conversione automatica.

DW: Nel mondo di TeX hai lunghi precedenti di lavoro collaborativo, in special modo con Rainer

Schöpf, agli albori di LaTeX. Deve proprio piacerti il lavoro collaborativo. Raccontaci qualcosa sui tuoi approcci a questo metodo.

FM: In effetti mi piace collaborare; nel corso degli anni ho lavorato con successo con molte persone diverse (su vari argomenti e in differenti campi). Per me, il valore principale della collaborazione sta nella fase di sviluppo delle idee che, a mio parere, sono meglio elaborate nell'ambito di uno scambio aperto. La mia immagine mentale, in questo caso, è un gioco tipo il ping pong, che si sviluppa solo se si reagisce direttamente a ciò che pensa l'avversario e si prende la palla man mano che viene giocata. La gente che ha lavorato con me sa che mi piacciono le sessioni disegnate alla lavagna bianca (ho bisogno di visualizzare lo svolgersi del gioco) e i metodi di *brainstorming* e delle mappe mentali.

Ma sono anche fissato con i dettagli, e posso impiegare molte energie e sforzi per finire veramente qualcosa (per soddisfazione personale), quando lo considero meritevole. La collaborazione a quel livello – dopo che il lavoro di sviluppo di concetto e progetto è terminato e il molto lavoro sul dettaglio comincia – assume normalmente una delle due possibili forme: o mi relego il più possibile al ruolo di mentore e lascio lavorare gli altri alle implementazioni vere e proprie, o metto così tante energie in un determinato compito che queste superano di un fattore grande il coinvolgimento degli altri. Il mio motto è “Es gibt nichts Gutes, außer man tut es” (traduzione libera: Non verrà fatto niente di buono finché non lo fai di “buono buono”) di Erich Kästner che, almeno in tedesco [e in italiano *NdT*], fa rima.

Un sacco di collaborazione avviene necessariamente per email (a causa dei diversi paesi in cui si vive, ecc.), ma trovo estremamente prezioso interrompere questo metodo di lavoro, a intervalli irregolari, con incontri *de visu* per tirare fuori le idee e renderle concrete quel tanto che basta per andare avanti in semi-isolamento, con le sole email e/o telefonate come mezzo di comunicazione “diretta”. Ciò spiega anche perché la gran parte del lavoro fondamentale, associato a Rainer Schöpf e me, risale al tempo in cui eravamo entrambi studenti universitari e avevamo modo di avere regolarmente scambi di idee davanti ad una lavagna bianca (bevendo galloni di tè).

In generale ritengo che Frederick Brooks ha ragione quando nel libro *The Mythical Man-Month* BROOKS, JR (1995) sostiene che, per avviare un progetto software di successo, hai bisogno di una squadra di persone relativamente piccola e strutturata, che sia responsabile di decidere il progetto finale. Un grande “comitato” di progetto porta solo a cattivi risultati, per arrivare a troppi compromessi tra fazioni differenti, o per l'incorporazione di concetti progettuali incompatibili.

GP: $T_{\text{E}}X$ è diffusamente considerato il miglior sistema di composizione tipografica, ma i compositori professionisti sembrano preferire l'uso di software visuale commerciale. Perché consiglieresti loro di usare $\text{\LaTeX}$ invece di un altro sistema? O, al contrario, perché no?

FM: Non c'è dubbio che $T_{\text{E}}X$ abbia qualità superiori in un buon numero di campi rispetto ad altri sistemi di composizione (ad esempio, la qualità della divisione dei capoversi, o la presentazione di formule matematiche, ecc.), ma è stato progettato come un programma di elaborazione batch e ciò implica il non permettere l'interazione con l'utente. Nei sistemi di composizione WYSIWYG l'utente può fare correzioni visuali, che sono dunque mostrate istantaneamente, mentre con $T_{\text{E}}X$ (o $\text{\LaTeX}$) devi modificare il tuo sorgente e poi riprocessarlo e controllare che le tue correzioni abbiano prodotto l'effetto desiderato.

Specialmente nell'ultima fase della produzione di un libro, la tendenza di $T_{\text{E}}X$ a raggiungere l'alta qualità solo per mezzo di cambiamenti non immediatamente chiari può veramente essere un difetto anziché un pregio (almeno se non consideri il perché e adatti il tuo metodo di lavoro). Per esempio, a causa dell'ottimizzazione globale della suddivisione dei capoversi, la *rimozione* di una singola parola in un capoverso determinerà un completo riallestimento dell'intero capoverso. Ciò potrebbe, in realtà, rendere il capoverso addirittura una riga *più lungo*, se $T_{\text{E}}X$ decidesse che un posizionamento più lasco di tutte le righe produrrebbe la migliore soluzione possibile. Se ciò accadesse durante la correzione dei refusi all'ultimo minuto, in un documento altrimenti ben formattato, sarebbe una bella seccatura.

La maggior parte dei grafici e dei compositori professionisti sono abituati a lavorare visivamente con risposta e controllo immediati. Dunque, ad essi, sistemi come $\text{\LaTeX}$ sembrano difficili da maneggiare e non vedono benefici in questi modelli di lavoro estranei.

Se l'uso di $\text{\LaTeX}$ sia consigliabile o meno dipende fortemente dal lavoro che si ha sotto mano, e la risposta è sempre dipendente dal contesto. In breve, suggerirei di usare $\text{\LaTeX}$ o sistemi simili quando uno o più fattori tra i seguenti giochi un ruolo importante nell'ambito del lavoro:

- preferenza dell'utente a pensare per strutture logiche;
- allestimenti che richiedono consistenza;
- documenti che non hanno un allestimento interamente definito o saranno presentati parallelamente in diverse vesti;
- documenti che richiedono suddivisione in capoversi di alta qualità;
- documenti che contengono molta matematica;

- contenuto prodotto automaticamente (ad esempio, proveniente da database);
- materiale corposo.

D'altro canto, i seguenti fattori fanno pendere l'ago della bilancia verso l'uso di un sistema visuale (di buona qualità!):

- preferenza dell'utente a pensare per strutture visive;
- utente non del tutto a suo agio a lavorare con linguaggi di programmazione (un front-end di alto livello per $T_{\text{E}}X$, ad esempio $\text{\LaTeX}$ o $\text{ConT}_{\text{E}}X$, aiuta, ma...);
- allestimenti che richiedono più flessibilità visuale che consistenza (ad esempio, le intestazioni sono disegnate una per una in base agli oggetti vicini);
- allestimenti che richiedono che il testo scorra intorno a forme arbitrarie ($T_{\text{E}}X$ non è stato progettato per questo);
- allestimenti che cambiano la misura orizzontale da colonna a colonna;
- materiale non corposo.

Quello che realmente fa la differenza può variare in base alle circostanze. Per esempio, in *The $\text{\LaTeX}$ Companion* MITTELBACH e GOOSSENS (2004), con i suoi circa 1000 esempi nel testo, è stato un enorme vantaggio poter riallestire l'aspetto degli esempi senza modificarli individualmente.

DW: C'è qualche relazione tra TUG, DANTE, o qualsiasi altro $T_{\text{E}}X$ Users Group e il $\text{\LaTeX}$ Project, e come i vari gruppi di utenti potrebbero mantenere la vitalità di $T_{\text{E}}X$ e dei suoi derivati?

FM: La relazione tra il $\text{\LaTeX}$ Project e i $T_{\text{E}}X$ Users Group può probabilmente essere ben descritta come lasca e informale. Diverse attività del team di progetto sono state supportate, in un modo o nell'altro, da un gruppo di utenti (ad esempio, fornendo spazi di incontro ad un convegno). Comunque, la maggior parte delle attività del team ha contato su risorse diverse, come il supporto dello ZDV (il laboratorio di calcolo dell'Università Gutenberg di Magonza), pagamenti dei diritti di *The $\text{\LaTeX}$ Companion*, e in piccola parte contributi dei singoli utenti. La maggiore alleanza con un gruppo di utenti è stata probabilmente lo sviluppo di un modello per un insieme di caratteri matematici, per cui il gruppo di utenti francese ha finanziato uno studente per tre mesi per lavorare con me su questo argomento. Il risultato di questo lavoro ZIEGLER (1994) sta finalmente portando alcuni frutti, dato che è stato d'aiuto per lo sviluppo del font STIX.

Comunque, i gruppi di utenti sono estremamente importanti per progetti come il L^AT_EX Project, in quanto forniscono una struttura di ricerca per contatti e discussioni faccia a faccia in convegni e pubblicazioni su rivista. Non può essere sottovalutato questo aspetto di fornire un ambiente di ricerca. E, se il gruppo di utenti non fosse più in grado di fornirla, si potrebbe arrivare anche alla morte della comunità. Certamente mi piace (e sono sicuro piaccia anche ad altri) lo scambio fruttuoso che è possibile solo in questa struttura.

È difficile rispondere alla domanda su che cosa i gruppi di utenti possano fare a lungo termine per sostenere la vitalità di T_EX e i suoi derivati – sfortunatamente, non sono sicuro che gli stessi gruppi di utenti sopravvivano a lungo termine. Il ruolo delle organizzazioni di utenti è cambiato negli ultimi venti anni. All’inizio, solo ottenere un sistema T_EX installato era lo sforzo principale, e i gruppi di utenti erano formati da persone interessate ad aiutarsi reciprocamente, a scambiarsi conoscenze e idee, e a supportare lo sforzo di sviluppo. Allora il ruolo dei gruppi di utenti era piuttosto chiaro e il beneficio per ogni membro era immediatamente visibile, ad esempio, ottenere informazioni altrimenti non disponibili, avere aiuto, ecc.

Col passare del tempo, l’accesso alle installazioni T_EX e ai suoi accessori (come pacchetti L^AT_EX, ecc.) è drasticamente cambiato – le moderne installazioni sono preimpacchettate, il software è disponibile in grandi archivi, e c’è molta documentazione disponibile. Come risultato, il tipico utente T_EX/L^AT_EX non ha bisogno di capire i meccanismi sottostanti e (sfortunatamente) non è più interessato a partecipare al loro sviluppo – gli utenti sono cambiati molto: da membri di un gruppo di persone con interessi simili che partecipano attivamente, eccitate dalla possibilità di fare composizione tipografica di alta qualità, a utenti finali di un “prodotto finito” che diventano molto tristi se il prodotto non fa esattamente ciò che gli utenti vogliono che faccia. Per questo nuovo tipo di utenti i gruppi di utenti non giocano un ruolo importante poichè, in questo momento, i gruppi di utenti non hanno risorse per aiutare fattivamente le persone nei loro problemi.

Ciò è in qualche modo ironico, dato che sono stati perlopiù i membri dei gruppi di utenti ad iniziare tutti quei cambiamenti che ora sembrano condurre alla fine della missione dei gruppi di utenti e della ragione di esistere agli occhi dell’utente tipo.

In teoria, penso che il modo migliore in cui i gruppi di utenti possano aiutare attualmente sia lavorare nelle seguenti aree:

- assumere e fornire le risorse che mantengano il “prodotto” vivo e ben mantenuto;
- fornire un forum adatto per la comunità di sviluppo;
- ottenere e gestire fondi di ricerca;

- attrarre nuovi utenti per allargare la base.

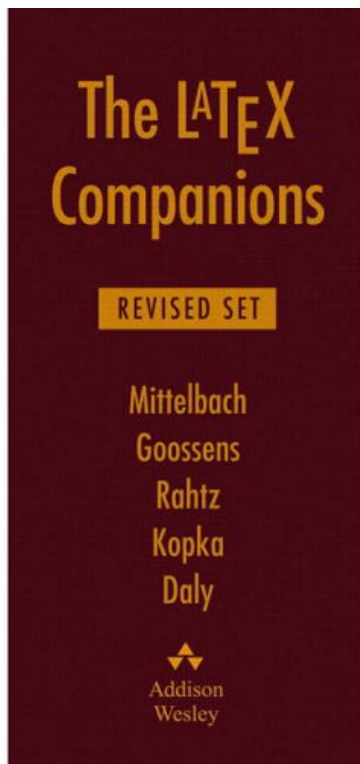
Sfortunatamente, nessuno di questi obiettivi è facilmente raggiungibile. Ci sarebbe bisogno di capitali (e risorse) consistenti, oltre quelli già a disposizione dei gruppi, e non è chiaro come ciò, come un charter, potrebbe attrarre abbastanza membri nuovi, che potrebbero dunque dividersi quei costi. Dopotutto, alla maggior parte delle persone il “prodotto” e il suo supporto sembra essere disponibile gratis, così, il pensiero degli utenti attuali è: “Perché pagare una (sostanziale) quota associativa periodica, quando tutto quello di cui si ha bisogno è di collegarsi a Internet e porre una domanda su `comp.text.tex` o scaricare qualche software dal CTAN? Io non vado a quei convegni a cui ‘loro’ partecipano, dunque, perché dovrei ‘finanziarli?’” Sarebbe necessario contrastare questo pensiero e far capire alla gente perché i gruppi di utenti sono, nonostante tutto, benefici per loro; ma, sfortunatamente, molte persone approfittano di un pasto gratis se possono, senza considerare le conseguenze.

GP: Il tuo progetto ha generato una licenza: la L^AT_EX Project Public License. Grazie alle ultime modifiche ad essa, L^AT_EX può veramente essere considerato un vero software libero. Come questo fatto ha incrementato la diffusione di questo strumento?

FM: Ad essere onesti, ho sempre considerato e tuttora considero L^AT_EX *vero software libero*, indipendentemente dalla licenza sotto cui era distribuito in passato o è distribuito ora. Il termine “libero” ha significati decisamente diversi per persone diverse, e non sono necessariamente d’accordo con l’idea di alcune persone che la loro libertà di poter cambiare arbitrariamente le cose senza restrizioni dovrebbe essere considerata un bene più importante del diritto di altri ad ottenere quello che si aspettano quando usano un certo prodotto.

L^AT_EX non è semplicemente un prodotto per un singolo utente, ma un linguaggio usato per la comunicazione di informazioni e uno dei punti più importanti è che permette di elaborare un documento in posti diversi con risultati identici, a patto che sia usata la stessa versione di L^AT_EX. Questa è una caratteristica su cui la gran parte della comunità fa affidamento, così la LPPL (L^AT_EX Project Public License) originale L^AT_EX3 PROJECT TEAM (2003) diceva, in breve: fate qualunque cosa vogliate del file X ma, se lo cambiate, (per esempio modificate il suo comportamento nel sistema) cambiategli nome in qualcos’altro, cosicché le persone che si affidano alla caratteristica della comunicazione di L^AT_EX non siano affette dalle vostre modifiche.

Tecnicamente, questo permetteva qualunque modifica e cambiamento desiderati, ma dava agli utenti di L^AT_EX la scelta informata se usare o meno una versione modificata con i propri documenti. In alcuni casi, ciò avrebbe comportato alcuni sforzi

FIGURA 3: The L^AT_EX Companions: il box set definitivo

extra alle persone impegnate nelle modifiche, ma nel complesso ritengo che fornisca una buona via di mezzo tra chi pensa che “libero” significhi il proprio diritto di cambiare a piacimento e chi pensa di avere diritto ad un mezzo di comunicazione di informazioni affidabile.

Comunque, alcuni sviluppatori nella comunità del software libero pensano che questa semplice regola restringa troppo i loro diritti (non potendo cambiare le cose a piacimento, inclusi modi che nascondano le modifiche agli utenti finali – pur non essendo questa l’intenzione) e così è iniziata una discussione sul fatto o meno che questa regola renda il software non libero – il principale ostacolo per molti era il requisito di cambiare i nomi in caso di cambiamento dei contenuti. Chiaramente, questo requisito è abbastanza diverso da quelli imposti dalla maggior parte delle licenze di software libero, ma queste licenze sono state scritte con in mente un modello di software abbastanza diverso (quello in cui l’attenzione è rivolta a componenti di software individuali, in cui le differenze tra versioni usate in posti diversi non inficiano l’usabilità del software)³. Ma, poiché non eravamo interessati ad imporre un cambiamento di nome fine a se stesso (anche se pensiamo ancora che medi adeguatamente tra i diversi bisogni), ho intavolato una lunga discussione con i legali di Debian e, come risul-

3. Per una discussione sul perché riteniamo che qualcosa come la GPL non sia un buon modello di licenza per i linguaggi liberi, che è uno degli aspetti di L^AT_EX, si veda L^AT_EX3 PROJECT TEAM (1995).

tato, abbiamo tirato fuori una nuova licenza che ha ammorbidito tali requisiti, preservando ancora il bisogno di stabilità e affidabilità della comunità. Essenzialmente, ritengo che la nuova licenza sia migliore in molte parti (e sono molto grato ad alcune persone dello staff legale di Debian per averci aiutato ad ottenere il miglioramento), ma è anche, forse inutilmente, più complessa di come sarebbe potuta essere in altre parti. Nel mondo di T_EX, la licenza originale tentava di codificare quello che era il comportamento standard e accettato, cioè, quando modificavi o miglioravi un pacchetto, lo chiamavi in modo diverso cosicché i vecchi documenti sarebbero stati compilati come atteso, mentre i nuovi potevano usare delle caratteristiche estese o cambiate, ed entrambi sarebbero potuti coesistere.

Così, la licenza ha incrementato la diffusione degli strumenti? Per come è concepito il mondo di T_EX, direi di no, dato che anche la licenza originale già codificava banalmente quello che la maggior parte della gente riteneva un buon modello per il software nel dominio di T_EX. Riferendoci al mondo più grande del software libero in generale, le modifiche hanno probabilmente aiutato la gente a capire che L^AT_EX e amici sono anche software “libero” e hanno fornito una base comune per capire che i diversi requisiti d’uso possono aver bisogno di diverse interpretazioni di “libero” per essere utili.

GP: Sei (sei stato, o sarai) coinvolto in altri progetti di software libero?

FM: La risposta dipende, probabilmente, dalla definizione dei termini “progetto” e “coinvolgimento”. Molti dei miei attuali interessi sono più di natura teorica e non necessariamente condurranno al software, o almeno non a software alla cui implementazione parteciperò direttamente; quei progetti a cui parteciperò più volentieri possono essere etichettati sotto il titolo generico di sviluppo di T_EX/L^AT_EX. Per esempio, un paio di settimane fa Hàn Thê Thành (lo sviluppatore di PDF_TE_X), Morten Høgholm e io abbiamo trascorso un produttivo fine settimana a casa mia a lavorare su alcune idee per la composizione a griglia⁴; quindi, sebbene non parteciperò all’implementazione, c’è e ci sarà coinvolgimento in progetti esterni a L^AT_EX. E, chissà, dato che nella mia vita professionale svolgo tutt’altro lavoro, forse anche quello condurrà un giorno verso un pacchetto di software libero o un altro in quell’area.

DW: Tu, più di altri, hai scritto circa il bisogno di andare oltre le limitazioni di T_EX, suggerendo approcci avanzati. Nella tua biografia su The L^AT_EX Companion MITTELBACH e GOOSSENS (2004) dici di voler lavorare all’accorpamento di estensioni come Omega ed ϵ -T_EX come base per un reale L^AT_EX3.

4. La composizione a griglia descrive allestimenti basati su un’invisibile griglia sottostante, che vincola il posizionamento (e la dimensione) degli oggetti.

Ovviamente, hai un percorso record nel realizzare progetti TEXosi grandi e complessi. Intravedi di venire coinvolto in qualcosa come il progetto exTEX o nell'inizio del tuo progetto di implementazione a basso livello per un miglioramento di TEX?

FM: Quando ho scritto *E-TEX: Guidelines for future TEX extensions* MITTELBACH (1990) nel 1990, il tempo non era ancora maturo per migliorare TEX e molte persone hanno considerato veramente un affronto a Don il fatto che avessi suggerito che ci potesse essere qualcosa da migliorare nel suo prodotto (ricordo, per esempio, pubbliche fantasterie circa strane teorie di un oscuro e sconosciuto tipografo tedesco. Be', quelle "strane teorie" mi sono state suggerite da nessun altro che Hermann Zapf che, sebbene tedesco, non può essere precisamente chiamato sconosciuto, meno che mai oscuro). Comunque, sia come sia, quello che stavo sfidando in quell'articolo era la qualità di composizione a livello di microtipografia. In ogni caso, dato che TEX era tecnicamente molto meglio di qualunque altra cosa all'epoca, la mia sfida fu probabilmente prematura, e ci è voluto circa un decennio prima che fossero condotti i primi veri esperimenti a quel livello e le cose avanzassero in quel campo (per gran parte con lo sviluppo di versioni sperimentali di PDFTEX, ma anche codice sperimentale di altri, come Matthias Clasen).

Discutendo dei miglioramenti a TEX bisogna distinguere tre aree fortemente disgiunte. La prima è l'area del linguaggio di programmazione e del fatto che questo linguaggio è incompleto e, per certi compiti, difficile da usare (o, come qualcuno afferma, una porcheria). A quel livello (senza divergere dai paradigmi fondamentali di TEX), estensioni come ϵ -TEX, e per certi versi Omega, hanno tentato di semplificare la programmazione fornendo le primitive mancanti che colmassero le ovvie lacune del linguaggio base. Ma, dato che tali funzionalità aggiuntive semplificavano solo la vita al programmatore senza migliorare realmente i risultati della composizione e le funzionalità di (quasi) tutte le nuove primitive potevano essere ottenute con qualche sforzo extra nel linguaggio base, abbiamo deciso di evitarle nello sviluppo di LATEX, dato che il loro uso avrebbe dato come risultato una versione di LATEX che avrebbe funzionato solo su una piccola frazione delle installazioni, senza alcun vantaggio significativo per l'utente. Il LATEX Project, insieme a persone di ConTEXt ed ϵ -TEX, si è veramente sforzato per produrre una definizione di sintassi arricchita per ϵ -TEX CARLISLE (1998) che pensavamo avrebbe fornito abbastanza benefici per passare ad un successore di TEX che implementasse questo set esteso. Tristemente, poco tempo dopo questa proposta, il lavoro su ϵ -TEX si è veramente interrotto e niente di tutto ciò fu mai implementato. Ciononostante, qualcosa è cambiato da allora: la base installata di installazioni

abilitate ad ϵ -TEX è cresciuta oltre la massa critica (largamente grazie a PDFTEX che includeva le estensioni ϵ -TEX) cosicché, qualche tempo fa, il LATEX Project ha annunciato ufficialmente che baserà le future versioni di LATEX su questo set esteso di primitive – ed ha recentemente iniziato a produrre realmente codice facente uso di di queste caratteristiche estese (sebbene ancora solo al di fuori del codice del kernel). In sostanza, non abbiamo mai voluto allontanarci dall'essere in grado di far girare LATEX appena spaccettato su una larga base di interpreti installati e abbiamo ritenuto ciò più importante di un linguaggio di programmazione potenzialmente più semplice o meglio ingegnerizzato che quasi nessuno potrebbe utilizzare⁵. Così, invece di provare semplicemente ad influenzare il linguaggio TEX estendendolo, io e alcuni altri nel progetto LATEX abbiamo lavorato anche dall'interno sviluppando il "linguaggio di programmazione sperimentale LATEX" CARLISLE *et al.* (1997). Questo è stato fatto su molti prototipi (il primo realizzato già nel 1993 o giù di lì). La versione corrente è qualcosa che pensiamo possa essere usato con successo ed abbiamo cominciato a fornire i primi pacchetti pubblici in questo linguaggio LATEX3 PROJECT TEAM.

La seconda area è quella che ha a che fare con i problemi di microtipografia, cioè quelli a cui ero principalmente interessato quando esponevo le deficienze di TEX in MITTELBACH (1990). In questa area il mio coinvolgimento era largamente confinato a dare inizio al lavoro per altri.

La terza area è quella che si interessa delle questioni generalmente aperte e irrisolte della tipografia digitale, come i modelli per la rappresentazione del materiale logico MITTELBACH (1997) e contenuti visuali ROWLEY e MITTELBACH (1997), le trasformazioni tra rappresentazione visuale e logica mediante metodi automatici che forniscano comunque la più alta qualità in accordo ad una metrica definita MITTELBACH e ROWLEY (1992), per dare alcuni esempi. Parte di questa ricerca è capire e codificare le regole tipografiche e sviluppare concetti e algoritmi che possano essere guidati da regole parametrizzate, ad esempio per produrre un posizionamento mobile di alta qualità.

Intravedo di iniziare il mio personale progetto di implementazione a basso livello per migliorare TEX? Quasi certamente no, ma intravedo di essere (di nuovo) coinvolto negli sviluppi correntemente in atto e spero di unire alcuni di questi sviluppi. Se questo avverrà in un progetto come exTEX o

5. Ho imparato che questo è un fattore critico quando abbiamo provato ad introdurre LATEX 2 ϵ nel 1994, il quale richiedeva l'installazione dei font codificati T1 (per esempio, font contenenti caratteri con segni diacritici). Il passaggio al nuovo sistema è quasi fallito perché gli utenti statunitensi non vedevano alcun vantaggio in un sistema contenente tutti quei caratteri inutili, necessari solo agli europei. Abbastanza fortunatamente, LATEX 2 ϵ aveva altri benefici che hanno prevalso per tutti gli utenti, ma è stato per un pelo.

PDF $\TeX$ è decisamente irrilevante. In questo momento ci sono ancora molte questioni irrisolte ed è ancora tempo di esperimenti (che possono avvenire contemporaneamente in diversi progetti), ma un obiettivo importante sarebbe riunire i vari sviluppatori per parlarsi delle loro idee e dei concetti dietro le idee.

GP: Hai citato due sistemi operativi verso cui è stato portato $\LaTeX$ e sappiamo che gira su diversi OS liberi e non liberi (sia commerciali che no). Che tipo di sistemi e programmi usi principalmente? E perché?

FM: Per essere precisi, $\LaTeX$ è interpretato, dunque non è un software che abbia bisogno di essere portato verso alcun sistema; $\LaTeX$ gira ovunque $\TeX$ sia stato portato – e $\TeX$, per quanto ne so, è stato portato più o meno su tutti i sistemi operativi mai esistiti (con l’eccezione di qualcosa simile a Palm OS); ad esempio, io l’ho usato su Mainframe, VMS, Unix, Multics e Windows.

Uso sia sistemi operativi liberi che commerciali; molto dipende dall’ambiente e dal compito sotto mano. A casa uso principalmente Linux con VMware per accedere ad alcuni programmi disponibili solo per Windows. Sul mio portatile la situazione è inversa: lì gira XP nativo e uso cygwin per un ambiente a linea di comando decente con tutti i vantaggi di un buon sistema Unix. Il mio editor preferito è Emacs, che uso praticamente su ogni piattaforma. Mi piace strutturare le cose usando le mappe mentali e per questo l’unico programma veramente buono che ho trovato è commerciale e gira solo sotto Windows – è una delle ragioni per cui questi giorni uso Windows regolarmente⁶.

Nel mondo professionale, in cui mi guadagno da vivere, il sistema operativo predominante è Windows, e nel mondo dei server troverai varianti commerciali di Unix, ma anche un numero crescente di server Linux.

Quello che uso dipende molto dal lavoro sotto mano: per qualcuno, come la gestione di progetti, il mondo Windows offre semplicemente gli strumenti migliori; in altri, il software libero (che gira su OS liberi o commerciali) offre migliore qualità o caratteristiche altrimenti non disponibili. Esempi potrebbero essere Perl, Apache, CVS, subversion e altri.

DW: Ricordo Knuth dire che scrivere The $\TeX$ book ha portato a centinaia di modifiche a $\TeX$, perché era costretto a spiegare le cose al lettore e, quando non ci riusciva, cambiava il programma. Tu hai scritto una certa quantità di libri e sei stato uno sviluppatore importante. Il tuo lavoro come autore ha influenzato il tuo lavoro come sviluppatore, o viceversa?

6. Ho recentemente riprovato il programma libero *free-mind* ed è stato molto migliorato, per cui la cosa può cambiare ancora.

FM: Penso che Don abbia assolutamente ragione nel fare quest’affermazione: penso che sia estremamente prezioso combinare lo sviluppo del software (in realtà qualunque cosa) con lo scriverne in qualche forma. Provare a spiegare ad altri le funzioni e i concetti dietro una creazione aiuta molto a scoprire se qualcosa funzionerà in pratica o no. Se non puoi spiegarlo, o se la spiegazione risulta orribilmente complicata, allora c’è qualcosa di fundamentalmente sbagliato nella tua creazione e dovresti tornare al tavolo da disegno.

È molto importante qui non fermarsi alla semplice documentazione delle funzioni o delle voci dei menu (sebbene sia un inizio), ma provare realmente a documentare il ciclo d’uso e le ragioni per cui si farebbero le cose in un modo o in un altro. Abbastanza spesso (sia con software libero che commerciale), si trova solo documentazione rudimentale che racconta che esistono questa e quest’altra caratteristica, ma non spiega mai perché si vorrebbe usare quella caratteristica in prima istanza. Quel tipo di documentazione, sebbene necessaria, non ti aiuta a migliorare il tuo strumento (e, abbastanza spesso, sembra che tali caratteristiche siano state implementate solo perché erano facili da aggiungere, senza fornire alcun beneficio reale).

Quindi sì, documentare le idee e i flussi di lavoro è sempre stato per me parte integrante dello sviluppo e/o del miglioramento del mio software, così come del software altrui. In *The $\LaTeX$ Companion* MITTELBACH e GOOSSENS (2004), per esempio, buona parte di ciò che descrivo è software sviluppato da altri e il processo di provare a spiegare come usarlo e trovare buoni esempi d’uso porta, in molti casi, a miglioramenti nella sintassi o nelle caratteristiche dopo averne discusso con gli autori.

Di conseguenza, il mio consiglio agli sviluppatori è di provare sempre ad autodocumentare le proprie creazioni, o almeno trovare qualcuno che lo faccia per loro (partendo dall’inizio dello sviluppo!), e valutare le scoperte di questo processo: ne risulteranno notevoli miglioramenti al prodotto.

DW: Grazie molte, Frank, per aver trovato il tempo di fare quest’intervista con noi. I tuoi approfondimenti su $\TeX$, $\LaTeX$ e lo sviluppo e la diffusione di sistemi complessi in un ambiente di elaborazione distribuito sono affascinanti. Credo che i lettori dell’Angolo delle Interviste del TUG saranno d’accordo.

E grazie a te, Gianluca, per aver accettato di farmi condividere quest’intervista con te. È stato un piacere lavorare con te.

GP: Grazie, Frank, per aver dato ai lettori di Free Software Magazine un saggio molto ben spiegato su un importante pezzo di software libero. Inoltre, la tua spiegazione sulla LPPL ha chiarito alcuni punti oscuri di un precedente articolo PIGNALBERI (2005). E grazie a te, Dave, per avermi dato la

possibilità di fare quest'intervista congiunta, che è molto più interessante di un'intervista "normale".

FM: Grazie, Dave e Gianluca, per aver condotto quest'intervista nel modo in cui lo è stata. Mi è piaciuto vederla svilupparsi domanda dopo domanda; nonostante il tempo che ha richiesto (colpa mia) e il fatto che viviamo distanti, mi è sembrato di fare un'intervista faccia a faccia che, penso, sia come doveva essere.

Note e licenza

La presente intervista è apparsa in origine su TUG Interview's Corner e Free Software Magazine n° 12. L'articolo è © 2006 degli autori, ed è rilasciato con la seguente licenza: "Sono permesse la copia e la distribuzione *verbatim* del presente articolo con ogni mezzo e senza pagamento di diritti d'autore soltanto se la presente nota è parte integrante dell'articolo". Traduzione a cura di Gianluca Pignalberi.

Riferimenti bibliografici

L^AT_EX3 PROJECT TEAM. «The L^AT_EX Project CVS Repository». Disponibile su <http://www.latex-project.org/cgi-bin/cvswweb/>.

— (1995). «Modifying L^AT_EX». Disponibile con tutte le distribuzioni L^AT_EX.

— (2003). «The L^AT_EX Project Public License (Version 1.3)». Disponibile su <http://www.latex-project.org/lppl/>.

BROOKS, JR, F. P. (1995). *The Mythical Man-Month; Essays on Software Engineering*. Addison-Wesley, Boston, Massachusetts, 2^a edizione.

CARLISLE, D. (1998). «Notes on the Oldenburg ε-T_EX/L^AT_EX3/ConT_EXt meeting». Disponibile su <http://www.latex-project.org/papers/etex-meeting-notes.pdf>.

CARLISLE, D., ROWLEY, C. e MITTELBAACH, F. (1997). «The L^AT_EX3 Programming Language—a proposed system for T_EX macro programming». *TUGboat*, **18** (4), pp. 303–308. Disponibile su <http://www.tug.org/TUGboat/Articles/tb18-4/tb57rowl.pdf>.

MITTELBAACH, F. (1990). «E-T_EX: Guidelines for Future T_EX Extensions». *TUGboat*, **11** (3), pp. 337–345. Disponibile su <http://www.tug.org/TUGboat/Articles/tb11-3/tb29mitt.pdf>.

— (1997). «Language Information in Structured Documents: Markup and rendering—Concepts

and problems». In *International Symposium on Multilingual Information Processing*. Tsukuba, Japan, pp. 93–104. Articolo invitato. Ripubblicato su *TUGboat* 18(3):199–205, 1997. Disponibile su <http://www.tug.org/TUGboat/Articles/tb18-3/tb56lang.pdf>.

MITTELBAACH, F. e GOOSSENS, M. (2004). *The L^AT_EX Companion*. Tools and Techniques for Computer Typesetting. Addison-Wesley, Boston, Massachusetts, 2^a edizione. With Johannes Braams, David Carlisle, and Chris Rowley.

MITTELBAACH, F. e ROWLEY, C. (1992). «The pursuit of quality: How can automated typesetting achieve the highest standards of craft typography?» In *EP92—Proceedings of Electronic Publishing, '92, International Conference on Electronic Publishing, Document Manipulation, and Typography, Swiss Federal Institute of Technology, Lausanne, Switzerland, April 7–10, 1992*, a cura di C. VANOIRBEEK e G. CORAY. Cambridge University Press, New York, pp. 261–273.

PIGNALBERI, G. (2005). «The L^AT_EX Project Public License». *Free Software Magazine*, (7), pp. 52–54. Disponibile su http://www.freesoftwaremagazine.com/free_issues/issue_07/tex_license/.

ROWLEY, C. A. e MITTELBAACH, F. (1997). «Application-independent representation of multilingual text». In *Europe, Software + the Internet: Going Global with Unicode: Tenth International Unicode Conference, March 10–12, 1997, Mainz, Germany*, a cura di UNICODE CONSORTIUM. The Unicode Consortium, San Jose, CA. Disponibile su <http://www.latex-project.org/papers/unicode5.pdf>.

ZIEGLER, J. (1994). «Technical report on math font encoding (version 2)». Technical report, L^AT_EX3 project. Disponibile su CTAN: [info/1tx3pub/13d007.*](http://ctan.org/tex3pub/13d007.*).

- ▷ Frank Mittelbach
L^AT_EX Project
frank.mittelbach@latex-project.org
- ▷ Gianluca Pignalberi
Free Software Magazine
g.pignalberi@freesoftwaremagazine.com
- ▷ Dave Walden
TUG
dave@walden-family.com

L^AT_EX e la cesura delle parole in fin di riga

Claudio Beccari

Sommario

Questo articolo è destinato a spiegare come fa il sistema T_EX a comporre i capoversi, all'occorrenza dividendo in sillabe le parole in fin di riga.

Questo articolo serve per fornire le indicazioni necessarie per capire certi funzionamenti strani del compilatore L^AT_EX (in realtà dell'interprete T_EX) quando sembra rifiutarsi di spezzare le righe in modo adeguato. L'errore, se c'è, è sempre umano, ahimè, e deriva dall'insufficiente comprensione del funzionamento della macchina che in questo caso è T_EX.

Abstract

This tutorial explains how T_EX (the program) typesets paragraphs, possibly by hyphenating words at line breaks.

This tutorial should explain L^AT_EX's (actually T_EX's) strange behavior in certain circumstances when apparently refuses to correctly break lines. If there is some error, unfortunately this is always a human one, and it is due to an insufficient understanding of T_EX's procedures and algorithms.

1 Composizione dei capoversi

T_EX forma i capoversi leggendo dal file sorgente la sequenza di lettere, parole, spazi, comandi, eccetera, per formare una lunga riga che contiene tutto il capoverso¹. Nel momento in cui questa lunga riga è pronta, T_EX ha già eseguito tutte le macro e i comandi che riguardano il capoverso stesso; in particolare ha già eseguito le scelte dei font, l'inserzione dei caratteri, compresi quelli a cui il compositore accede per mezzo di comandi, ha inserito i richiami di nota e le espressioni matematiche in linea, eccetera.

Generalmente questa lunga riga va spezzata in righe della giustezza specificata per quel determinato capoverso, che potrebbe trovarsi nel testo principale, o dentro una lista, o dentro un `minipage` o una `\parbox`. Le giustezze cambiano quindi a seconda della posizione del capoverso, ma che la giustezza sia pari a `\textwidth`, a `\linewidth`, a `\columnwidth` o ad un altro valore non ha nessuna importanza; resta il fatto che la lunga riga deve

essere spezzata in parti più corte tutte della stessa lunghezza, tranne l'ultima riga del capoverso.

T_EX cerca di trovare dapprima le divisioni della riga in parti senza dividere in sillabe le parole in fin di riga, inserisce la gomma elastica negli spazi interparola e attribuisce un coefficiente di merito al risultato basandolo sul calcolo di quanto ha dovuto allargare o restringere ogni riga parziale: il coefficiente di merito, o meglio di demerito, si chiama *badness* o *bruttezza*. Se questa bruttezza è inferiore ad un certo valore specificato nel file di classe con cui si sta componendo (il parametro `\pretolerance`), bene, la divisione è fatta e il capoverso così formato dalle diverse righe parziali viene aggiunto alla pagina che si sta componendo, prima di attivare l'algoritmo per spezzare la pagina; via via che questa pagina va riempiendosi, viene determinata la sua "bruttezza" in base a quanto bisogna allargare o restringere la gomma verticale per raggiungere l'altezza della gabbia di composizione e poi, quando questa bruttezza è minima, si taglia via la parte superiore insieme alle note e agli oggetti flottanti di cui si è già tenuto conto e si accoda questo materiale al file DVI o PDF che si sta componendo.

Tuttavia, ritornando al capoverso, se nessuna divisione in righe parziali, fra le tante divisioni possibili, della lunga unica riga riesce a dare una bruttezza inferiore a `\pretolerance`, allora T_EX riprende la lunga unica riga e cerca le divisioni spezzando alcune parole in fin di riga; per ogni divisione in sillabe, cioè per ogni cesura eseguita in fin di riga, alla riga viene associata una penalità che fa aumentare la bruttezza. Se due righe consecutive hanno subito la cesura viene aggiunta un'altra penalità; se la penultima riga del capoverso ha subito la cesura si aggiunge ancora un'altra penalità. Di tutte queste penalità si tiene conto insieme ai coefficienti di allargamento o restringimento delle varie righe parziali per determinare la bruttezza complessiva per le varie possibili scelte delle cesure. Viene scelto quell'insieme di cesure che produce la minima bruttezza. Se questa scelta produce una bruttezza inferiore ad un secondo parametro `\tolerance` specificato nel file di classe, bene, il capoverso viene accodato così diviso in righe alla scatola 255, altrimenti T_EX fa un terzo tentativo con una tolleranza maggiore, ma comunque emette un messaggio di avvertimento segnalando la riga peggiore della prescelta divisione in righe e produce i vari messaggi di `overfull hbox` o di `underfull hbox`; le righe (`hbox`) troppo

1. Si ricorda che in italiano si chiama *capoverso* quello che in inglese si chiama *paragraph*; e in italiano si chiama *paragrafo* quello che in inglese si chiama *section*. Bisogna evitare di confondere i concetti.

piene fuoriescono dal margine destro e ne viene comunicato l'ammontare solo se supera la quantità specificata con il parametro `\hfuzz`. Le righe troppo vuote, cioè contenenti troppo spazio bianco, sono state riempite di colla che ha dovuto essere allungata troppo contribuendo in modo sostanziale alla bruttezza del capoverso; per queste righe viene comunicata la bruttezza, solo se è superiore al parametro `\hbadness` specificato nel file di classe. Generalmente `\hfuzz` è dell'ordine di grandezza di un punto tipografico, circa un terzo di millimetro. La bruttezza che passa senza messaggi sullo schermo è dell'ordine di qualche centinaio; una bruttezza di 10000 per T_EX significa "bruttezza infinita".

È evidente che per comporre bene T_EX deve avere le condizioni migliori per farlo; con righe più lunghe di 100mm la possibilità di dividere bene il capoverso è praticamente assicurata in ogni caso scrivendo in italiano, purché non si usino font della famiglia a spaziatura fissa, per i quali la cesura è interdetta. In italiano si riesce a comporre abbastanza bene fino a giustezze di circa 30mm, ma ovviamente la cosa dipende molto dal contenuto del capoverso; in altre lingue giustezze così piccole generano una quantità di avvertimenti di `overfull` e `underfull hbox`. È quindi chiaro che la capacità di giustificare bene le righe di un capoverso dipende molto dall'efficienza dell'algoritmo di divisione in sillabe oltre che dalle regole grammaticali di ogni lingua.

2 Le parole del file sorgente

Il programma T_EX non conosce la lingua con cui sta componendo; riesce a distinguere le parole con regole euristiche abbastanza semplici:

1. una parola è una sequenza di lettere; una lettera è un segno di un font al quale è stato attribuito il codice di categoria² 11 o 12 e al quale è associato un codice di lettera minuscola (ritorneremo su questi concetti fra poco);
2. la sequenza di lettere è composta tutta con lo stesso font;
3. la sequenza di lettere comincia dopo uno spazio (colla) o dopo un segno di categoria 12;
4. la sequenza di lettere finisce con il primo carattere o con il primo oggetto diverso da una lettera, sia esso uno spazio, un segno di interpunzione, un comando, una scatola, della matematica, o qualsiasi altra cosa diversa da una lettera;

2. Per una chiara spiegazione dei codici di categoria ci si può riferire all'articolo di GREGORIO (2006), oltre che all'immane *T_EXbook* di KNUTH (1984).

5. dopo la fine di quella che T_EX ha identificato come una sequenza di lettere, T_EX non ricomincia ad esaminare altre sequenze di oggetti (*token*, in gergo T_EX) finché non trova di nuovo un valido inizio di sequenza di lettere.

In generale non è così scontato che una sequenza di lettere che per noi è una parola lo sia anche per T_EX. Per esempio, la parola "elettricità" è completamente una parola se il font di uscita è codificato con la codifica T1³ o equivalente, ma la parola che T_EX riconosce è solamente "elettric" se sta lavorando con la codifica OT1, perché la rappresentazione interna della 'à' è data da "elett^{ri}cit\`a" ovvero "elett^{ri}cit\accent18a"; il comando per collocare l'accento termina la sequenza di lettere che T_EX crede che sia una parola. In definitiva con la codifica T1 T_EX riesce a trovare le possibili posizioni delle separazioni delle sillabe "e-let-tri-ci-tà", mentre con la codifica OT1 ne perde una e nel caso riesce a dividere la parola "elettricità" solo con la possibile divisione "e-let-tri-ci-tà".

In francese la cosa è ancora più critica; la parola "électricité" è divisibile in "é-lec-tri-ci-té" con la codifica T1, ma non è divisibile per niente con la codifica OT1, perché dopo il primo comando per l'accento, T_EX non trova un altro possibile inizio di parola se non con l'inizio della parola successiva.

Va notato che negli esempi riportati sopra si sono indicati i possibili punti di cesura secondo le regole grammaticali. In tipografia le regole di divisione devono rispettare sia le regole grammaticali, sia le regole di buona composizione tipografica, secondo la quale non si va mai a capo in fin di riga lasciando nella riga una sillaba troppo corta o rimandando alla riga successiva una sillaba troppo corta. Per ogni lingua è definito un valore per ciascuno dei due parametri `\lefthyphenmin` e `\righthyphenmin`; per l'inglese e il francese essi valgono rispettivamente 2 e 3; per l'italiano essi valgono 2 e 2; per il greco 1 e 1; questi numeri rappresentano il numero di lettere minimo accettabile per la sillaba a sinistra della prima cesura tipografica e, rispettivamente, a destra dell'ultima cesura tipografica. Di fatto, se avendo in vigore la codifica T1, si dessero i comandi

```
\showhyphens{elettricità}
\showhyphens{électricité}
```

3. Qui si dà per scontato che il lettore abbia un minimo di conoscenza delle codifiche OT1 e T1 per i caratteri latini che vengono usati dal sistema T_EX. La vecchia codifica OT1 corrisponde a font che contengono solo 128 segni (codifica a 7 bit) e quindi non può contenere le lettere accentate, ma produce l'uscita mediante comandi di basso livello che sovrappongono il segno dell'accento al segno della lettera. La codifica T1 (codifica a 8 bit) contiene 256 segni che comprendono la quasi totalità dei segni accentati che si usano nelle lingue europee, quindi l'uscita è prodotta direttamente con i segni accentati e senza acrobazie per sovrapporre l'accento sulla lettera da accentare.

LaTeX fornirebbe sullo schermo e nel file `.log` le informazioni seguenti

```
Underfull \hbox (badness 10000)
      in paragraph at lines 146--146
[] \T1/cmr/m/n/10.95 elet-tri-ci-tà
```

```
\hbox(7.54149+0.0)x16383.99998,
      glue set 8998.3789 []
```

```
Underfull \hbox (badness 10000)
      in paragraph at lines 148--148
[] \T1/cmr/m/n/10.95 élec-tri-cité
```

```
\hbox(7.54149+0.0)x16383.99998,
      glue set 8998.3789 []
```

Il comando `\showhyphens` compone il suo argomento dentro una scatola orizzontale di lunghezza pari a 16384 punti (circa 5m) e ovviamente la scatola risulta quasi vuota e infinitamente mal composta (`badness 10000`) per cui viene emesso il messaggio di avvertimento e le informazioni relative alla scatola mal composta con il suo contenuto diviso in sillabe come TeX lo farebbe. Si vede chiaramente che in italiano è persa la prima possibile cesura grammaticale, perché lascerebbe a sinistra un sillaba di una sola lettera. Analogamente per il francese vengono perse sia la prima sia l'ultima cesura grammaticale, perché le sillabe a sinistra e a destra conterebbero meno caratteri di quanto specificato dai parametri `\dots hyphenmin` validi per il francese.

Grazie al fatto che in italiano gli accenti obbligatori cadono solamente sulla vocale terminale delle parole tronche, la convenzione tipografica non produce effetti molto diversi con le codifiche OT1 o T1; in francese, al contrario, l'effetto con la codifica OT1 sarebbe devastante.

Va notato che per l'italiano le regole di sillabazione sono state create assumendo i valori di `\dots hyphenmin` entrambi pari a 1, ma le impostazioni nel file di descrizione della lingua italiana li mettono entrambi a 2. Nessuno vieterebbe al compositore di assegnare i valori unitari ai due parametri suddetti, ma certamente la sua composizione sarebbe soggetta a critiche estetiche non indifferenti.

3 I codici di categoria

I codici di categoria, in particolare i codici 11 e 12 si riferiscono rispettivamente alle lettere dell'alfabeto e ai segni di interpunzione. Le impostazioni di LaTeX sono tali che se viene invocato il pacchetto `fontenc` con l'opzione T1, sono dichiarate lettere non solo le 26 lettere minuscole e le 26 lettere maiuscole dell'alfabeto latino, ma anche tutte le loro varianti con tutti i possibili segni diacritici oltre ai segni speciali che compaiono in varie lingue, come

per esempio ß, ŋ, ð, đ, þ. A ogni lettera minuscola è associato il codice della corrispondente lettera maiuscola e viceversa, così che possano essere eseguite le trasformazioni da maiuscolo a minuscolo e viceversa mediante i comandi `\MakeUppercase` e `\MakeLowercase`.

Tuttavia il codice di lettera minuscola è un requisito essenziale per considerare un carattere come facente parte di una parola. Anche i segni di interpunzione possono avere associato il codice di lettera minuscola, sebbene a prima vista non si capisca bene a che cosa possa servire. Serve semplicemente a TeX per stabilire se quel segno può alterare la divisione in sillabe oppure no. Le regole devono quindi tenere conto della presenza di questi segni considerandoli come parti integranti della parola da dividere in sillabe.

Non è che ci siano tanti segni di questo genere. Lo scrivente è al corrente che solo l'italiano, il francese e il catalano usano l'apostrofo con il significato di elisione di una vocale. In questo caso una parola in senso lato come "bell'accoglienza" è divisibile in sillabe dopo l'apostrofo se e solo se l'apostrofo ha associato il codice di lettera minuscola (un valore qualunque non negativo, e tanto vale allora che questo valore coincida con l'indirizzo dell'apostrofo nella polizza dei caratteri in uso). Si noti che se l'apostrofo non fosse associato al suo valore di lettera minuscola, l'esempio proposto sopra troverebbe TeX incapace di dividere in sillabe "bell'accoglienza", perché l'unica parola che riconoscerebbe sarebbe "bell", divisibile grammaticalmente in "bel-l", ma l'ultima sillaba a destra sarebbe troppo corta. Dichiarando che l'apostrofo, pur mantenendo la categoria 12 di segno non letterale, ha associato un codice di lettera minuscola, TeX riconosce "bell'accoglienza" come un'unica parola e deve conoscere le regole di divisione per parole composte anche di apostrofi. La sua divisione, con queste regole, è pertanto "bel-l'ac-co-glien-za"; come si vede, queste regole impediscono di andare a capo dopo l'apostrofo: questa pratica è comunemente accettata in tipografia, anche se si chiude un occhio quando si compone in colonne molto strette; tuttavia TeX così come è configurato si rifiuta categoricamente di eseguire una operazione del genere.

4 I pattern di sillabazione

Le regole grammaticali di divisione in sillabe variano ovviamente da lingua a lingua: per l'italiano sono molto semplici; in francese esse sono un po' più complesse. Per l'inglese sono estremamente complesse da eseguire a macchina perché dipendono dalla accentazione e parole omografe risultano divise in sillabe differentemente a seconda di come vengono pronunciate: si pensi che *record* (verbo) e *record* (sostantivo) vengono pronunciate rispettivamente *recòrd* e *rècord*; le divisioni in sillabe

diventano pertanto rispettivamente *re-cord* e *rec-ord*; lo stesso succede per parole come *analyses* pronunciato *anàlyses* (plurale del sostantivo *analysis*) o *analýses* (terza persona singolare del verbo *to analyse*); inoltre le regole di sillabazione per la varietà americana sono diverse da quelle della varietà britannica.

In tedesco sono complessissime a causa delle parole composte, dove la cesura non può non cadere nei punti di giunzione delle parole componenti e può provocare anche un cambio di ortografia.

Tenuto conto del fatto che l'algoritmo di divisione in sillabe viene invocato da T_EX molto frequentemente, è necessario che queste regole siano formulate in modo che T_EX le possa consultare con la massima rapidità.

Frank Liang era studente a Stanford quando Knuth stava producendo la prima versione del programma T_EX; egli svolse la sua tesi di master sulla sillabazione mediante calcolatore, LIANG (1983); il suo problema era quello di trovare una struttura dati molto rapida da esplorare e sviluppò una struttura nuova che risultò particolarmente efficace. La sua tesi è accessibile alla lettura di molte persone che abbiano una minima conoscenza dell'informatica, o meglio della "computer science": anche se perdono qualche dettaglio teorico, la lettura è affascinante.

Il metodo di scrittura delle regole di Liang si basa sui *pattern*; questi sono frammenti di parola che contengono le informazioni per inserire o vietare una cesura fra le lettere che compongono il frammento. Usare i frammenti di parola rende l'elenco delle cesure permesse o vietate molto più compatto che non un intero dizionario di parole divise in sillabe; inoltre questo insieme di frammenti può essere messo dentro una struttura dati adeguata per la ricerca rapida dei frammenti.

Ciò richiede che sia creata la lista dei pattern e che questa sia predigerita da T_EX in modo che la struttura dati sia già pronta quando T_EX lavora per la composizione.

Il primo punto può essere eseguito con il programma **patgen**, mentre il secondo è quell'aspetto delicato che sfugge alla maggior parte degli utenti del sistema T_EX, cioè che questa predigestione è svolta nel momento in cui si crea il file di formato con estensione *.fmt* o *.efmt*. Spesso gli utenti si lamentano dicendo: "Ho caricato il pacchetto **babel** con l'opzione **italian**, ma poi L^AT_EX divide le sillabe in modo sbagliato". Certo, ciò succede perché T_EX non trova nel file di formato le regole di sillabazione nella struttura dati con i pattern per l'italiano, perché il responsabile del sistema, di solito il proprietario o l'amministratore del computer, si è dimenticato di creare o ricreare il formato.

4.1 Il file di formato contiene i pattern di sillabazione

Le varie distribuzioni del sistema T_EX contengono meccanismi vari per creare il file di formato; alla fin fine il tutto si riduce a eseguire il programma **initex** specificando come input il nome del file di cui si vuole predigerire il contenuto e creare il file di formato.

Lanciando **initex** sul file **latex.ltx** viene creato il formato **latex.fmt**; un file di configurazione specifica per quali lingue siano da incorporare i pattern; molte delle ultime distribuzioni del sistema T_EX predigeriscono in fase di installazione tutti i pattern di tutte le lingue di cui esistono i corrispondenti file di pattern. MiK_T_EX ha un comodo programma di configurazione con interfaccia grafica che consente di marcare le lingue desiderate e di eseguire la creazione dei file di formato con pochi click su appositi bottoni.

Ogni volta che si cambia l'insieme delle lingue bisogna ricreare i file di formato; la prima volta che si installa la distribuzione del sistema T_EX bisogna creare il file di formato o almeno bisogna verificare che il file di default creato durante l'installazione contenga le regole di sillabazione per tutte le lingue che interessano, altrimenti bisogna ritornare sulla configurazione, eseguire le modifiche necessarie e, infine, ricreare i file di formato.

Per essere sicuri che il proprio sistema T_EX sia correttamente configurato anche per l'italiano basta creare il semplice file sorgente seguente

```
% file sillabazione.tex
\documentclass{minimal}
\usepackage[italian]{babel}
\begin{document}
\showhyphens{equazione}
\end{document}
```

Lanciando la compilazione di questo brevissimo file non si produce nessuna pagina in uscita ma si ottiene sullo schermo (e nel file *.log*) quello che L^AT_EX pensa sia la divisione in sillabe corretta; se sullo schermo appare **equa-zio-ne**, l'italiano è ben configurato anche come sillabazione; se sullo schermo appare **equ-a-zione** la divisione in sillabe corrisponde alla lingua di default, cioè all'inglese. Bisogna quindi rivedere la questione della configurazione e procedere alla creazione del nuovo formato.

4.2 Come funzionano i pattern

I pattern, dunque, sono dei frammenti di parola (scritti in lettere minuscole) dove sono indicati i possibili punti dove la divisione in sillabe è consentita oppure è vietata. Questi punti sono indicati mediante cifre da 0 a 9 intercalate alle lettere: le cifre dispari consentono la cesura in quel punto, mentre le cifre pari la vietano; la cifra zero è sottintesa e quindi non viene quasi mai indicata

b	e	l	l	'	a	c	c	o	g	l	i	e	n	z	a																			
b0e01010	'0a0c0c0o0g010i0e0n0z0a																																	
110	0'2	1c0				1g0				1n0																								
110		1c0				110				1z0																								
21010		2c0c0				0g210				2n0z0																								
b0e21110	'2a2c1c0o1g210i0e2n1z0a																																	
bel-l'ac-co-glien-za																																		

TABELLA 1: Processo di divisione in sillabe mediante i pattern della lingua italiana; la seconda riga della tabella contiene i codici di default, tutti valori nulli a sinistra e a destra di ogni lettera.

esplicitamente. Se ad una parola si applicano diversi pattern e fra due lettere un pattern indica un valore mediante una cifra e fra le stesse due lettere un altro pattern indica un diverso valore, prevale il valore più grande.

Nella tabella 1 è indicata la serie di pattern che si applica alla parola “bell'accoglienza”, intesa da TeX come un'unica parola, mentre noi umani sappiamo che si tratta di due parole fra le quali è intervenuta l'elisione.

Dalla tabella si può osservare, esaminando ogni riga dall'alto al basso, la parola iniziale seguita dai vari pattern che si trovano nel file `ithyph.tex` e che contengono frammenti della parola da dividere in sillabe: i pattern sono incolonnati insieme ai valori numerici che informano sulla possibilità di inserire o l'obbligo di non inserire una cesura; in questo modo è agevole controllare colonna per colonna qual è il valore numerico più alto e definire quindi l'intera parola con tutti i valori numerici intercalati. In corrispondenza dei valori finali dispari è possibile inserire la cesura, per cui il risultato finale riportato nell'ultima riga è perfettamente coincidente con quanto insegna la grammatica.

4.3 La creazione dei pattern

Per usare il programma `patgen`, creato da Liang a questo scopo, sarebbe necessario disporre di un elenco di parole con le divisioni in sillabe già segnate; Liang ha fatto questo lavoro per l'inglese usando circa 25 000 parole e ha ottenuto per la lingua inglese della varietà americana poco meno di 4800 pattern. Riapplicando questi pattern all'elenco delle 25 000 parole ha constatato che il 90% di queste risultavano divise in modo corretto. Il che la dice lunga sulla difficoltà di dividere in sillabe le parole inglesi.

Il programma TeX accetta anche un elenco di eccezioni dato come argomento al comando `\hyphenation`:

```
\hyphenation{<lista delle eccezioni>}
```

dove la *lista delle eccezioni* è costituita dall'elenco delle parole scritte in lettere minuscole e con i trattini indicanti le cesure lecite. In italiano si potrebbero per esempio inserire nella lista le parole

composte che si desidera dividere etimologicamente invece che foneticamente come la grammatica consente e suggerisce:

```
\hyphenation{su-per-in-dut-to-re
su-per-in-dut-to-ri
ma-cro-istru-zio-ne
ma-cro-istru-zio-ni}
```

Per l'americano la lista delle eccezioni contiene ora molte centinaia di parole e la rivista TUGboat pubblica gli aggiornamenti regolarmente; l'elenco aggiornato si trova sempre negli archivi CTAN.

Per l'italiano lo scrivente ha inizialmente creato una lista di pattern usando solo la grammatica e ha controllato la correttezza delle cesure su un elenco di parole appositamente creato per eseguire questo controllo; dopo pochi aggiornamenti il file `ithyph.tex`, che accompagna ogni distribuzione del sistema TeX, è rimasto praticamente inalterato e contiene circa 320 pattern con il livello più alto di consenso/proibizione di eseguire la cesura che arriva alla cifra 4.

Successivamente ha avuto a disposizione una versione del programma `patgen` accompagnato da istruzioni per l'uso abbastanza decifrabili ed ha provato a verificare i suoi pattern alla luce dello stesso elenco di parole appositamente costruito. I pattern ottenuti sono stati marginalmente in numero maggiore e il massimo livello di consenso/proibizione arriva a 5. Tutto questo conferma la bontà del programma `patgen` che arriva quasi a fare altrettanto bene di quello che può fare una persona istruita.

Nella tabella 2 si riporta solo l'elenco dei pattern tratto dal file `ithyph.tex` per eseguire qualche commento; il lettore interessato può avere a disposizione per la lettura l'intero file che è contenuto nella sua distribuzione del sistema TeX. Ci si ricordi solamente che nei pattern il punto segna l'inizio o la fine di una parola.

Si nota che l'intero contenuto della tabella di pattern è racchiuso fra le parentesi graffe; in questo modo eventuali definizioni o assegnazioni di valori restano confinate al gruppo e non si propagano oltre.

Infatti si assegna subito all'apostrofo il valore di lettera minuscola:

```
\lccode{'\}'='\'}
```

Le lettere `lc` stanno per le iniziali di “lower case”; la scrittura `'\'` si riferisce al codice numerico del segno `'`; con l'assegnazione indicata nel file si attribuisce come codice di lettera minuscola all'apostrofo il suo stesso codice. L'apostrofo continua a essere una “non lettera” con codice di categoria pari a 12, ma con il codice di minuscola non negativo esso è considerato da TeX parte di quelle stringhe che lui considera parole.

Seguono poi come argomento di `\pattern` tutti i pattern per l'italiano: si comincia dall'apostrofo

TABELLA 2: I pattern per la lingua italiana

```

%% file ithyph.tex
%
{\lccode'\='\' % Apostrophe has its own lccode
% so that it is treated as a letter
%>> 1998/04/14 inserted grouping
%
\patterns{
% After the Garzanti dictionary:
% a-pnea, a-pnoi-co,...
.a3p2n
.anti1 .anti3m2n
.biol
.ca4p3s
.circu2m1
.di2s3cine
.fran2k3
.free3
.narco1
.opto1
.orto3p2
.para1
.poli3p2
.pre1
.p2s
.sha2re3
.tran2s3c .tran2s3d .tran2s3f .tran2s3l .tran2s3n
.tran2s3p .tran2s3r .tran2s3t
.su2b3lu .su2b3r
.wa2g3n
.wel2t1
alia alie alio aliu aluo alya 2at.
eliu e2w
olia o1ie o1io o1iu
'2
1b 2bb 2bc 2bd 2bf 2bm 2bn 2bp 2bs 2bt 2bv b2l b2r 2b. 2b'. 2b''
1c 2cb 2cc 2cd 2cf 2ck 2cm 2cn 2cq 2cs 2ct 2cz 2chh c2h 2chb ch2r 2chn
c2l c2r 2c. 2c'. 2c'' .c2
1d 2db 2dd 2dg 2dl 2dm 2dn 2dp d2r 2ds 2dt 2dv 2dw 2d. 2d'. 2d'' .d2
1f 2fb 2fg 2ff 2fn f2l f2r 2fs 2ft 2f. 2f'. 2f''
1g 2gb 2gd 2gf 2gg g2h g2l 2gm g2n 2gp g2r 2gs 2gt 2gv 2gw 2gz 2gh2t
2g. 2g'. 2g''
1h 2hb 2hd 2hh hi3p2n h2l 2hm 2hn 2hr 2hv 2h. 2h'. 2h''
1j 2j. 2j'. 2j''
1k 2kg 2kf k2h 2kk k2l 2km k2r 2ks 2kt 2k. 2k'. 2k''
1l 2lb 2lc 2ld 2l3f2 2lg 12h 2lk 2ll 2lm 2ln 2lp 2lq 2lr 2ls 2lt 2lv 2lw
2lz 2l. 2l'. 2l''
1m 2mb 2mc 2mf 2ml 2mm 2mn 2mp 2mq 2mr 2ms 2mt 2mv 2mw 2m. 2m'. 2m''
1n 2nb 2nc 2nd 2nf 2ng 2nk 2nl 2nm 2nn 2np 2nq 2nr 2ns 2nt 2nv 2nz n2g3n
2nheit. 2n. 2n' 2n''
1p 2pd p2h p2l 2pn 3p2ne 2pp p2r 2ps 3p2sic 2pt 2pz 2p. 2p'. 2p''
1q 2qq 2q. 2q'. 2q''
1r 2rb 2rc 2rd 2rf r2h 2rg 2rk 2rl 2rm 2rn 2rp 2rq 2rr 2rs 2rt rt2s3 2rv 2rx
2rw 2rz 2r. 2r'. 2r''
1s2 2shm 2s3s s4s3m 2s3p2n 2stb 2stc 2std 2stf 2stg 2stm 2stn 2stp 2sts 2stt 2stv 2sz 4s.
4s'. 4s''
1t 2tb 2tc 2td 2tf 2tg t2h t2l 2tm 2tn 2tp t2r 2ts 3t2sch 2tt 2tv 2tw t2z
2tzk 2tzs 2t. 2t'. 2t''
1v 2vc v2l v2r 2vv 2v. 2v'. 2v''
1w w2h wa2r 2w1y 2w. 2w'. 2w''
1x 2xt 2xw 2x. 2x'. 2x''
y1ou y1i
1z 2zb 2zd 2zl 2zn 2zp 2zt 2zs 2zv 2zz 2z. 2z'. 2z'' .z2
}} % Pattern end
\endinput

```

isolato, ma guardando più avanti si ritrova l'apostrofo alla fine delle parole, dove svolgerebbe la funzione di virgolette; i pattern con gli apostrofi in funzione di virgolette sono necessari per evitare che le virgolette vadano a capo da sole, lasciando la parola italiana alla fine della riga precedente immediatamente seguita dalla lineetta di cesura. È chiaro che questo sarebbe un modo ridicolo di gestire la cosa, mentre trattando l'apostrofo come una lettera comune e inserendo i codici pari necessari per interdire le cesure si risolve il problema in modo semplicissimo.

La lista contiene poi le divisioni di prefissi e prefissoidi; la grammatica consente di eseguire la cesura in modo fonetico per qualunque parola, tuttavia divisioni come “su-blu-na-re”, “tran-sna-zio-na-le”, che la grammatica consente, sarebbero difficili da leggere. Per altro la grammatica consente anche la divisione etimologica, quindi le divisioni indotte dai pattern specificati saranno “sub-lu-na-re” e “trans-na-zio-na-le”.

Si nota l'assenza di qualunque pattern che consideri vocali isolate o considerate a coppie; per le vocali isolate bastano i valori nulli di default, mentre per i dittonghi veri o per le coppie di vocali si sono lasciati i codici nulli (e quindi non sono stati indicati) per evitare di spezzare fra vocali. È vero che “au” può formare un dittongo e quindi sarebbe inseparabile, tuttavia esso è un dittongo solo se la vocale ‘u’ non è tonica; ma T_EX non conosce la pronuncia dell'italiano e quindi è meglio lasciare sempre la coppia “au” indivisa. Si perderà qualche possibile cesura, ma per il lettore la lettura sarà certamente agevolata: in questo modo egli troverà la divisione “bau-le” invece di “ba-u-le” e non si troverà mai la parola divisa dopo la ‘a’.

Compaiono invece i trittonghi, i gruppi di tre vocali che si pronunciano con un'unica emissione di voce: avremo così, per esempio, le divisioni “aiuo-la”⁴ e “ma-ieu-ti-ca”.

Con questa decisione non ci sono problemi a che i pattern trattino correttamente anche le vocali accentate sia in posizione terminale, come nelle parole tronche, sia quando ricevono una accentazione facoltativa, come quando si vogliono distinguere parole altrimenti omografe quali “séguito” e “seguito”⁵.

Seguono poi i pattern che riguardano le consonanti, per le quali bisogna distinguere le liquide e le nasali dalle altre; bisogna gestire correttamente la ‘s’ impura, caratteristica della sola lingua

4. Di fatto solamente “aiuo-la” perché la prima sillaba grammaticale è troppo corta; tuttavia se la parola è preceduta dall'articolo o da una preposizione articolata, il trittongo riappare: la sillabazione di “l'aiuola” diventa “l'a-aiuo-la”.

5. La grammatica consente l'accento facoltativo solo sulle parole sdrucciole o con l'accento ancora più arretrato. Non consente l'accento sulle parole piane, perché quella è la posizione di default, tuttavia consente l'accentazione delle parole piane per distinguerne il tono come in “cólto” e “còlto”.

italiana fra le varie lingue romanze. Bisogna trattare la ‘w’ e la ‘y’ in modo ragionevole; bisogna prevedere gruppi di consonanti non presenti nelle parole italiane in senso stretto ma derivate da radici straniere. In questo modo, sebbene la grammatica non dica niente al riguardo, vengono divise correttamente parole come maxwelliano, newyorke-se, leishmaniosi, lewisite, wahhabita... ma anche parole italiane dalla grafia insolita, come Santhià o Thiene.

5 Perché T_EX talvolta non divide in sillabe?

Talvolta T_EX si “rifiuta” di eseguire le necessarie cesure pur non essendone impedito dal tipo di font usato; si ricorda, infatti, che il font a spaziatura fissa non consente la divisione in sillabe.

I motivi possono essere diversi, ma spesso si tratta di errori di composizione; per esempio ci si scorda di lasciare lo spazio fra una parola e la susseguente o la precedente espressione matematica in linea: se il segno di dollaro è adiacente alla parola, T_EX non divide in sillabe, ma quasi sempre la mancanza dello spazio è un errore di battitura, e quindi la “colpa” non è di T_EX.

Ma ci sono tre situazioni subdole che il compositore stenta a riconoscere. Una situazione si ha quando la parola segue un cambiamento di font, specialmente se eseguito mediante una dichiarazione. Un'altra situazione si ha quando alla parola è “appesa” una nota, o meglio, il richiamo di nota; quest'ultimo infatti è composto dal comando \footnote o da \footnotemark alzando il contrassegno mediante il meccanismo degli esponenti matematici oppure mediante la manipolazione di “scatole”. La terza situazione si ha quando la parola è composta mediante elementi separati dal trattino.

In quest'ultimo caso T_EX non divide in sillabe né la prima né la seconda parola legate dal trattino, ma divide solo in corrispondenza del trattino. Questo comportamento è ragionevole, ma talvolta, con parole lunghe, sarebbe desiderabile consentire la cesura anche nel primo o nel secondo moncone. Siccome il problema si incontra in quasi tutte le lingue, la prima cosa da fare è leggere il file di testo che descrive quanto si è realizzato per quella particolare lingua; il pacchetto babel è corredato da un file per ogni lingua che riesce a gestire.

Per l'italiano basta fare precedere il doppio apice al trattino, scrivendo, quindi, “- al posto del semplice trattino, e la cesura viene abilitata in qualunque punto della parola composta.

Negli altri casi la soluzione è semplice ma appare come un “dirty trick”, per chiamarlo col nome che Knuth stesso ha attribuito a questi “sporchi trucchi” dedicando loro addirittura l'appendice D del suo T_EXbook, KNUTH (1984).

Basta inserire un grumetto di colla di larghezza

nulla prima o, rispettivamente, dopo l'oggetto che impedisce la sillabazione. Nello stesso tempo si desidera che T_EX non vada a capo in corrispondenza di questo grumetto di colla, specialmente nel caso che esso sia collocato fra la parola e il richiamo di nota.

La soluzione migliore, secondo lo scrivente, è quello di usare una breve macro come la seguente

```
\newcommand*\hz{%
  \nobreak\hskip 0pt\relax}
```

così da usarla in situazioni del tipo

```
... capoverso\hz\footnote{Si ricorda ...}
```

Non è il caso di inserire \hz prima di ogni nota, ma se si dovesse trovare non giustificata qualche parola per la presenza del richiamo di nota, allora in seconda battuta si può inserire \hz solo prima di quella nota.

Analogamente se una parola non risultasse giustificata dopo un cambiamento di font, basta premettere \hz alla parola (lasciando uno spazio subito dopo, spazio che T_EX usa solo per capire dove finisce la macro) e il problema è risolto. Si noti che si ha un cambiamento di font implicito all'inizio di ogni capoverso e all'inizio del testo introdotto da \item in una lista. Normalmente la prima parola di un capoverso o di un \item non richiede la sillabazione, ma se si sta componendo in colonne strette e/o la parola è molto lunga, allora anche in quel caso conviene cominciare il capoverso o l'\item con \hz.

Non è opportuno usare il comando \allowhyphens del pacchetto babel, anche se scrivendo in italiano quel pacchetto viene

sempre richiamato, perché esso *non* inserisce il grumetto di colla quando è in vigore la codifica T1, quella che si dovrebbe sempre usare quando si compone in italiano.

6 Conclusione

Questo breve tutorial in merito alla sillabazione dovrebbe essere riuscito a spiegare sia come fa T_EX ad eseguire la cesura delle parole in fin di riga, sia l'importanza di includere i pattern di sillabazione di una data lingua nel file di formato del programma che si intende usare, sia, infine, che cosa fare per consentire l'esecuzione della cesura anche quando T_EX ne è impedito dalle sue regole stringenti per isolare la sequenza di lettere che costituisce una parola.

Riferimenti bibliografici

- GREGORIO, E. (2006). «Codici di categoria». *ArsT_EXnica – Rivista italiana di T_EX e L^AT_EX*, (2).
- KNUTH, D. E. (1984). *The T_EXbook*. Addison Wesley Publ. Co., Reading, Mass.
- LIANG, F. (1983). *Word Hy-phen-a-tion by Computer*. Tesi di Laurea, Stanford University. La tesi originale è stata scandita e resa disponibile nel sito del T_EX Users Group; <http://www.tug.org/docs/liang/liang-thesis-hires.pdf>.

▷ Claudio Beccari

L^AT_EXpedia: il futuro della documentazione su L^AT_EX

Lapo Filippo Mori

Sommario

La documentazione è un fattore molto importante per il successo di software *open source* perché ne influenza la diffusione. Chi inizia a studiare L^AT_EX, e anche l'utente intermedio, si lamenta spesso della documentazione: è difficile trovare una risorsa aggiornata, completa e ben strutturata. Questo articolo analizza tutti i tipi di risorsa su L^AT_EX, evidenziando vantaggi e svantaggi, per poi proporre un nuovo tipo: un'enciclopedia *online* e *open content*, L^AT_EXpedia.

Abstract

Software documentation is a very important success factor for open source software because it bolsters its diffusion. People who start learning L^AT_EX and even intermediate users often complain about L^AT_EX documentation: it is hard to find an updated, complete and well structured resource. This article evaluates advantages and disadvantages of the different sorts of resources for L^AT_EX documentation available and proposes a new kind of documentation source: a free-content, web-based encyclopedia, L^AT_EXpedia.

1 Introduzione

La documentazione di un programma è molto importante, soprattutto se *open source*, in quanto essa influenza la sua diffusione e il suo sviluppo, quindi il suo successo. Se il miglior programma ha una documentazione scarsa, non attrae nuovi utenti e il numero degli sviluppatori resta basso.

Spesso mi chiedono quale sia la migliore guida o libro per imparare L^AT_EX e io ho sempre difficoltà a rispondere; questo fa pensare che la documentazione su L^AT_EX abbia un problema e, se è così, è necessario trovare una soluzione. Essendo un amministratore del forum di G_UI_T (Gruppo Utilizzatori Italiani di T_EX) e l'autore di due guide MORI (2005, 2007), ho maturato nel tempo un'idea personale delle debolezze e dei punti di forza delle risorse di informazione disponibili su L^AT_EX ma fino ad adesso non sono mai riuscito a trovare una soluzione.

In questo articolo analizzo le risorse attualmente disponibili e i motivi del loro cattivo funzionamento, per poi proporre una nuova soluzione a questo importante problema: L^AT_EXpedia.

2 Il problema

2.1 Troppe risorse

Lavorando sulla bibliografia di una guida di livello intermedio sulla scrittura di tesi di laurea con L^AT_EX MORI (2005, 2007) mi sono reso conto di quante risorse siano disponibili.¹ Una delle bellezze di L^AT_EX, così come degli altri programmi *open source*, è che è stato oggetto di molte guide e libri. Questo significa che un utente avanzato, ogni volta che si presenta un nuovo problema, può confidare su una letteratura enorme e trovare la soluzione allo stesso problema o a uno simile in pochi minuti. Per un principiante significa che, alla difficoltà intrinseca allo studio di un nuovo linguaggio, si aggiunge quella di trovare buone fonti su cui studiare. Un occhio esperto riesce a capire velocemente se vale la pena leggere una guida o no, ma al principiante sembrano tutte uguali.

2.2 Obsolescenza

Non solo le risorse sono troppe, ma spesso sono anche in contraddizione. Alcune sono state scritti diversi anni fa e le informazioni che riportano sono obsolete e in contrasto con guide più moderne. L^AT_EX, essendo un programma *open source*, è costantemente aggiornato e decine di nuovi pacchetti vengono scritti ogni anno. A causa della poca coordinazione tra gli autori dei pacchetti, talvolta essi sono incompatibili. In questo scenario è molto importante che le guide siano aggiornate regolarmente.

Come può un principiante sapere qual è la soluzione giusta di un problema se non sa nemmeno quali sono le risorse da scartare?

2.3 Serve un nuovo tipo di risorsa

Il vero problema è che nessuna delle risorse disponibili è perfetta. Analizziamo dunque ogni tipo di risorsa, identificandone i lati positivi e negativi, con lo scopo di proporre un nuovo tipo.

2.3.1 Libri

I libri, ad esempio BECCARI (1991); DILLER (1999); GOOSSENS *et al.* (1995, 1997); GRÄTZER (1999, 2000); HAHN (1993); HIGHAM e GRIFFITHS (1997); KNUTH (1992); KOPKA e DALY (1995); LAMPORT (1994a), sono un eccellente punto di partenza per imparare L^AT_EX perché sono, per loro natura, completi. Sfortunatamente, i principianti di solito si avvicinano al nuovo linguaggio leggendo le guide

1. Si vedano per esempio i riferimenti bibliografici riportati nei par. 2.3.1 e 2.3.2.

che sono disponibili gratuitamente su internet. I libri sono spesso acquistati da utenti intermedi che, dopo aver imparato le basi su queste guide e trovato L^AT_EX adeguato alle loro esigenze, decidono di studiarlo più sistematicamente e comprano un libro.

Di solito un libro viene scritto da pochi autori e, essendo solitamente lungo, richiede un grande impegno specifico (impegno per autore). Per questo il *time-to-market* è di almeno un anno e il numero di nuove edizioni è generalmente molto basso; i libri sono dunque destinati a divenire obsoleti.

In teoria un libro potrebbe divenire l'unica risorsa per L^AT_EX con una soluzione a tutti i problemi rilevanti; tuttavia il numero degli autori deve rimanere basso per ridurre i problemi di coordinazione e quindi un progetto del genere richiederebbe anni e, nel frattempo, la maggior parte del materiale diventerebbe obsoleta.

Suggerimenti per L^AT_EXpedia.

La nuova risorsa deve:

1. **essere gratuita** in modo da essere invitante anche per i principianti;
2. coinvolgere **molte persone** ed **essere rilasciata in parti prima della sua conclusione** (altrimenti il *time-to-market* diventa troppo lungo e gran parte del lavoro viene rovinato dall'obsolescenza).

2.3.2 Guide

Tra le varie risorse, le guide sono senza dubbio le più numerose. Su internet sono disponibili sia guide generiche per principianti BAUDOUIN (1998); FLYNN (2005); GREGORIO (2006); OETIKER *et al.* (2006); THE TUTORIAL TEAM (2000); TRETTIN e FENN (2006) che guide su argomenti specifici quali tabelle CARUSILLO (2005); CAUCCI e SPADACCINI (2005); FEAR (2005); MORI (2006); VOSS (2003b), figure CAUCCI e SPADACCINI (2005); DALY (1998); HUTCHINSON (2005); RECKDAHL (1997); VOSS (2003b), font CARUSILLO (2006); LEHMAN (2004), matematica AMERICAN MATHEMATICAL SOCIETY (1995); VOSS (2003a), creazione di file PDF PRATESI *et al.* (2003); TALBOT (2004), tanto per citarne alcuni. Questi riferimenti sono quelli che di solito suggerisco, tuttavia una rapida ricerca su internet ne fornirebbe molti altri CAON (2003); CARUSILLO (2004); CECCHINI (1996); DOWTY (2002); ELECTRONIC PUBLISHING UNIT (2001); GONZATO (2003); GREENBERG (2006); LAMPORT (1994b); LOVE (1997); MURRI (2000); TONTI (1998); WARBRICK (2002); WILKINS (1995).

Caratteristiche comuni.

Come si può facilmente immaginare le guide, dato il loro numero, si differenziano molto per contenuto, stile, lunghezza e livello di dettaglio; è tuttavia

possibile identificare alcune caratteristiche comuni. Essendo più corta di un libro, una guida o copre molti argomenti poco dettagliatamente oppure pochi argomenti con maggiore dettaglio. Questo significa che, a differenza di un libro, è improbabile che una sola guida sia sufficiente per imparare a usare L^AT_EX. Nonostante che le guide richiedano un impegno specifico inferiore rispetto a un libro (questo spiega la loro numerosità), anch'esse hanno una tendenza all'obsolescenza.

Ognuno usa L^AT_EX per fare cose diverse e quindi è improbabile che la guida scritta da poche persone possa affrontare i problemi di tutti. Scrivendo una guida MORI (2005, 2006, 2007), ho sempre posto l'accento sui problemi che avevo trovato fino quel momento da utente L^AT_EX; alcuni utenti incontrano gli stessi problemi, ma molti altri hanno esigenze differenti quindi sarebbe meglio che il maggior numero di persone possibile contribuisca a una guida. Sfortunatamente è molto difficile coordinare la scrittura di un documento con molti autori: la coordinazione assorbirebbe più risorse della scrittura stessa.

Suggerimenti per L^AT_EXpedia.

La nuova risorsa deve:

1. **coprire con sufficiente dettaglio molti argomenti** in modo da poter essere usata per imparare L^AT_EX senza il bisogno di altre risorse.
2. **essere scritta da molte persone** in modo che gli argomenti siano molti e vari.

2.3.3 Forum

Con l'uso sempre più diffuso di internet, i forum sono diventati molto popolari. Per quanto riguarda L^AT_EX è necessario ricordare il gruppo di Usenet su L^AT_EX² e il forum di G_LT_EX.³

Cattive notizie.

Quando gli utenti esperti sono sufficientemente numerosi, la risposta ai topic è generalmente molto veloce e il forum funziona da *help center*. Il problema principale dei forum è che gli utenti L^AT_EX più esperti di solito spendono la maggior parte del loro tempo per rispondere a domande elementari che spesso sono già state affrontate in topic precedenti. Lo staff del forum di G_LT_EX ha definito alcune regole di condotta (*netiquette*) per intervenire su questo problema. Nonostante un buon motore di ricerca e un database di oltre duemila topic, i nuovi utenti continuano a fare sempre le stesse domande e gli amministratori continuano a indirizzarli verso topic precedenti o, peggio ancora, a fornire le stesse risposte.

2. <http://groups.google.com/group/comp.text.tex>
3. <http://guit.sssup.it/phpbb/index.php>

Quando un problema è già stato affrontato, gli utenti sono indirizzati verso questo. Sfortunatamente, spesso la risposta non è contenuta in un solo topic, ma in diversi.

Un altro problema dei forum è che le informazioni contenute, per quanto enormi, non sono strutturate. Secondo me questo è il principale svantaggio dei forum: non possono essere usati come strumento didattico perché le informazioni sono così sparse che la loro ricerca richiede più tempo del loro studio.

Buone notizie.

D'altro canto, un forum è probabilmente la risorsa di informazioni più aggiornata dato che gli utenti possono commentare i nuovi pacchetti e le nuove funzioni praticamente in tempo reale: ogni utente può creare un nuovo topic in qualche minuto e questo è immediatamente disponibile per l'intera comunità.

Un'altra caratteristica unica dei forum è la capacità di coinvolgere gli utenti. Quando un utente trova il forum utile, è probabile che in futuro aiuti altri utenti. In questo modo la conoscenza non è creata solo da pochi esperti ma da una base sempre più grande. Ovviamente, gli esperti, che sono spesso gli amministratori del forum, controllano che le informazioni fornite dagli altri siano corrette.

*Suggerimenti per L^AT_EX*pedia.

La nuova risorsa deve:

1. **permettere agli utenti intermedi di contribuire** allo sviluppo del progetto altrimenti la crescita è troppo lenta e gli utenti con più esperienza spendono tutto il loro tempo in attività di basso livello invece che in coordinazione.
2. **essere strutturata** in modo da poter essere usata come strumento didattico dagli utenti principianti e intermedi.

2.3.4 Liste di FAQ

Su internet sono disponibili numerose FAQ (Frequently Asked Questions). Queste contengono un certo numero di domande che possono sorgere usando L^AT_EX e le corrispondenti risposte. Di solito il materiale non è strutturato e quindi difficilmente può essere usato da un principiante che ha bisogno di essere guidato. Le FAQ sono spesso sviluppate in base al feedback degli utenti, tuttavia il numero di persone che mantiene queste liste è solitamente basso e quindi sono sviluppate e aggiornate lentamente. Il principale svantaggio di una FAQ rispetto a un forum è che gli utenti non possono fare domande direttamente e il processo di aggiornamento è molto più lento.

Tra le FAQ sul L^AT_EX merita ricordare la "UK List of T_EX Frequently Asked Questions". È mantenuta dal TUG (T_EX Users Group) inglese ed è

disponibile sia in formato HTML⁴ che in formato PDF FAIRBAIRNS (2005); contiene molte informazioni, è abbastanza strutturata e viene aggiornata regolarmente. Nonostante questi meriti, non può essere usata da principianti e gli aggiornamenti sono molto più lenti di un forum.

*Suggerimenti per L^AT_EX*pedia.

La nuova risorsa deve:

1. **essere aggiornata velocemente** in base al feedback degli utenti.
2. **essere ben strutturata** come un libro in modo da poter sia essere usata per cercare un argomento specifico (come una FAQ o un forum) sia essere letta dall'inizio alla fine (come un libro o una guida).
3. **essere disponibile sia in formato html che pdf.**

3 La soluzione

La soluzione è un'enciclopedia su L^AT_EX basata su un motore Wiki⁵ a cui tutti possano contribuire; il nome naturale per il progetto è dunque L^AT_EXpedia.⁶

L'idea di un'enciclopedia su L^AT_EX non è nuova. Denis Roegel e Jean-Michel Antoine nel 1995 hanno creato "L^AT_EX Navigator: A L^AT_EX Enciclopedia"⁷ che è ospitata da LORIA (Laboratoire Lorrain de Recherche en Informatique et ses Applications), un centro di ricerca francese in informatica. Tuttavia qui proponiamo di andare molto oltre: seguire l'esempio di Wikipedia.

Wikipedia⁸ è un'enciclopedia in rete, a contenuto libero e multilingue, lanciata nel gennaio 2001 da Larry Sanger e Jimmy Wales. Wikipedia fu inizialmente pensata come complemento a Nupedia,⁹ enciclopedia defunta nel 2003. Nupedia è scomparsa perché era scritta solo da esperti (diversi ordini di grandezza meno numerosi degli autori di Wikipedia) e perché il processo di revisione di ogni articolo era molto complicato: questi due aspetti fondamentali rendevano la crescita dell'enciclopedia troppo lenta. Wikipedia usa un motore per generare la grafica¹⁰ che permette di scrivere matematica con la sintassi di L^AT_EX: questo sicuramente aiuterebbe gli autori di L^AT_EXpedia a preparare gli esempi e potrebbe anche essere sfruttato per convertire automaticamente l'intera enciclopedia in

4. <http://www.tex.ac.uk/cgi-bin/texfaq2html>

5. http://en.wikipedia.org/wiki/Wiki_software

6. Wiki + encyclopedia = Wikipedia e analogamente L^AT_EX + encyclopedia = L^AT_EXpedia.

7. <http://tex.loria.fr/index.html>

8. <http://www.wikipedia.org/> e <http://en.wikipedia.org/wiki/Wikipedia:Introduction>

9. <http://en.wikipedia.org/wiki/Nupedia>

10. [http://en.wikipedia.org/wiki/Wikipedia:Manual_of_Style_\(mathematics\)](http://en.wikipedia.org/wiki/Wikipedia:Manual_of_Style_(mathematics)) e <http://meta.wikimedia.org/wiki/Help:Formula>

un file L^AT_EX per rilasciare periodicamente una versione PDF di L^AT_EXpedia.

Sicuramente scrivere un'enciclopedia richiede molto impegno, ma anche la manutenzione di un forum richiede molto lavoro da parte degli amministratori. Questo lavoro sarebbe certamente investito meglio in un progetto di lunga durata, con ottime prospettive e immediatamente disponibile come L^AT_EXpedia.

3.1 Vantaggi

L^AT_EXpedia raccoglie i vantaggi delle altre risorse e ne aggiunge di nuovi. Ricordiamo i principali dieci:

1. **Il numero di persone che contribuiscono al progetto è illimitato.** L'esperienza di Wikipedia ci insegna che spesso gli utenti che utilizzano l'enciclopedia contribuiscono al progetto e quindi il numero di persone che vi lavorano cresce continuamente.
2. **La varietà degli argomenti affrontati è enorme.** A differenza di ogni altra risorsa (a eccezione dei forum), il numero di utenti che contribuiscono a L^AT_EXpedia è enorme e ognuno di essi porta alla luce i problemi che ha trovato nella sua esperienza di utente L^AT_EX. Quando invece il numero degli autori è basso, il contenuto della risorsa non può coprire ogni possibile applicazione del L^AT_EX.
3. **Il lavoro è distribuito tra utenti con differenti capacità.** Tutte le altre risorse sono sviluppate solo da utenti esperti, mentre tutti gli altri (intermedi e principianti) beneficiano e basta del loro lavoro. Con L^AT_EXpedia ognuno può contribuire e quindi gli esperti possono concentrare le loro energie sulla revisione del lavoro svolto dagli altri e sulla strutturazione del progetto. Una migliore distribuzione del lavoro permette al progetto di crescere molto più velocemente.
4. **L'accesso al progetto è gratuito.** Nonostante che L^AT_EXpedia sia completa come un libro (o meglio), è gratuita e quindi allettante anche per chi si avvicina per la prima volta a L^AT_EX.
5. **Un potente motore di ricerca aiuta a trovare gli argomenti.** Come su un forum, gli utenti possono usare un motore di ricerca per trovare i singoli argomenti.
6. **Il progetto è ben strutturato.** Come un libro o una guida e a differenza di una lista di FAQ, L^AT_EXpedia può essere letta dall'inizio alla fine, il che è particolarmente importante per principianti. Questo è anche utile quando un utente intermedio sta cercando la soluzione di un problema particolare: se non sa di preciso l'argomento, per lui sarebbe difficile trovarlo

con un motore di ricerca ma la stessa struttura di L^AT_EXpedia può aiutarlo a capire cosa sta cercando.

7. **La documentazione diventa immediatamente disponibile.** Tutte le altre risorse (a eccezione dei forum) richiedono un certo lasso di tempo tra l'inizio della scrittura e la pubblicazione della prima edizione (dalla più lenta alla più veloce abbiamo i libri, le guide e le liste di FAQ). In questo progetto il materiale diventa immediatamente disponibile al pubblico, evitando il rischio dell'obsolescenza.
8. **Lo sviluppo del progetto è continuo.** A differenza dei libri e delle guide, il progetto L^AT_EXpedia continua fino a quando sono sviluppati L^AT_EX e i suoi pacchetti.
9. **Il processo di revisione è veloce.** Il processo di revisione, condotto sia da esperti che da utenti intermedi, è veloce perché ognuno può operare indipendentemente (in parallelo) quando ha tempo da dedicare al progetto.
10. **Il progetto è disponibile sia in versione pdf che html.** Mentre la versione HTML è aggiornata in tempo reale, la versione PDF può essere pubblicata periodicamente. Questo segue l'esempio di successo di "UK List of T_EX Frequently Asked Questions".

3.2 Dalla teoria alla pratica

3.2.1 Con un forum

Dato che molti TUG hanno un forum, quest'ultimi potrebbero essere usati per lo sviluppo di L^AT_EXpedia. Vediamo brevemente quale sia il funzionamento di un forum e come questo possa essere modificato per sviluppare un'enciclopedia.

Quando un utente apre un nuovo topic e propone una domanda a cui è già stato risposto sul forum, gli amministratori o un altro utente riportano il link del topic precedente. Spesso il topic precedente risponde solo in parte alla domanda e quindi nuovi pezzi di informazione vengono aggiunti al topic. Se invece il problema presentato è nuovo, si apre una nuova discussione.

Questa dinamica può essere leggermente modificata per sviluppare un'enciclopedia. Quando un utente ha un problema, lo espone sul forum. Se la soluzione è già disponibile su L^AT_EXpedia, gli amministratori riportano il link alla relativa pagina nell'enciclopedia e chiudono il topic. Se invece il problema è nuovo o non è completamente affrontato su L^AT_EXpedia, la discussione sul forum continua. Quando ottiene la soluzione, l'utente crea una pagina di L^AT_EXpedia con la descrizione del problema e la sua soluzione (oppure aggiunge le nuove informazioni a una pagina preesistente sull'argomento). Gli amministratori spostano quindi

il topic in una sezione del forum riservata ai problemi risolti e aggiungono un link a LaTeXpedia. Nel frattempo verificano anche che la pagina dell'enciclopedia contenga informazioni corrette e sia posizionata nella giusta sezione (ovvero verificano il rispetto della struttura dell'opera). In questo modo l'utente che riceve l'aiuto può ripagare la comunità collaborando alla scrittura dell'enciclopedia ed eventualmente riportando anche esempi di codice. LaTeXpedia è l'unico tipo di documentazione a cui possono contribuire anche i principianti.

3.2.2 Senza un forum

Nonostante che la presenza di un forum velocizzi il processo, LaTeXpedia può anche essere sviluppata senza: quando un utente non trova la soluzione a un problema sull'enciclopedia, crea una nuova pagina dove spiega il problema. Gli utenti che conoscono la soluzione modificano quella pagina fornendo un esempio. Gli esperti verificano la correttezza della soluzione e posizionano la pagina nella giusta sezione di LaTeXpedia.

4 Conclusioni

La natura multilingua di Wikipedia è uno dei suoi principali pregi. LaTeX è già un progetto multilingua con decine di TUG sparsi per il mondo. Lo sviluppo di LaTeXpedia gioverebbe sicuramente della collaborazione tra TUG e, al tempo stesso, lavorare su un progetto comune rafforzerebbe i legami tra i TUG. Quindi vorrei concludere incoraggiando gli altri TUG a unirsi a GJIT in questa nuova avventura.

Ringraziamenti

Vorrei ringraziare tutti i membri di GJIT che hanno contribuito allo sviluppo di questa idea e hanno offerto il loro aiuto per realizzarla, in particolare Sandro Allemanni, Daniele Avitabile, Gustavo Cevolani, Massimiliano Dominici, Maurizio Himmelmann ed Emiliano Vavassori.

Riferimenti bibliografici

- AMERICAN MATHEMATICAL SOCIETY (1995). *AMS-LaTeX User's Guide*. www.dd.chalmers.se/latex/Docs/PDF/amslatex.pdf.
- BAUDOUIN, M. (1998). *Impara LaTeX (...e mettilo da parte)*. http://www.mat.uniroma1.it/centro-calcolo/manuali/impara_latex.pdf.
- BECCARI, C. (1991). *LaTeX, Guida a un sistema di editoria elettronica*. Hoepli.
- CAON, D. (2003). *LaTeX: manuale di base*. www.guit.sssup.it/downloads/man_base.pdf.
- CARUSILLO, L. (2004). *LaTeX*.
- (2005). *Le tabelle in LaTeX*. <http://www.webalice.it/lgrs11/pgl/latex/Tabelle.pdf.tar.gz>.
- (2006). *I caratteri in LaTeX*.
- CAUCCI, L. e SPADACCINI, M. (2005). *Gestione di Figure e Tabelle con LaTeX*. http://www.guit.sssup.it/downloads/fig_tut.pdf.
- CECCHINI, R. (1996). *Guida locale all'uso del LaTeX*. www.fi.infn.it/pub/tex/doc/latexu.pdf.
- DALY, P. W. (1998). *Graphics and Colour with LaTeX*. <http://tex.loria.fr/graph-pack/grf/grf.pdf>.
- DILLER, A. (1999). *LaTeX Line by Line: Tips and Techniques for Document Processing*. John Wiley & Sons.
- DOWTY, D. (2002). *Introductory LaTeX Tutorial*. www.ling.ohio-state.edu/events/lcc/tutorials/intro-latex.pdf.
- ELECTRONIC PUBLISHING UNIT (2001). *Beginner's LaTeX*. UCC Computer Centre. 158.110.32.35/LATEX/beginlatex.pdf.
- FAIRBAIRNS, R. (2005). *The UK TeX FAQ*. <http://www.ctan.org/tex-archive/help/uk-tex-faq/letterfaq.tex>.
- FEAR, S. (2005). *Publication quality tables in LaTeX*. <http://www.ctan.org/tex-archive/macros/latex/contrib/booktabs/booktabs.pdf>.
- FLYNN, P. (2005). *A beginner's introduction to typesetting with LaTeX*. <http://www.tug.org/tex-archive/info/beginlatex/>.
- GONZATO, G. (2003). *LaTeX for Word Processor Users*. tug.ctan.org/tex-archive/info/latex4wp/latex4wp.pdf.
- GOOSSENS, M., MITTELBAACH, F. e SAMARIN, A. (1995). *The LaTeX Companion*. Addison-Wesley.
- GOOSSENS, M., RAHTZ, S. e MITTELBAACH, F. (1997). *The LaTeX Graphics Companion: Illustrating Documents with TeX and PostScript*. Addison-Wesley.
- GRÄTZER, G. (1999). *First Steps in LaTeX*. Springer Verlag.
- (2000). *Math into LaTeX*. Birkhauser.
- GREENBERG, H. J. (2006). *A Simplified Introduction to LaTeX*. www.cs.usask.ca/~wew036/latex/simplified-intro/latex.pdf.

- GREGORIO, E. (2006). *L^AT_EX: Breve guida ai pacchetti di uso più comune*. profs.sci.univr.it/~gregorio/breveguida.pdf.
- HAHN, J. (1993). *L^AT_EX for Everyone: A Reference Guide and Tutorial for Typesetting Documents Using a Computer*. Prentice Hall.
- HIGHAM, D. J. e GRIFFITHS, D. F. (1997). *Learning L^AT_EX*. Society for Industrial and Applied Mathematics.
- HUTCHINSON, I. (2005). *Graphics for Inclusion in Electronic Documents*. silas.psfc.mit.edu/elec_fig/elec_figures.pdf.
- KNUTH, D. (1992). *The T_EXbook*. Addison-Wesley.
- KOPKA, H. e DALY, P. (1995). *A Guide to L^AT_EX – Document Preparation for Beginners and Advanced Users*. Addison-Wesley.
- LAMPORT, L. (1994a). *L^AT_EX: A Document Preparation System, User's Guide and Reference Manual*. Addison-Wesley.
- (1994b). *L^AT_EX 2_ε: The macro package for T_EX*.
- LEHMAN, P. (2004). *The Font Installation Guide*. <http://www.ctan.org/tex-archive/info/Type1fonts/fontinstallationguide/fontinstallationguide.pdf>.
- LOVE, T. (1997). *Advanced L^AT_EX*. people.csail.mit.edu/custo/FTP/latex_tutorial.pdf.
- MORI, L. F. (2005). «Scrivere la tesi di laurea con L^AT_EX 2_ε». In *Proceedings del G_UITmeeting 2005*. G_UIT (Italian T_EX Users Group), Pisa, Italy, pp. 69–92. www.guit.sssup.it/guitmeeting/2005/articoli/mori.pdf.
- (2006). «Tabelle su L^AT_EX 2_ε: pacchetti e metodi da utilizzare». *Ars T_EXnica*, (2), pp. 31–47.
- (2007). «Scrivere la tesi di laurea con L^AT_EX 2_ε». *Ars T_EXnica*, (3).
- MURRI, R. (2000). *Introduzione a L^AT_EX: Scrivere in L^AT_EX con t_ET_EX ed Emacs su GNU/Linux*. <http://poisson.phc.unipi.it/~battiston/murripublichtml/lpratico/lpratico.pdf>.
- OETIKER, T., PARTL, H., HYNÄ, I. e SCHLEGL, E. (2006). *The Not So Short Introduction to L^AT_EX 2_ε*. www.ctan.org/tex-archive/info/lshort/english/lshort.pdf.
- PRATESI, M., LATINI, M. e ANTONECCHIA, M. (2003). *L^AT_EX-PDF-Howto*. <http://www.telug.it/marco/LaTeX-PDF-HOWTO/LaTeX-PDF-HOWTO-1.1.1.pdf>.
- RECKDAHL, K. (1997). *Using Imported Graphics in L^AT_EX 2_ε*. <http://www.ctan.org/tex-archive/info/epslatex.pdf>.
- TALBOT, N. (2004). *Creating a PDF document using PDF_LA_TE_X*. <http://theoval.cmp.uea.ac.uk/~nlct/latex/pdfdoc/pdfdoc-a4.pdf>.
- THE TUTORIAL TEAM (2000). *On-line Tutorial on L^AT_EX*. Indian T_EX Users Group. <http://www.tug.org.in/tutorials.html>.
- TONTI, E. (1998). *Il L^AT_EX mediante esempi*. www.dic.univ.trieste.it/perspage/tonti/MANUALI/latex.pdf.
- TRETTIN, M. e FENN, J. (2006). *An essential guide to L^AT_EX 2_ε usage: Obsolete commands and packages*. <http://www.ctan.org/tex-archive/info/l2tabu/english/l2tabuen.pdf>.
- VOSS, H. (2003a). *Mathmode*. www.ctan.org/tex-archive/info/math/voss/mathmode/Mathmode.pdf.
- (2003b). *Rotating Text, Tabulars and Images*. <http://perce.de/LaTeX/rotating.pdf>.
- WARBRICK, J. (2002). *Essential L^AT_EX*. www.sci.usq.edu.au/staff/robertsa/LaTeX/Others/essential.pdf.
- WILKINS, D. R. (1995). *Getting Started with L^AT_EX*. www.copernicus.org/guidelines/GSWLaTeX.pdf.

▷ Lapo Filippo Mori
Department of Mechanical Engineering,
Northwestern University, Evanston 60208 IL, USA
mori@northwestern.edu

Scrivere la tesi di laurea con L^AT_EX 2_ε

Lapo Filippo Mori*

Sommario

Lo scopo del presente articolo è fornire gli strumenti per scrivere una tesi di laurea utilizzando L^AT_EX 2_ε. Tale obiettivo è conseguito analizzando i problemi tipici incontrati durante la stesura della tesi e le possibili soluzioni; si pone particolare attenzione ai pacchetti da usare nelle varie circostanze. I singoli argomenti non vengono approfonditi nei dettagli ma si rimanda, ove necessario, alla letteratura specifica o ad i manuali dei pacchetti suggeriti.

Abstract

The goal of this article is to provide the tools to write a thesis with L^AT_EX 2_ε. The article analyzes the problems that are usually encountered while writing a thesis and their solution; a particular emphasis is on the packages to use in each case. The topics are not examined in deep and, when necessary, the reader is referred to specific literature or to the manual of the suggested packages.

Premessa

Il presente articolo non è una guida alla scrittura per tesi di laurea,¹ bensì spiega come utilizzare L^AT_EX 2_ε per scriverla. La filosofia seguita è quella di non approfondire i vari temi nei dettagli (per i quali si rimanda a manuali specifici), ma di trattare il più vasto numero di argomenti e di indicare le soluzioni che l'autore ritiene migliori. La scelta dei contenuti deriva in primo luogo dall'esperienza dell'autore ed in secondo dai numerosi interventi riguardanti tesi di laurea presenti sul forum del q_LT (Gruppo Utilizzatori Italiani di T_EX),² che resta sempre un valido riferimento per tutte le tematiche trattate nel presente documento.

*Ringrazio in primo luogo Fabiano Busdraghi che ha collaborato alla scrittura del par. 4, Massimiliano Dominici che ha curato gli aspetti relativi ai sistemi Linux e UNIX nel par. 4.1.2 e Riccardo Campana che ha curato gli aspetti relativi alla piattaforma Macintosh nel par. 6.1.4. Ringrazio inoltre tutti coloro che mi hanno consigliato durante la stesura e la revisione di questo documento, ed in particolare Claudio Beccari, Gustavo Cevolani, Massimo Guiggiani, Maurizio Himmelmann, Lorenzo Pantieri ed Emiliano Vavassori: senza il loro eccellente contributo questo articolo non avrebbe mai raggiunto la forma attuale.

1. Sono attualmente disponibili numerose guide alla scrittura sia per tesi di laurea di tipo generico ECO (1977); LESINA (1994) che specifiche per tesi di laurea scientifiche MATRICCIANI (2000, 2003).

2. Il forum del q_LT è raggiungibile al seguente indirizzo <http://www.guit.sssup.it/forum/>.

Il testo presume che il lettore conosca già i rudimenti di L^AT_EX 2_ε, ovvero che abbia letto una delle numerose guide di base disponibili gratuitamente in rete AMERICAN MATHEMATICAL SOCIETY (1995); BAUDOUIN (1998); FLYNN (2005); GREGORIO (2006); OETIKER *et al.* (2000); THE TUTORIAL TEAM (2000); VOSS (2003a) oppure un manuale cartaceo BECCARI (1991); DILLER (1999); GOOSSENS *et al.* (1995); GRÄTZER (1999, 2000); HAHN (1993); HIGHAM e GRIFFITHS (1997); KNUTH (1992); KOPKA e DALY (1995); LAMPORT (1994). Per alcuni argomenti (ad esempio figure e tabelle) si rimanda alla lettura di testi specifici ed in generale si invitano gli utenti a consultare le fonti Web presentate nel par. 7. Ogniquale volta si cita un pacchetto, non si fornisce una descrizione completa del suo funzionamento, per la quale si rimanda al relativo manuale,³ ma si analizzano le opzioni più importanti e se ne suggerisce l'utilizzo.

1 La documentclass

Per una tesi di laurea è consigliabile utilizzare la classe `book`. Nelle opzioni della classe, oltre alla dimensione del font di base (10pt, 11pt o 12pt)⁴ e a quella del foglio (tipicamente `a4paper`), è possibile scegliere:

- se avere un documento fronte-retro (`twoside`) o solo fronte (`oneside`),
- se collocare la prima pagina dei capitoli su facciate destre (`openright`) o indifferentemente (`openany`).

Si suggerisce di utilizzare la classe `book` invece di quella `report` in quanto la prima prevede tre comandi (`\frontmatter`, `\mainmatter` e `\backmatter`)⁵ che controllano il formato del numero di pagina e la numerazione dei capitoli. Nel `frontmatter` le pagine sono numerate con i numeri romani minuscoli (i, ii, iii, ecc.) ed i capitoli non sono numerati (come se si utilizzasse il comando asteriscato `\chapter*{}`). Nel `mainmatter` le pagine sono numerate con numeri arabi (la numerazione riparte da 1) e i capitoli sono anch'essi numerati con cifre arabe. Nel `backmatter` le pagine sono numerate

3. La maggioranza dei pacchetti per L^AT_EX è accompagnata da un manuale che ne descrive l'utilizzo e spesso presenta degli esempi. La posizione del manuale dipende dalla distribuzione T_EX che si usa ma in generale esso si trova in una sottocartella di `/texmf/doc`.

4. Per avere una buona leggibilità su fogli A4 è consigliabile usare un font di base di dimensione 11 pt.

5. Per l'uso di tali comandi si rimanda al par. 3.

come nel mainmatter (la numerazione prosegue da questa) ma i capitoli non sono numerati.

Si consiglia inoltre di utilizzare l'opzione fronte-retro (`twoside`) in quanto:

- si dimezza lo spreco di fogli di carta,⁶
- è possibile usare testatine differenziate per pagine sinistre e destre,
- i libri sono scritti in questo modo (e dunque ci si aspetta che chi legge la tesi sia abituato a questo layout).

Se ad esempio si vuole avere la tesi con dimensione del corpo 11 pt, stampata fronte-retro su fogli A4, con collocazione della prima pagina dei capitoli su facciate destre, dovremo usare il comando

```
\documentclass[11pt,a4paper,twoside,%
openright]{book}
```

Alternativamente può essere utilizzata la classe `memoir` che risulta particolarmente flessibile e permette di personalizzare molti aspetti del documento (testatine, titoli capitoli, note, indici, ecc.) senza dover caricare altri pacchetti; si rimanda alla documentazione della classe per i dettagli.

2 Organizzazione dei files

La gestione di documenti articolati come un libro o una tesi di laurea può diventare complessa e dunque è auspicabile suddividere il testo in più file. LATEX 2_ε permette di avere un *main file* che viene compilato ed in cui sono richiamati gli altri file con i comandi `\include` e `\input`. Il comando `\input{nomefile}` permette il *nesting*, ovvero rende possibile richiamare un file che ne richiama a sua volta un altro. Il comando `\include{nomefile}` non permette il *nesting*, ma inserisce un `\clearpage` prima del testo che contiene e permette di utilizzare il comando `\includeonly{nomefile1,nomefile2,...}` per inserire solo i file specificati tra parentesi. Quando si usa `\includeonly{nomefile1,nomefile2,...}` vengono compilati solamente i file tra parentesi graffe ed i contatori (numeri di pagina, numeri di note, ecc.) non vengono aggiornati.

3 Sezioni della tesi

L'organizzazione della tesi di laurea è argomento dei manuali di scrittura ECO (1977); LESINA

6. Un comportamento comune a molti laureandi consiste nell'usare qualunque strumento tipografico possibile per aumentare il numero di pagine della tesi (allargando i margini, aumentando la dimensione del font, aumentando l'interlinea, inserendo molte figure, stampando solo fronte, ecc.). Tralasciando il fatto che la qualità dei contenuti è più importante della quantità, spesso questi espedienti producono dei risultati tipografici pessimi. Si consiglia dunque di concentrarsi sui contenuti e lasciar perdere l'impostazione tipografica (a questo pensa il LATEX) ed in particolare il numero di pagine prodotto.

(1994); MATRICCIANI (2000, 2003) ed in particolar modo della normativa ISO relativa alla presentazione dei rapporti scientifici e tecnici 5966 (1989). In questo paragrafo si propone una possibile struttura per la tesi e si affrontano le problematiche relative ad ogni sezione.

Una tesi può in generale presentarsi con la seguente struttura:⁷

- | | | |
|---|---|-------------|
| <ul style="list-style-type: none"> • Il frontespizio° • La dedica*° • Il sommario*° • I ringraziamenti*° • Gli indici° | } | frontmatter |
| <ul style="list-style-type: none"> • I simboli e le notazioni* • La prefazione* | } | mainmatter |
| <ul style="list-style-type: none"> • I capitoli interni • Le appendici* | } | mainmatter |
| <ul style="list-style-type: none"> • La bibliografia • L'elenco degli acronimi* • L'indice analitico* | } | backmatter |

3.1 Il frontespizio

La struttura ed il contenuto del frontespizio sono generalmente imposti dalla facoltà presso cui la laurea è conseguita, dunque è necessario crearlo *ad hoc*. Spesso il frontespizio è costituito da due pagine, una in cui compare solo il nome del candidato ed un'altra in cui è presente anche quello dei relatori e dei correlatori, accanto ai quali vengono apposte le rispettive firme. Per la creazione di queste pagine si faccia riferimento ai comandi standard del LATEX 2_ε BECCARI (1991); DILLER (1999); FLYNN (2005); GOOSSENS *et al.* (1995); GRÄTZER (1999); KNUTH (1992); KOPKA e DALY (1995); LAMPORT (1994); OETIKER *et al.* (2000). Alcuni spunti su come impostare il codice per generare il frontespizio sono disponibili in rete.⁸

Per ottenere il logo dell'università sullo sfondo, come in fig. 1(a) a fronte, può essere utilizzato il pacchetto `eso-pic` e definito il comando

```
\newcommand\AlCentroPagina[1]{%
\AddToShipoutPicture*{\AtPageCenter{%
\makebox(0,0){\includegraphics%
[width=0.9\paperwidth]{#1}}}}
```

7. Il simbolo * contraddistingue le sezioni facoltative mentre ° indica che le sezioni non devono essere presenti nell'indice.

8. http://zoonek.free.fr/LaTeX/LaTeX_samples_title/0.html

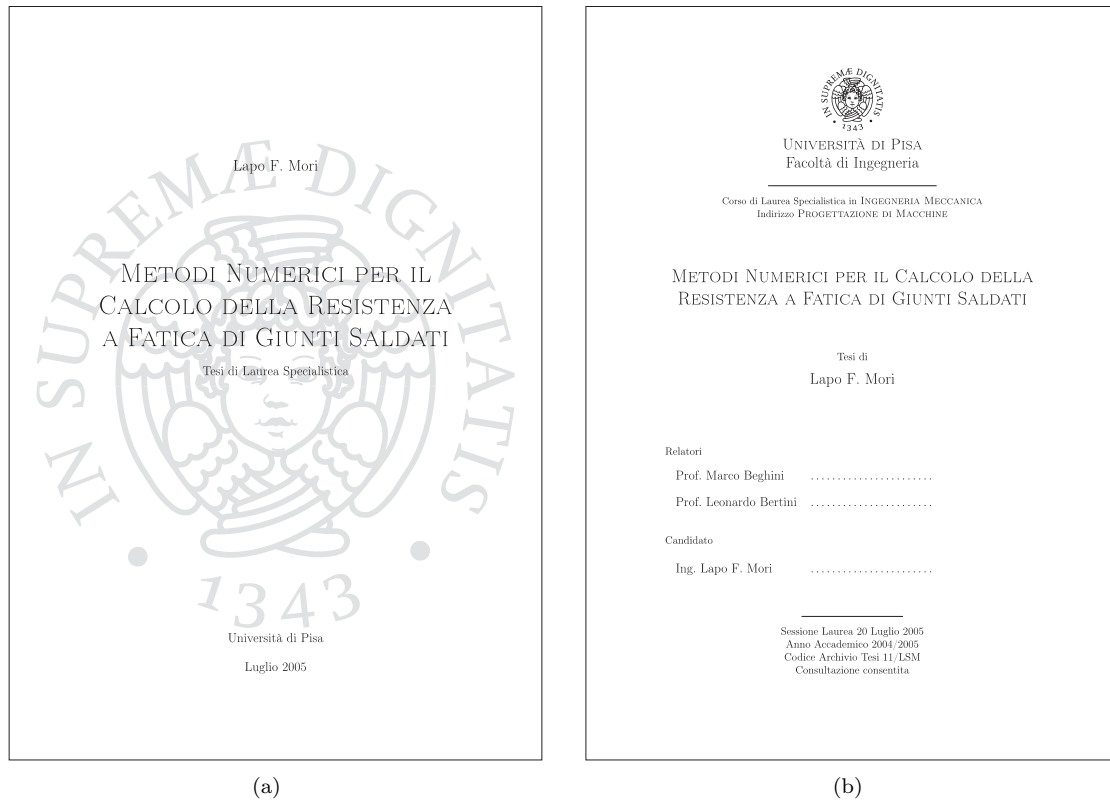


FIGURA 1: Esempio di frontespizio.

che si usa così

```
\AtCentroPagina{nome_logo}
```

I puntini su cui fare la firma (vedi fig. 1(b)) possono essere ottenuti con il comando `\dotfill`.

Il pacchetto `titling` permette di modificare il *type-setting* prodotto con il comando `\maketitle`, ma spesso il frontespizio di una tesi di laurea differisce troppo da quello prodotto dalle classi standard ed è quindi necessario ridefinirlo completamente.

3.2 La dedica

La dedica, ove presente, può assumere le più svariate forme a seconda dei gusti dell'autore. Di solito (vedi ad esempio la fig. 2 nella pagina seguente) è costituita da una riga allineata a destra ad esempio con il comando

```
\begin{flushright}
...
\end{flushright}
```

La posizione verticale della riga nella pagina può essere scelta a piacere e per controllarla risulta particolarmente conveniente l'uso di una coppia di comandi `\vspace{\stretch{...}}`. In questo modo è infatti possibile impostare il rapporto tra lo spazio che precede la dedica e quello che segue. Se ad esempio si vuole che lo spazio che segue sia il doppio di quello che precede, è possibile usare i comandi

```
\null\vspace{\stretch{1}}
...
\vspace{\stretch{2}}\null
```

3.3 Il sommario

Il sommario o *abstract* viene generato con l'ambiente

```
\begin{abstract}
...
\end{abstract}
```

che è definito nelle classi `article` e `report`. Se si utilizza la classe `book` è necessario inserire nel preambolo la definizione di tale ambiente (si riporta la definizione presente nella classe `report`)⁹

```
\usepackage{fancyhdr}
\pagestyle{empty}

\newenvironment{abstract}%
{\cleardoublepage\null\vfill\begin{center}%
\bfseries\abstractname\end{center}}%
{\vfill\null}
```

Per le tesi di laurea in italiano è spesso richiesto che sia presente anche la traduzione inglese dell'abstract. Utilizzando il pacchetto `babel` è possibile selezionare la lingua per le due versioni dell'abstract in modo che sia effettuata la corretta sillabazione delle parole e che sia caricato in automatico il corretto titolo del sommario. Dopo aver richiamato il pacchetto nel preambolo con il comando

```
\usepackage[english,italian]{babel}
```

è sufficiente inserire i sommari come segue

```
\begin{abstract}
... versione del sommario in italiano ...
```

9. Per l'utilizzo del pacchetto `fancyhdr` si rimanda al par. 6.2.1.

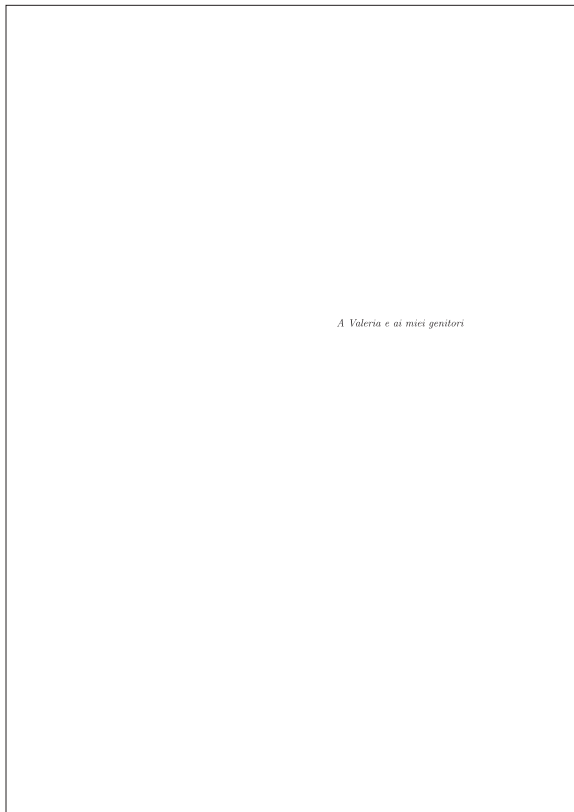


FIGURA 2: Esempio di dedica.

```
\end{abstract}
\selectlanguage{english}%
\begin{abstract}
... English version of the abstract ...
\end{abstract}
\selectlanguage{italian}%
```

Il risultato è riportato in fig. 3 nella pagina successiva.

3.4 Gli indici

Gli indici di solito sono posizionati subito dopo il sommario nel seguente ordine:

- indice
- elenco delle figure
- elenco delle tabelle
- altri elenchi

e vengono prodotti automaticamente da L^AT_EX 2_ε con i comandi

```
\tableofcontents
\listoffigures
\listoftables
```

Per creare elenchi di oggetti flottanti personalizzati (ad esempio listati di programmi, algoritmi, ecc.) si faccia riferimento al pacchetto `float` ed ai relativi comandi `\newfloat` e `\listof`. Per modificare il layout degli indici è possibile utilizzare il pacchetto `tocloft`.

3.5 I simboli e le notazioni

Talvolta risulta opportuno far precedere al testo della tesi un elenco dei simboli e delle notazioni utilizzate. A questo scopo può essere utilizzato il pacchetto `nomencl` che genera l'elenco automaticamente per mezzo del programma `MakIndex`. Alternativamente è possibile creare manualmente l'elenco ad esempio utilizzando l'ambiente `tabular`. In fig. 4 nella pagina 32 si riporta un esempio.

3.6 Le appendici

Le appendici sono dei normali capitoli la cui numerazione è però in lettere latine. L^AT_EX 2_ε permette di crearle semplicemente con il comando `\chapter{...}` preceduto da `\appendix`; se si hanno più appendici, `\appendix` deve essere richiamato solo una volta. Si riporta un esempio:

```
...
\mainmatter
\include{capitolo1}
\include{capitolo2}
\include{capitolo3}

\appendix
\include{appendice1}
\include{appendice2}
...
```

3.7 L'indice analitico

L'indice analitico può essere creato in automatico per mezzo del pacchetto `makeidx` e del programma `MakIndex`. Nel preambolo deve essere presente anche il comando `\makeindex`. Per bilanciare le colonne dell'ultima pagina dell'indice analitico è sufficiente inserirlo all'interno di un ambiente `multicols`.¹⁰ A questo scopo è possibile ridefinire l'ambiente `theindex` con il seguente codice

```
\let\orgtheindex\theindex
\let\orgendtheindex\endtheindex
\def\theindex{%
  \def\twocolumn{\begin{multicols}{2}}%
  \def\onecolumn{}%
  \clearpage
  \orgtheindex
}
\def\endtheindex{%
  \end{multicols}%
  \orgendtheindex
}
```

3.8 La bibliografia

Così come l'indice analitico, anche la bibliografia può essere gestita in modo automatico da L^AT_EX 2_ε. Essa può essere creata manualmente per mezzo dell'ambiente `thebibliography`, tuttavia è altamente preferibile l'utilizzo del programma B_IB_TE_X che permette di separare il contenuto dei riferimenti bibliografici (archiviati in appositi *databases* `.bib`) e la loro formattazione (gestita da stili `.bst`). Il file `.bst` sono semplici file di testo e possono essere creati manualmente, tuttavia si consiglia di

10. Deve essere caricato il pacchetto `multicol`.

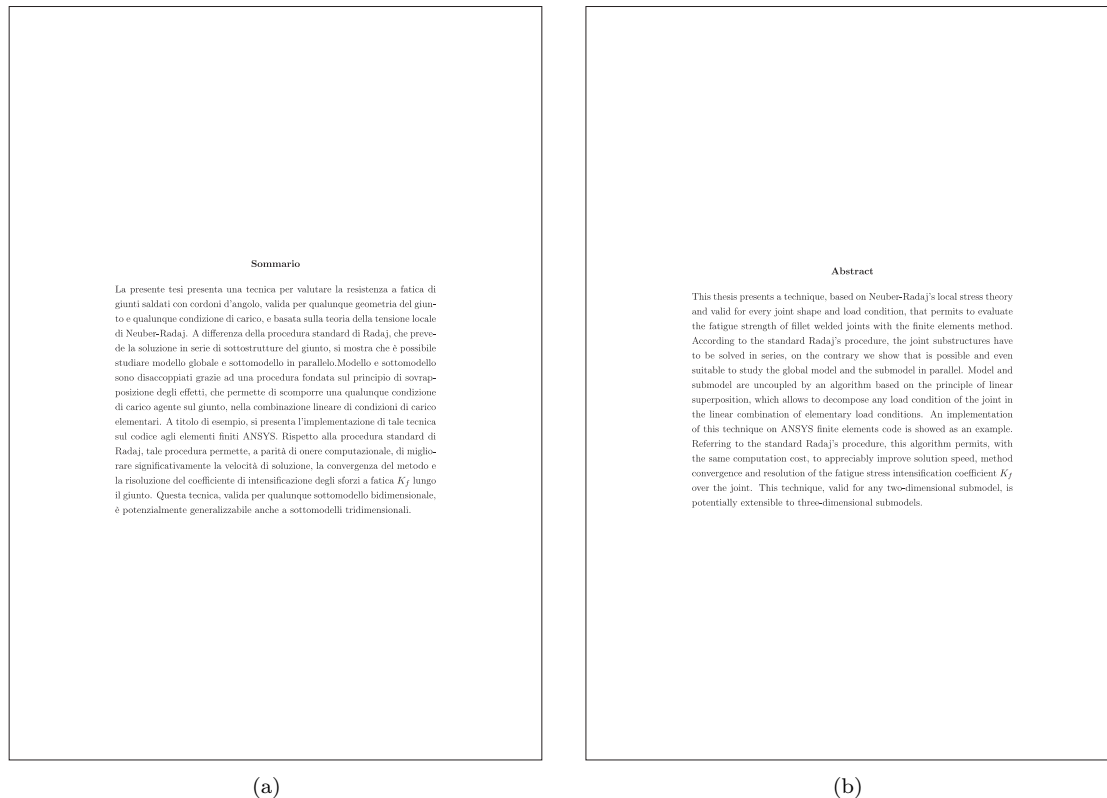


FIGURA 3: Esempio di abstract in doppia lingua.

utilizzare software dedicati alla gestione delle bibliografie: tra questi uno dei migliori è JabRef¹¹ che gira su Java VM e quindi funziona su tutte le piattaforme (Windows, Linux and Mac OS X); per Mac OS X si segnala BibDesk.¹²

Un pacchetto particolarmente utile per la gestione della bibliografia è costituito da natbib, che è accompagnato da una esaustiva guida ed al quale si può affiancare lo stile natbib_ita.bst per bibliografie in italiano.¹³ Qualora sia necessario inserire degli indirizzi di siti web nella bibliografia è consigliabile l'utilizzo del pacchetto url. Il comando `\url{}` offerto da tale pacchetto è una forma di `\verb` che permette l'a capo in corrispondenza di certi caratteri che tipicamente compaiono degli indirizzi web ed email (ad esempio / e .). Un esempio di utilizzo è presente a pagina 43.

La distribuzione standard di $\text{\LaTeX}$ e il pacchetto natbib offrono numerosi stili bibliografici (file .bst), tuttavia esiste un modo facile e veloce di crearsi uno stile personalizzato: il programma makebst, disponibile su CTAN.¹⁴ È sufficiente compilare con $\text{\LaTeX}$ il file makebst.tex e rispondere in modo interattivo alle domande. Alla fine viene creato un file .dbj che va rinominato .tex e compilato con $\text{\LaTeX}$ per ottenere il .bst.

11. <http://jabref.sourceforge.net/>
 12. <http://bibdesk.sourceforge.net/>
 13. natbib_ita.bst è scaricabile da http://www.guit.sssup.it/downloads/natbib_ita.zip.
 14. <http://www.ctan.org/>

Per far comparire la bibliografia nell'indice (`\tableofcontents`) è necessario utilizzare il comando `\addcontentsline`. Il seguente codice visualizza i riferimenti bibliografici con lo stile plain, aggiunge all'indice la sezione (nel caso di una tesi è un capitolo) della bibliografia e richiama il database tesibib.bib; si noti che viene utilizzato il comando `\bibname` per rendere il nome della bibliografia dipendente dalla lingua utilizzata.¹⁵

```
\cleardoublepage
\bibliographystyle{plain}
\refstepcounter{chapter}
\addcontentsline{toc}{chapter}{\bibname}
\bibliography{tesibib}
```

4 Gli oggetti

4.1 Le figure

Le figure sono uno degli argomenti trattati più estesamente dalle guide; esistono anche guide CAUCCI e SPADACCINI (2005); DALY (1998); RECKDAHL (1997) e libri GOOSSENS *et al.* (1997) interamente dedicati all'argomento e a cui si rimanda il lettore per ogni approfondimento. I problemi incontrati dagli utenti $\text{\LaTeX} 2_{\epsilon}$ durante l'inserimento di figure sono generalmente di due tipi. Una parte dei problemi derivano dalle figure in sé, ovvero dal file che si cerca di inserire in un documento (verrà trattato nel par. 4.1.1), mentre un altro tipo

15. Con l'opzione *italian* di babel `\bibname` produce "Bibliografia" mentre con l'opzione *english* produce "Bibliography".

Teoria della tensione locale		
K_f	coefficiente di intensificazione degli sforzi (o di intaglio) a fatica; il pedice <i>el</i> significa che ci si riferisce ad una condizione di carico elementare (ovvero appartenente alla base), mentre il pedice <i>eq</i> significa equivalente	
K_t	coefficiente di intaglio	
λ	lunghezza di supporto microstrutturale	(mm)
ρ_0	raggio dell'intaglio reale	(mm)

FIGURA 4: Esempio di elenco dei simboli.

di problemi, totalmente distinto dal precedente, è quello degli oggetti flottanti (e verrà trattato nel par. 4.3).

4.1.1 Formati

Esistono due grandi classi di figure, le immagini vettoriali e le immagini bitmap. Le prime sono descritte da forme e possono essere scalate e/o deformate senza perdere definizione; sono soprattutto adatte per i grafici e per gli schemi. Le seconde sono matrici di pixel colorati e sono adatte per le fotografie.

La prima cosa da fare è produrre figure nel formato più adatto per i propri scopi. È inutile salvare grafici o schemi in `.jpeg` per poi convertirli in `.eps`, in quanto la conversione di un'immagine bitmap in `.eps` include semplicemente il file bitmap in una "cornice" `.eps` senza migliorare in alcun modo la qualità. È inutile anche fare la conversione opposta, da file vettoriale a bitmap, perché in questo modo si perdono le informazioni sulla geometria contenuta nella figura e quindi si abbassa la qualità del file.

Il parametro più importante di un `.eps` sono i Bounding Box (BB), che determinano la taglia effettiva dell'immagine e che servono a LATEX 2_ε per calcolare lo spazio da riservare alla figura. Idealmente i BB dovrebbero essere al limite massimo del contenuto dell'immagine, ma spesso i programmi di grafica lasciano grandi bordi bianchi attorno alla figura disegnata. Questo porta spesso a grandi confusioni, perché di fatto LATEX 2_ε sta lasciando alla figura lo spazio corretto, ma visivamente parte di questo è utilizzato per il bordo bianco, quindi la figura appare troppo piccola, non centrata, con eccessivi margini verticali, etc. La prima cosa da verificare è quindi che il nostro programma di grafica generi degli `.eps` con BB corretti. Per farlo basta aprire la figura con Ghostview e attivare la visualizzazione dei BB. Se questi non sono cor-

retti bisogna cercare di configurare correttamente il programma di grafica, ma il problema non ha niente a che vedere con LATEX 2_ε. Nel caso si abbia una o due figure con BB errati questi si possono correggere a mano aprendo il file `.eps` con qualunque editor di testo e modificando i valori dei BB, che si trovano facilmente in una delle prime righe. Se però il problema riguarda molti file bisogna cercare di correggerlo all'origine. Ulteriori dettagli sull'utilizzo delle figure con PDFLATEX 2_ε e sulla conversione da file `.eps` a `.pdf` sono riportati nel par. 5.2.

4.1.2 Pacchetti utili

Per inserire le figure è necessario caricare il pacchetto `graphicx`, della cui guida si consiglia la lettura. Per ottenere sottofigure (vedi ad esempio la fig. 1 nella pagina 29) è necessario caricare il pacchetto `subfig`.¹⁶

In casi semplici non è necessario ricorrere a quest'ultimo pacchetto visto che all'interno degli ambienti `figure` e `table` si può mettere più di un grafico o di una tabella. Se si hanno quindi due o più figure che possono essere raggruppate insieme, scrivendo

```
\begin{figure}[tb]
\includegraphics[width=0.3\textwidth]{fig:a}
\caption{didascalia:a}\label{fig:a}
\hspace{4em}
\includegraphics[width=0.3\textwidth]{fig:b}
\caption{didascalia:b}\label{fig:b}
\end{figure}
```

In questo modo si riduce il numero di oggetti flottanti e se ne facilita l'inserimento.

Al fine di mantenere ordine nei file sorgenti, è consigliabile raccogliere tutte le figure in una o più sottocartelle; se ad esempio tali sottocartelle si chiamano `dir_1` e `dir_2`, è sufficiente inserire nel preambolo

```
\graphicspath{{dir_1/},{dir_2/}}
```

16. Tale pacchetto sostituisce il pacchetto `subfigure` che è stato dichiarato obsoleto dal suo stesso autore.

L'argomento di `\graphicspath` può essere relativo alla cartella dove si trova il file `.tex` che viene compilato (come nell'esempio precedente), oppure può essere un percorso assoluto,¹⁷ come ad esempio

```
\graphicspath{{c:/documenti/tesi/immagini/}}
```

La formattazione delle didascalie può essere convenientemente controllata con il pacchetto `caption`. Un utile lettura sull'argomento è CAUCCI e SPADACCINI (2005).

4.2 Le tabelle

Così come per le figure, anche per le tabelle esistono guide specifiche a cui si rimanda per ogni approfondimento MORI (2006).

Per migliorare la spaziatura dell'ambiente `tabular` standard è possibile utilizzare il pacchetto `ctable`, mentre se si vogliono colorare le righe o le colonne è necessario caricare il pacchetto `xcolor` con l'opzione `table`. Nel caso di tabelle di grandi dimensioni è possibile:

- ridurre la dimensione della tabella effettuando una scalatura, ad esempio con i seguenti comandi:

```
\begin{center}
\resizebox{0.95\textwidth}{!}{%
\begin{tabular}
...
\end{tabular}}
\end{center}
```

- ruotare la tabella di 90° con il pacchetto `rotating`,¹⁸
- spezzare la tabella su più pagine con il pacchetto `supertabular`.

4.3 Controllo degli oggetti flottanti

Spesso gli utenti si lamentano del fatto che L^AT_EX 2_ε sposti le figure (e in generale gli oggetti flottanti) lontano dal punto in cui vengono inserite. Nella maggioranza dei casi questo è dovuto ad un utilizzo erraneo delle opzioni di posizionamento degli oggetti flottanti. Nel presente testo si vuole enfatizzare che alcune scelte devono essere prese nella fase di stesura del testo (par. 4.3.1) mentre altre sono riservate, quando necessarie, alla fase di revisione (par. 4.3.2).

4.3.1 Cosa fare durante la stesura del testo

In primo luogo bisogna accettare il fatto che se L^AT_EX 2_ε sposta un oggetto flottante è perché lo spazio è fisicamente insufficiente, o per considerazioni estetiche. Per esempio L^AT_EX 2_ε non metterà mai una figura seguita da un titolo di sezione e

da un cambio pagina, ma preferirà stampare la sezione e poi la figura, oppure se aggiungiamo un oggetto flottante in fondo ad una pagina, L^AT_EX 2_ε è obbligato a spostarlo almeno nella pagina successiva. Se lo spazio è insufficiente, è inutile cercare di forzare L^AT_EX 2_ε a mettere l'oggetto flottante in tale posizione: se lo spazio fisico non c'è, non possiamo inventarlo.

Per fortuna con un minimo di accortezza L^AT_EX 2_ε fa un ottimo lavoro. Per prima cosa è opportuno utilizzare sempre il posizionamento automatico evitando di aggiungere `\clearpage` o comandi simili: in fase di redazione dovremmo solo concentrarci sui contenuti e non sull'impaginazione. In generale i posizionamenti fatti a mano interferiscono con la complessa routine di L^AT_EX 2_ε per il posizionamento degli oggetti flottanti e portano a risultati peggiori rispetto a quelli di default. Seguendo le semplici indicazioni che seguono, il posizionamento automatico mantiene gli oggetti flottanti vicini al punto di inserimento ed inoltre evita che l'utente si preoccupi continuamente del posizionamento dei float, lasciando più tempo per lavorare sui contenuti.

Una delle origini dei problemi lamentati è l'utilizzo eccessivo dell'opzione `[h]` (che chiede di posizionare la figura nel punto dove compare nel codice): gli oggetti flottanti vengono spesso inseriti con l'opzione `[htbp]` o peggio `[h!t]`. In generale si pensa che questa opzione sia la migliore per mantenere gli oggetti flottanti vicino al punto di inserimento. In realtà può funzionare bene solo quando gli oggetti inseriti sono molto piccoli (dove per piccolo si intende con un'altezza molto inferiore rispetto all'altezza del corpo del testo). Il modo migliore per utilizzare le opzioni di posizionamento è quello di domandarsi in primo luogo se l'oggetto flottante sarà abbastanza piccolo per stare in una pagina di testo o se avrà bisogno di una pagina tutta per sé. Nel primo caso lo introdurremo quindi con un'opzione di posizionamento `[tb]`, nel secondo con `[p]`. Se non ci sono oggetti flottanti in sospenso, nel primo caso L^AT_EX 2_ε potrà spostare l'oggetto subito prima del punto di inserzione (cosa che non può fare se si usa `[h]` prima di `[t]`) o nella pagina immediatamente successiva. Usando invece `[p]` per i grossi oggetti flottanti, questi verranno immediatamente stampati in una pagina dedicata, e non verranno spostati alla fine del capitolo come succede con `[tbp]`. Basta sfogliare un qualunque testo ben impaginato per accorgersi che le figure sono introdotte proprio in questo modo: in generale all'inizio o alla fine della pagina, in una pagina intera se sono grandi, raramente nel corpo del testo se sono davvero piccole. Alcuni utenti sono infastiditi dal fatto che alcuni oggetti flottanti appaiano prima del testo in cui sono citati (ad esempio una figura in alto nella pagina in cui è citata): per risolvere questo problema è possibile utilizzare il

17. Si noti che su sistemi Linux o UNIX il percorso assoluto va specificato per esteso, dato che l'espansione della tilde non funziona: `\graphicspath{{/home/lapo/documenti/tesi/immagini/}}`, quindi, e non `\graphicspath{{~/documenti/tesi/immagini/}}`.

18. Altre tecniche per ruotare immagini e tabelle sono indicate in VOSS (2003b).

pacchetto **flafter** che impedisce agli oggetti flottanti di apparire prima della loro definizione nel testo.

Infine è utile ricordare che L $\TeX$ 2 ϵ riesce a posizionare tutte le figure in modo corretto solo se il rapporto

$$\frac{\text{testo}}{\text{figure}}$$

è sufficientemente alto. Da questo segue che è auspicabile (per altro non solo per fini tipografici) scrivere qualche cosa di interessante piuttosto che riempire le lacune con immagini! Se tale rapporto è troppo basso, può accadere che la compilazione si interrompa e venga restituito il seguente errore

```
! LaTeX Error: Too many unprocessed floats.
```

Questo è dovuto al fatto che L $\TeX$ 2 ϵ ha una certa quantità di memoria dedicata al posizionamento degli oggetti flottanti; se troppi oggetti si accumulano durante la compilazione tale memoria può esaurirsi FAIRBAIRNS (2005). Per risolvere questo problema è possibile utilizzare il pacchetto **placeins** che definisce il comando **\FloatBarrier** che non può essere oltrepassato dagli oggetti flottanti e quindi impone il posizionamento di tutti quelli che sono ancora in memoria. Nel caso che il documento presenti dei posti dove possa essere inserita un'interruzione di pagina, conviene utilizzare **\clearpage**. Tale comando, oltre a creare un'interruzione di pagina, impone il posizionamento di tutti gli oggetti flottanti ancora in memoria in modo analogo a **\FloatBarrier**. Il pacchetto **morefloats** aumenta il numero di oggetti flottanti che possono essere mantenuti in memoria durante la compilazione da 18 a 36. Talvolta alcune riviste richiedono che nelle bozze tutte le figure e tabelle siano riportate alla fine del documento: per fare questo in automatico è possibile utilizzare il pacchetto **endfloat**.

Se tutto ciò non fosse sufficiente, nella fase precedente la stampa, e solamente allora, è possibile intervenire manualmente come spiegato nel paragrafo seguente.

4.3.2 Cosa fare durante la revisione del testo

Nella fase che precede la stampa può essere necessario intervenire manualmente per correggere il posizionamento degli oggetti flottanti (quali ad esempio le figure e le tabelle). A questo riguardo esistono numerosi pacchetti, di cui i più utili sono costituiti da **float**, che permette di forzare il posizionamento dell'oggetto nel punto in cui è situato il relativo ambiente per mezzo dell'opzione **H**, e da **placeins**, che permette di mettere delle barriere invalicabili per gli oggetti flottanti con il comando **\FloatBarrier**.

L $\TeX$ mette a disposizione parametri che controllano gli oggetti flottanti:

\setcounter{topnumber}{...} massimo numero di float in posizione **t** per ogni pagina

\def\topfraction{...} massima frazione di pagina per i float in posizione **t** per ogni pagina

\setcounter{bottomnumber}{...} massimo numero di float in posizione **b** per ogni pagina

\def\bottomfraction{...} massima frazione di pagina per i float in posizione **b** per ogni pagina

\setcounter{totalnumber}{...} massimo numero di float nella stessa pagina

\setcounter{dbltopnumber}{...} massimo numero di float grandi nella stessa pagina

\def\textfraction{...} minima frazione di pagina per il testo

\def\floatpagefraction{...} minima frazione di pagina per i float in posizione **p**

\def\dbltopfraction{...} massima frazione di pagina per i float a piena pagina in composizione a due colonne in posizione **t**

\def\dblfloatpagefraction{...} minima frazione di pagina per i float a piena pagina in composizione a due colonne in posizione **p**

5 Compilare il codice

5.1 Scelta del formato

Il codice L $\TeX$ può essere compilato per ottenere in output file Device-Independent (**.dvi**) oppure file Portable Document Format (**.pdf**).¹⁹ Entrambi i formati hanno pro e contro ed in genere vengono utilizzati con scopi differenti. Il **.dvi** permette la ricerca diretta (facendo doppio click sul codice all'interno dell'editor, il programma di visualizzazione del **.dvi** trova il rispettivo output) ed inversa (facendo doppio click all'interno del programma di visualizzazione del **.dvi**, il cursore viene posizionato sul rispettivo codice all'interno dell'editor), molto utili in fase di elaborazione della tesi. Tuttavia la maggioranza dei visualizzatori di **.dvi** non visualizzano correttamente gli effetti di comandi che modificano le figure quali **\resizebox** e **\rotatebox**²⁰ ed inoltre non sfruttano il pacchetto **microtype** (vedi il par. 5.2). Il **.pdf**, invece, pur non offrendo la ricerca diretta ed inversa,²¹ visualizza correttamente tutti gli effetti del pacchetto

19. In realtà una terza opzione è costituita dai file PostScript (**.ps**) ma questi sono stati di fatto soppiantati dal formato **.pdf** e sono quindi sconsigliati.

20. Le versioni di YAP (visualizzatore di **.dvi** di MiK $\TeX$) successive alla 2.5 risolvono questi problemi.

21. Su macchine Macintosh che usano la distribuzione Mac $\TeX$, la ricerca è eseguibile in entrambi i sensi anche su file **.pdf**.

`graphicx`, sfrutta il pacchetto `microtype`, è un formato molto diffuso anche tra coloro che non usano L^AT_EX, permette di sfruttare i riferimenti ipertestuali forniti dal pacchetto `hyperref`, e permette di restringere l'accesso al documento per mezzo di password.²²

In definitiva si consiglia di usare il `.dvi` durante la stesura della tesi ed il `.pdf` per la stampa e per la distribuzione della tesi in formato elettronico.

5.2 Creazione di PDF

La creazione di un PDF può essere effettuata in differenti modi, di cui i principali sono:

- la conversione di un file `.dvi` o `.ps` attraverso Ghostscript,
- la compilazione diretta del sorgente `.tex` con PDFL^AT_EX 2_ε.

Senza entrare nei dettagli, per i quali si rimanda a testi specifici come PRATESI *et al.* (2003); TALBOT (2004), per sfruttare appieno le potenzialità del formato PDF²³ è necessario utilizzare PDFL^AT_EX 2_ε che è presente in tutte le più importanti distribuzioni di L^AT_EX 2_ε ed è gestito dai principali editor di testo dedicati al L^AT_EX 2_ε.

La differenza fondamentale tra le due procedure consiste nel formato delle immagini inserite con il comando `\includegraphics`: mentre la conversione da file `.dvi` o `.ps` utilizza gli stessi file `.eps` che servono per la compilazione del `.dvi`, PDFL^AT_EX 2_ε richiede file `.pdf` (se vettoriali) oppure `.jpg` o `.png` (se bitmap).²⁴ Volendo utilizzare PDFL^AT_EX 2_ε, è dunque necessario effettuare una conversione preliminare dei file `.eps` in `.pdf`; questa può essere convenientemente effettuata con Ghostscript per mezzo dell'interfaccia grafica `eps2pdf`.²⁵

Se si prevede di compilare sia con L^AT_EX 2_ε che con PDFL^AT_EX 2_ε è consigliabile non specificare l'estensione dei file grafici caricati da `\includegraphics`; volendo ad esempio inserire il file `figura_01.eps`, è necessario creare il `figura_01.pdf` con `eps2pdf` e poi inserire nel testo

```
\includegraphics{figura_01}
```

22. È possibile limitare l'accesso al file, limitarne la stampa (impedirla, oppure consentirla solo a bassa risoluzione) e limitarne le modifiche (estrazione di testo, estrazione o eliminazione di pagine, ecc.).

23. Il formato PDF permette di utilizzare l'ipertesto, i `bookmark` e i `thumbnail` che non vengono prodotti nella conversione a partire da file `.dvi` o `.ps`.

24. Per i dettagli riguardo ai formati delle figure si rimanda al par. 4.1.1.

25. `eps2pdf` è un'interfaccia grafica per Windows disponibile su CTAN (<http://www.ctan.org/tex-archive/support/eps2pdf/>). Su Linux esiste una omonima procedura scritta in `sh` ed anche la procedura `epstopdf`: entrambe operano da linea di comando. Tutti i programmi citati azionano GhostScript per effettuare la conversione da `.eps` a `.pdf`.

Così facendo, quando si compila con L^AT_EX 2_ε viene caricato il file `figura_01.eps`, mentre quando si compila con PDFL^AT_EX 2_ε viene caricato il file `figura_01.pdf`.

Per abilitare i riferimenti ipertestuali all'interno del file PDF²⁶ è necessario caricare il pacchetto `hyperref`. Per approfondire come utilizzare font TrueType con T_EX (L^AT_EX) e PDFT_EX (PDFL^AT_EX) si consiglia di visitare il sito <http://www.radamir.com/tex/ttf-tex.htm>. Si consiglia inoltre di utilizzare il pacchetto `microtype` che migliora il riempimento delle righe abilitando:

- la *font expansion*: espande secondo l'asse orizzontale i caratteri per ottenere un riempimento ottimale della riga;
- la *character protrusion*: permette che certi caratteri sporgano leggermente a fine riga (tipicamente segni di punteggiatura e trattini).

Per utilizzare queste due modalità, è sufficiente caricare il pacchetto senza specificare alcuna opzione

```
\usepackage{microtype}
```

6 Pacchetti utili

6.1 La lingua italiana

6.1.1 Norme tipografiche

In italiano la maggioranza delle regole tipografiche non sono universali e vincolanti, ma dipendono piuttosto da convenzioni e abitudini o dal gusto dell'autore. Nonostante questo, è importante che l'autore della tesi conosca quali sono le principali "norme" tipografiche italiane. CEVOLANI (2006) ne offre una sintesi e, per ogni regola, mostra come applicarla in L^AT_EX.

6.1.2 La sillabazione

Per attivare la sillabazione italiana e caricare i nomi delle sezioni²⁷ in lingua italiana, è necessario caricare il pacchetto `babel` con l'opzione per l'italiano per mezzo del comando

```
\usepackage[italian]{babel}
```

L'utilizzo di `babel` è necessario per avere la sillabazione italiana ma non sufficiente in quanto serve anche che il file `ithyph.tex` sia attivato (riferirsi al manuale della distribuzione L^AT_EX 2_ε che si usa).²⁸

L^AT_EX 2_ε sillaba correttamente quasi tutte le parole italiane, tuttavia esistono casi in cui si utilizzano nomi propri oppure parole rare; in questa

26. Anche alcuni visualizzatori di file `.dvi` supportano i riferimenti ipertestuali.

27. Ad esempio sommario, bibliografia, indice, ecc.

28. A titolo di esempio si riporta la procedura per attivare il file `ithyph.tex` su MiKTeX: dal pannello "languages" su "MiKTeX options" attivare la voce "italian - ithyph.tex" e poi rigenerare il file di formato con "Update Formats" nella finestra "General".

eventualità, se la sillabazione tentata da L^AT_EX 2_ε non è soddisfacente, è possibile suggerirla con il comando `\hyphenation` (va posizionato nel preambolo): si devono scrivere le parole sillabate tra parentesi graffe, separate da uno spazio, come nel seguente esempio

```
\hyphenation{sil-la-ba-zio-ne sim-pa-ti-ca}
```

Il precedente comando può anche essere utilizzato quando si vuole che alcune parole *non* vengano sillabate: è sufficiente scriverle senza trattini come nel seguente esempio

```
\hyphenation{MATLAB Mathematica}
```

Tale comando può anche essere utilizzato per forzare una sillabazione particolare; se ad esempio si vuole che la parola “melograno” sia spezzata tra “melo” e “grano” e non in altri punti, è sufficiente scrivere:

```
\hyphenation{melo-grano}
```

Se la parola in questione compare una sola volta, è possibile suggerirne la sillabazione direttamente nel testo con `\-`; avremo ad esempio

```
sil\ -la\ -ba\ -zio\ -ne
```

Vale la pena di segnalare che con l'opzione *italian* di *babel* il carattere “ è attivo e serve per svolgere una serie di funzioni, tra le quali:

- introdurre una possibile cesura disabilitando le altre cesure “troppo vicine” ma consentendo la sillabazione di entrambi i monconi della parola; per esempio *dispepsia* viene divisa naturalmente in *di-ssep-sia* mentre *dis"pepsia* viene divisa in *dis-pep-sia*;
- andare a capo negli URL e nelle parole composte in cui i componenti sono separati da una barra; `input/output` spontaneamente non sarebbe divisibile ma con il segno “ lo diventa, e lo diventano i singoli monconi.

A conclusione di questo paragrafo, è doveroso ricordare che gli interventi manuali sulla sillabazione dovrebbero sempre essere fatti nella fase di revisione che precede immediatamente la stampa. Spesso è preferibile riformulare una frase che dà luogo ad un errore di *overfull* piuttosto che imporre particolari sillabazioni.

6.1.3 Il rientro sulla prima riga

I libri italiani contemporanei generalmente non hanno il primo capoverso dopo il titolo di sezione rientrato, tuttavia alcuni autori preferiscono avere tale rientro. Per attivare il rientro sulla prima riga di ogni sezione, sottosezione, ecc. è necessario caricare il pacchetto *indentfirst*, in quanto la convenzione anglosassone (di default su L^AT_EX 2_ε) non lo prevede. Si veda la fig. 5 a fronte.

6.1.4 Caratteri accentati

Su L^AT_EX 2_ε i caratteri accentati possono essere introdotti con i comandi standard²⁹ `\‘{e}`, `\’{e}`, ecc. oppure direttamente da tastiera è, é, ecc. se nel preambolo si carica il pacchetto *inputenc* con la appropriata codifica. Le opzioni di tale pacchetto dipendono dall’editor che si sta usando. *[ansinew]* va usato con la maggioranza degli editor su Windows (tra cui WinEdt e T_EXnik Center); *[latin1]* o *[latin9]* con la maggioranza degli editor su piattaforme Linux, UNIX, e Mac OS X; *[applemac]* su macchine Macintosh con sistema operativo precedente all’OS X e su OS X stesso a seconda della codifica utilizzata nell’editor;³⁰ *[utf8x]* è accettabile solo con editor adeguati, che lo consentono come opzione, tra cui Kile su Linux e T_EXShop su Mac OS X.

6.2 Il layout

6.2.1 Le testatine ed i piè di pagina

Per personalizzare testatine e piè di pagina è possibile usare il pacchetto *fancyhdr*. Per una tesi è probabile che si abbiano impostazioni differenti a seconda della sezione e dunque è conveniente definirsi alcuni comandi che modifichino testatine e piè di pagina; un esempio per *frontmatter* e *mainmatter* potrebbe essere:

```
\newcommand{\fncyfront}{%
  \fancyhead[R0]{\footnotesize\rightmark}
  \fancyfoot[R0]{\thepage}
  \fancyhead[LE]{\footnotesize\leftmark}
  \fancyfoot[LE]{\thepage}
  \fancyhead[RE,L0]{}
  \fancyfoot[C]{}
  \renewcommand{\headrulewidth}{0.3pt}}
\newcommand{\fncymain}{%
  \fancyhead[R0]{\footnotesize\rightmark}
  \fancyfoot[R0]{\thepage}
  \fancyhead[LE]{\footnotesize\leftmark}
  \fancyfoot[LE]{\thepage}
  \fancyfoot[C]{}
  \renewcommand{\headrulewidth}{0.3pt}}
```

da utilizzare nel seguente modo

```
\pagestyle{fancy}
\fncyfront
\frontmatter
...
\fncymain
\mainmatter
```

La definizione di tali comandi è differente a seconda che il testo sia fronte-retro (*twoside*) o solo fronte (*oneside*).

Utilizzando l’opzione *openright* è possibile avere una pagina bianca alla fine del capitolo; per

29. Per i comandi standard si faccia riferimento a PAKIN (2003).

30. *[applemac]* corrisponde alla codifica MacOSRoman, o Mac occidentale, di default sul Mac sia OS 9 che OS X. Tuttavia è possibile utilizzare anche altre codifiche, a seconda dell’editor utilizzato. Per esempio T_EXShop permette di salvare file in qualunque codifica (MacOSRoman di default, ma anche Latin1, Latin9, Unicode, e tutte le altre). La codifica può essere impostata nelle preferenze dell’editor. Se su piattaforma Macintosh si usa software di derivazione *nix, come Emacs, di solito i file vengono salvati con codifica *[latin1]*.

Problema

Il metodo appena presentato nel par. 2.5.1 presenta un serio problema di fondo. Esso è infatti valido nell'ipotesi che l'insieme di vettori $\hat{\Pi}$ costituisca una base di $\mathbb{R}^6$, ovvero che essi siano linearmente indipendenti. Tuttavia è facile dimostrare che $\det \hat{\Pi} = 0$ indipendentemente dai valori assegnati alle sei coppie di elementi non nulli della matrice. Conseguenza immediata è che la matrice $\hat{\Pi}$ non è invertibile e che dunque non è possibile risolvere il problema con questa via.

Conclusioni

Interessante è individuare l'origine della lineare dipendenza dei vettori di $\hat{\Pi}$. Come già osservato, l'insieme di vettori Π è un buon candidato a divenire base di $\mathbb{R}^6$, purché nella scelta dei 12 elementi non nulli si rispetti la condizione (2.4).

Tale condizione viene dunque a mancare nel momento in cui si impongono i vincoli (2.5); in altre parole, il metodo viene a fallire quando si impone che le condizioni di carico di base siano una ad una equilibrate. D'altronde il vincolo di condizioni di carico di base equilibrate è imprescindibile dato che altrimenti risulta impossibile risolvere con metodi numerici agli EF il sottomodello. È dunque necessario trovare un'altra strada per risolvere il problema della scomposizione della condizione di carico generica applicata al giunto Φ .

(a) senza pacchetto indentfirst

Problema

Il metodo appena presentato nel par. 2.5.1 presenta un serio problema di fondo. Esso è infatti valido nell'ipotesi che l'insieme di vettori $\hat{\Pi}$ costituisca una base di $\mathbb{R}^6$, ovvero che essi siano linearmente indipendenti. Tuttavia è facile dimostrare che $\det \hat{\Pi} = 0$ indipendentemente dai valori assegnati alle sei coppie di elementi non nulli della matrice. Conseguenza immediata è che la matrice $\hat{\Pi}$ non è invertibile e che dunque non è possibile risolvere il problema con questa via.

Conclusioni

Interessante è individuare l'origine della lineare dipendenza dei vettori di $\hat{\Pi}$. Come già osservato, l'insieme di vettori Π è un buon candidato a divenire base di $\mathbb{R}^6$, purché nella scelta dei 12 elementi non nulli si rispetti la condizione (2.4).

Tale condizione viene dunque a mancare nel momento in cui si impongono i vincoli (2.5); in altre parole, il metodo viene a fallire quando si impone che le condizioni di carico di base siano una ad una equilibrate. D'altronde il vincolo di condizioni di carico di base equilibrate è imprescindibile dato che altrimenti risulta impossibile risolvere con metodi numerici agli EF il sottomodello. È dunque necessario trovare un'altra strada per risolvere il problema della scomposizione della condizione di carico generica applicata al giunto Φ .

(b) con pacchetto indentfirst

FIGURA 5: Rientro sulla prima riga.

evitare che in questa pagina siano presenti testatine o piè di pagina, è sufficiente aggiungere nel preambolo i seguenti comandi

```
\makeatletter
\def\cleardoublepage{\clearpage\if@twoside
\ifodd\c@page
\else\hbox{}\thispagestyle{empty}\newpage
\if@twocolumn\hbox{}\newpage\fi\fi\fi}
\makeatother
```

In alternativa a `fancyhdr` è possibile usare `ncc-fancyhdr`. Tale pacchetto offre gli stessi strumenti di `fancyhdr` ma ne risolve alcuni problemi.

6.2.2 Il layout della pagina

Molto di frequente i regolamenti di facoltà richiedono un layout della pagina differente da quello prodotto di default dalle classi di L^AT_EX 2_ε ed è dunque necessario modificarlo. Il primo modo per intervenire è l'utilizzo di comandi interni del L^AT_EX 2_ε, quali `\textwidth`, `\oddsidemargin`, ecc., tuttavia questa strada è sconsigliabile per molte ragioni FAIRBAIRNS (2005); TRETTIN (2005). Una migliore soluzione è costituita da pacchetti rigidi, come `layaureo`,³¹ che sono di facile utilizzo³² ma che lasciano all'utente una scarsa possibilità di personalizzare il layout. Nel caso in cui il layout prodotto dai pacchetti rigidi non soddisfi le esigenze, è possibile utilizzare il pacchetto `geometry` che è completamente configurabile. Nel caso che siano necessari degli interventi locali a pagine o a paragrafi è possibile utilizzare il pacchetto `chnpage`.

Per rilegare la tesi può essere conveniente indicare sulle pagine dove tagliare il foglio; questo può essere agevolmente realizzato utilizzando in coppia i pacchetti `geometry` e `crop`. Si veda ad esempio la fig. 6 nella pagina successiva.

Di default L^AT_EX 2_ε cerca di coprire interamente l'altezza della pagina e, ove necessario, inserisce degli spazi aggiuntivi tra i capoversi oppure dilata gli spazi tra le voci degli elenchi puntati e così via. Se si vuole disattivare questa impostazione ed avere dello spazio bianco a piè di pagina quando non si riesce a coprirla tutta, è sufficiente aggiungere nel preambolo il comando `\raggedbottom`. Il comportamento di default è invece dovuto al comando `\flushbottom`. Per migliorare la copertura delle pagine è possibile permettere che siano spezzati ambienti matematici di tipo `display` aggiungendo al preambolo il comando `\allowdisplaybreaks`.

È conveniente non modificare il comportamento di default di L^AT_EX 2_ε fino a quando non si arriva alla versione definitiva del testo (che precede immediatamente la stampa). Solo in questa fase è

31. In aggiunta a `layaureo`, esistono altri pacchetti rigidi, tra cui `widemargins`, `a4` e `a4wide`, ma se ne sconsiglia l'utilizzo perché essi sono obsoleti e circolanti in versioni fra loro incompatibili.

32. Il pacchetto `layaureo`, invece, incorpora due modalità di layout (con e senza l'opzione `big`) e permette anche di gestire comodamente lo spazio per la rilegatura per mezzo del comando `binding`.

possibile intervenire modificando il posizionamento degli oggetti flottanti (vedi il par. 4.3), oppure intervenendo con i comandi appena citati. Prima di modificare le impostazioni di L^AT_EX 2_ε è conveniente provare ad effettuare piccole modifiche al testo che spesso sono sufficienti per risolvere i problemi e permettono di ottenere layout più eleganti.

6.2.3 L'interlinea

Spesso i regolamenti di facoltà impongono un'interlinea differente da uno (di default su L^AT_EX 2_ε). Per modificare l'interlinea del documento esistono più strade (si veda al riguardo TRETTIN (2005)), tuttavia la più indicata consiste nel caricare il pacchetto `setspace`. Tale pacchetto fornisce tre interlinee predefinite richiamate con i comandi `\singlespacing` (interlinea singola), `\onehalfspacing` (interlinea 1,5) e `\doublespacing` (interlinea doppia); se è necessaria un'interlinea differente, è sufficiente utilizzare il comando `\linespread{...}`: tra parentesi graffe va il numero che rappresenta il fattore di scala per l'avanzamento di riga.

6.3 Lo stile

6.3.1 I fonts

In primo luogo è consigliabile utilizzare l'encoding T1 che rappresenta il nuovo standard di codifica dei caratteri di L^AT_EX 2_ε; tale codifica non è ancora la predefinita solo per motivi di compatibilità con le versioni di L^AT_EX precedenti a L^AT_EX 2_ε ma può essere attivata per mezzo del comando

```
\usepackage[T1]{fontenc}
```

L'utilizzo di tale encoding può dare luogo a problemi di visualizzazione di file PDF su versioni di Adobe Acrobat precedenti alla 6; in particolare può accadere che i font appaiano sgranati FAIRBAIRNS (2005). La soluzione di questo problema è fornita dalla famiglia di font `cm-super`; i font di tale famiglia sono PostScript Type 1 con praticamente la stessa forma dei font `cm` e `tc`, ma con molte migliorie in termini di qualità di typesetting, tra cui il supporto per set di caratteri non-ASCII.³³ Per maggiori dettagli su questa famiglia si rimanda, oltre che alla relativa documentazione, a LEHMAN (2004). Nel caso che la tesi sia di tipo scientifico, è conveniente abilitare i font matematici forniti dall' $\mathcal{A}\mathcal{M}\mathcal{S}$ con il comando

```
\usepackage{amsfonts}
```

A questo conviene aggiungere anche

```
\usepackage{amsmath}
```

che fornisce svariate estensioni per il miglioramento della struttura informativa e della stampa di documenti che contengono formule matematiche. Per modificare la dimensione del font, in aggiunta ai comandi standard,³⁴ è utile il pacchetto `relsize`

33. <http://en.wikipedia.org/wiki/ASCII>

34. `tiny`, `scriptsize`, `footnotesize`, `small`, `normalsize`, `large`, `Large`, `LARGE`, `huge` e `Huge`.

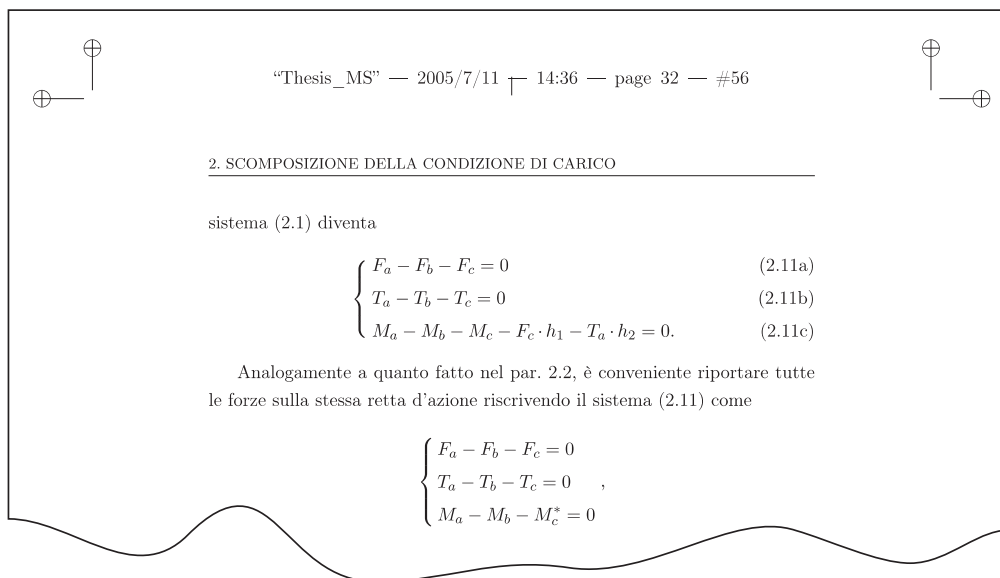


FIGURA 6: Esempio di segni per il taglio del foglio.

che consente di assegnare dimensioni relative con i comandi `\smaller` e `\larger`.

Gli utenti $\text{\LaTeX}$ che provengono da software What-You-See-Is-What-You-Get (WYSIWYG),³⁵ per lo meno all’inizio, sentono l’esigenza di cambiare font. Questo fenomeno è probabilmente dovuto al fatto che i software WYSIWYG hanno una resa tipografica talmente scarsa che l’utente insoddisfatto prova a migliorarla cambiando font. Il mio consiglio è di utilizzare i font che $\text{\LaTeX}$ carica di default, ovvero la famiglia Computer Modern sviluppata dallo stesso inventore del $\text{\TeX}$, Donald Knuth. Se si vuole a tutti i costi cambiare font, è bene ricordare che è necessario scegliere quattro famiglie (Serif, Sans-serif, Typewriter e i font per la matematica) che formino una buona combinazione. A tale proposito, è importante ricordare che i font, tranne alcune eccezioni,³⁶ non hanno tutti i simboli necessari per la matematica e quindi non possono essere usati se non nel testo. Per cambiare font è possibile caricare uno dei numerosi pacchetti dedicati come `pxfonts`, `mathpazo`, `courier`, `eulervm`, `literat`, `lucida`, `pandora`, `mathptmx`, `helvet`, `courier`, `kerkis`, `kmath`, `qpxmath`, `qtxmath`. In alternativa all’uso dei pacchetti, è possibile installare font seguendo le istruzioni di LEHMAN (2004). Un elenco dei font disponibili è riportato sul sito del TUG olandese.³⁷

6.3.2 Il titolo dei capitoli

Per personalizzare il formato dei titoli dei capitoli è possibile utilizzare il pacchetto `fncychap`. In fig. 8 nella pagina 41 è mostrato il formato standard dei capitoli prodotto dalla classe `book`, mentre in

fig. 7 nella pagina successiva è stato utilizzato il comando

```
\usepackage[Lenny]{fncychap}
```

In alternativa anche il pacchetto `quotchap` permette di personalizzare i titoli dei capitoli, seppur offrendo molte meno opzioni del precedente.

6.3.3 Liste

Per personalizzare il tipo di numerazione negli ambienti `enumerate` è possibile utilizzare l’omonimo pacchetto `enumerate` che permette di modificare lo stile del contatore³⁸ e del simbolo delimitatore.³⁹

6.3.4 I “mini indici”

Quando i capitoli hanno una struttura particolarmente complessa, può essere conveniente riportare nella pagina iniziale l’indice del capitolo (vedi ad esempio la fig. 7 nella pagina seguente). Questi “mini indici” possono essere prodotti automaticamente con il pacchetto `minitoc`.

6.3.5 Le epigrafi

Talvolta si vogliono inserire epigrafi nella pagina iniziale dei capitoli. Per farlo è possibile utilizzare il pacchetto `epigraph`; un esempio è riportato in fig. 8 nella pagina 41.

6.3.6 Le note

$\text{\LaTeX} 2_{\epsilon}$ produce di default un layout delle note di alta qualità; esistono tuttavia alcuni accorgimenti per modificarlo, quando lo si ritenga strettamente necessario. Il pacchetto `footmisc` fornisce molti controlli sulle note tra cui la possibilità di forzare le

35. <http://en.wikipedia.org/wiki/WYSIWYG>

36. Il TUG danese ospita una pagina in cui sono riportati tutti i font che supportano la matematica (<http://www.tug.dk/FontCatalogue/mathfonts.html>).

37. <http://www.tug.dk/FontCatalogue/>

38. È possibile ottenere numeri arabi (1,2,3,...), numeri romani maiuscoli (I,II,III,...) e minuscoli (i,ii,iii,...), lettere latine maiuscole (A,B,C,...) e minuscole (a,b,c,...).

39. È possibile scegliere qualunque carattere come delimitatore ottenendo ad esempio 1), 1., 1 −, ...

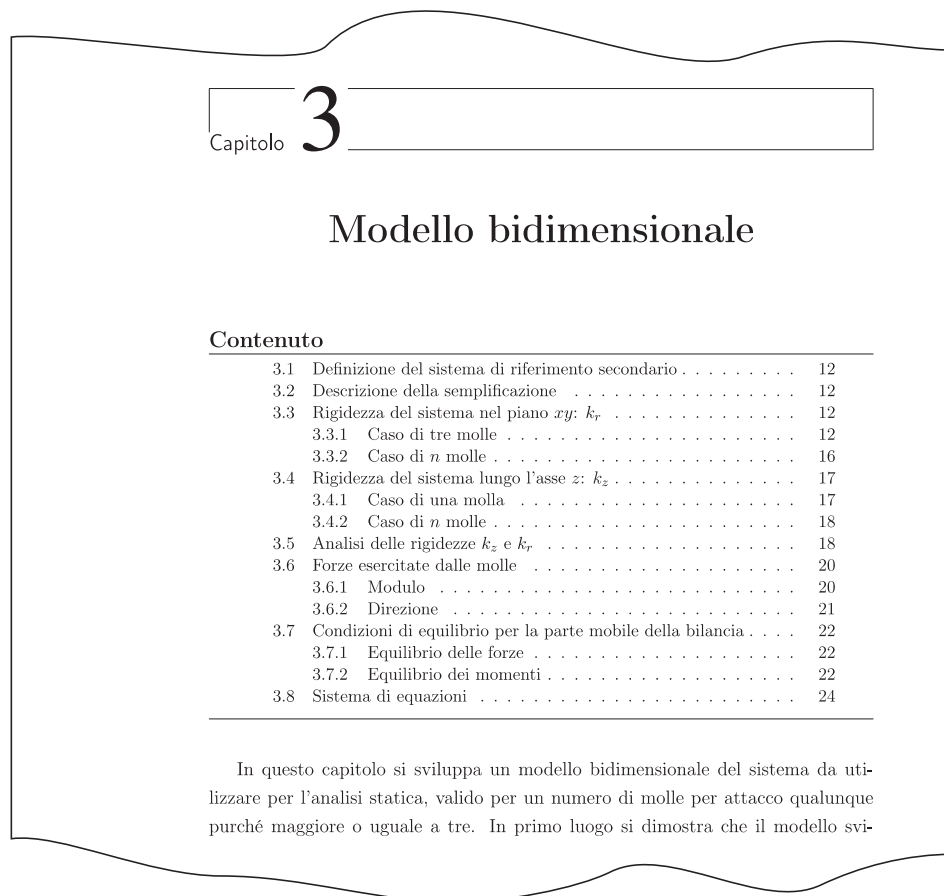


FIGURA 7: Esempio di “mini indice”.

note al fondo della pagina⁴⁰ con l'opzione `bottom`; si veda la fig. 9 nella pagina 42.

Per impedire che le note vengano spezzate su più pagine è sufficiente assegnare al parametro di penalità un valore molto elevato, ad esempio

```
\interfootnotelinepenalty=10000
```

mentre per controllare la dimensione della zona assegnata alle note a piè di pagina si può usare il comando

```
\dimen\footins=2cm
```

6.4 La matematica

6.4.1 I simboli “speciali”

Intendendo con “simboli speciali” tutti quelli che non sono inseribili direttamente dalla tastiera, è necessario distinguere tra quelli matematici e quelli non matematici: per i primi dovrebbe essere sufficiente caricare i simboli dell' $\mathcal{AMS}$ con il pacchetto `amssymb`; per tutti gli altri simboli sono necessari pacchetti appositi che possono essere facilmente identificati grazie a PAKIN (2003).

40. Normalmente L^AT_EX 2_ε unisce le note con l'ultima riga della pagina e dunque su pagine non piene non si hanno le note a fondo pagina.

6.4.2 Rappresentazione dei numeri

Un pacchetto molto utile per la rappresentazione di numeri è `numprint`. Tra le funzioni di tale pacchetto si ricordano l'inserimento di un separatore ogni tre cifre per le migliaia e l'approssimazione automatica. Ad esempio

```
\numprint{2.742647826672E-01}
```

produce

```
2,743 · 10-01
```

6.4.3 Unità di misura

Per evitare di formattare a mano le unità di misura è consigliabile l'utilizzo del pacchetto `SIunits`. Ad esempio

```
\unita{32,1}{\micro\metre}
```

produce

```
32,1 μm
```

Si ricordi che se si utilizza l'opzione `italian` di `babel`, è necessario usare l'opzione `italian` anche su `SIunits`; se si usa il pacchetto `amssymb`, è necessario usare l'opzione `square` su `SIunits` perché il comando `square` viene già definito dal precedente.

6.4.4 Altri pacchetti

Per evidenziare gli ambienti matematici può essere utilizzato il pacchetto `empheq`. Per la personalizza-

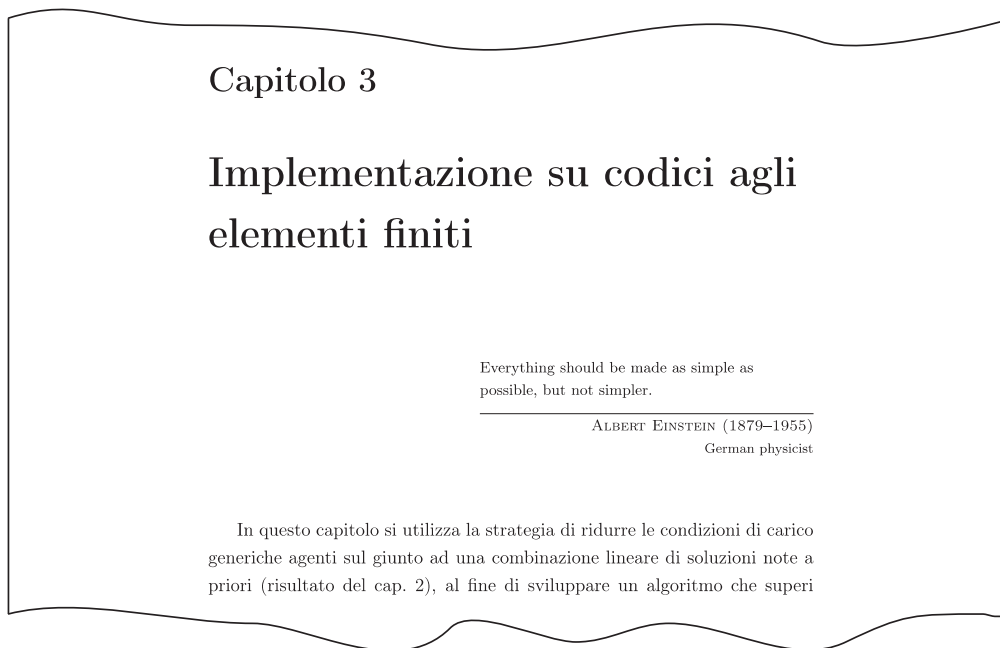


FIGURA 8: Esempio di epigrafe.

zione degli ambienti “tipo teorema” è necessario il pacchetto `theorem`. Il pacchetto `xfrac` permette invece di scrivere correttamente le frazioni nel testo e nel testo matematico (ad esempio: $\frac{5}{7}$).

6.4.5 Sistema di equazioni

Talvolta si ha la necessità di raggruppare un sistema di equazioni con una parentesi graffa. A questo scopo è possibile definire un nuovo ambiente

```
\newenvironment{система}
{\left\lbrace\begin{array}{@{}l@{}}
{\end{array}\right.}
```

Ad esempio il seguente codice

```
\[\begin{система}
x_1=\sigma_b^2-\sigma_a\\
x_2=\sigma_a^2-\sigma_b
\end{система}\]
```

produce

$$\begin{cases} x_1 = \sigma_b^2 - \sigma_a \\ x_2 = \sigma_a^2 - \sigma_b \end{cases}$$

6.5 Gli acronimi

La gestione degli acronimi può essere efficacemente gestita con il pacchetto `acronym`; tra le altre cose, tale pacchetto genera automaticamente i riferimenti ipertestuali tra acronimi nel testo e la loro definizione all'interno dell'elenco degli acronimi. Un esempio di elenco degli acronimi è riportato in fig. 10 nella pagina 43.

6.6 Codici ed algoritmi

L'inserimento di codici all'interno della tesi è facilmente realizzabile con il pacchetto `verbatim`; per avere un controllo più vasto della formattazione del codice è possibile utilizzare il pacchetto `listings` che riconosce numerosi linguaggi di programmazione. In alternativa può essere utilizzato il pacchetto `fancyvrb`, che è però meno flessibile.

Per la formattazione di algoritmi sono invece consigliabili i pacchetti `algorithm` e `algpseudocode`: il primo genera degli oggetti flottanti mentre il secondo no.

6.7 Riferimenti incrociati

In molti casi è comodo usare contemporaneamente i comandi `\ref` e `\pageref` per riferirsi a figure e tabelle, specialmente quando ci sono più pagine tra il riferimento e l'oggetto. Per questo, alcuni utenti utilizzano comandi come

```
\newcommand{\fullref}[1]{%
\ref{#1} a pagina-\pageref{#1}}
```

che semplifica la scrittura del riferimento. Tuttavia, non sapendo a priori dove sia posizionato l'oggetto a cui ci si riferisce, utilizzando un comando del genere può capitare che il `\pageref` punti alla pagina stessa dove si trova il riferimento producendo un risultato insoddisfacente.

Per rendere automatica la scrittura dei riferimenti completi è possibile utilizzare il pacchetto `varioref` che introduce il comando `\vref` da usarsi nello stesso modo del comune `\ref`. Tale pacchetto funziona in parallelo a `babel` e quindi si adatta alla lingua utilizzata nel testo. Ad esempio

```
si veda la fig.-\vref{f5}
```

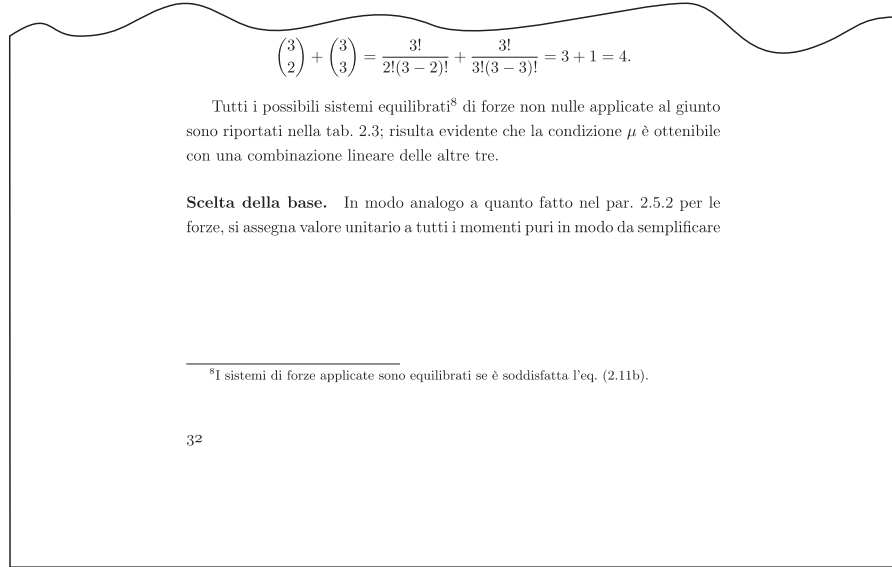
produce, a seconda di dove viene posizionata la figura, qualcosa del tipo

```
si veda la fig. 3.1 nella pagina successiva
```

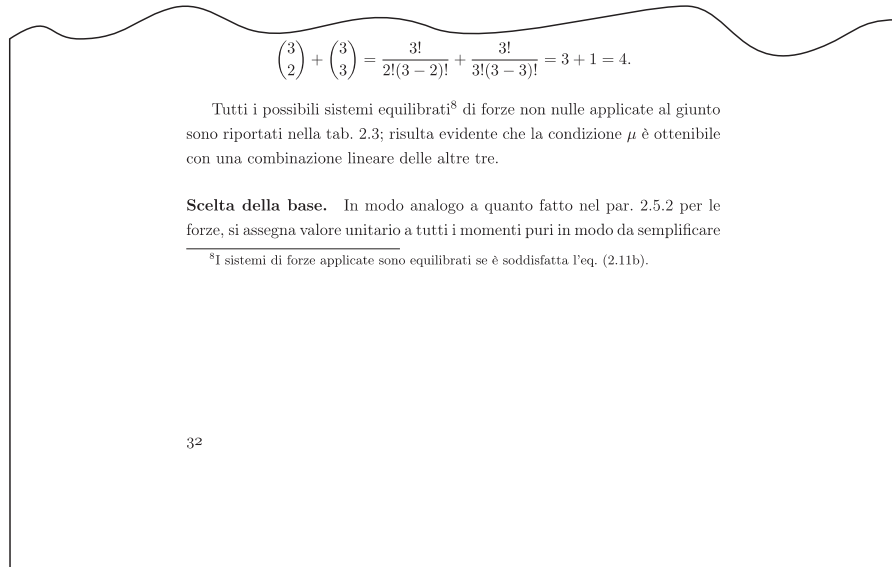
oppure

```
si veda la fig. 3.1 a pagina 24
```

Per quanto riguarda invece il riferimento ad equazioni, è consigliabile utilizzare il comando `\eqref{...}` al posto di (`\ref{...}`). Ad esempio



(a) con opzione `bottom`



(b) senza opzione `bottom`

FIGURA 9: Posizione delle note.

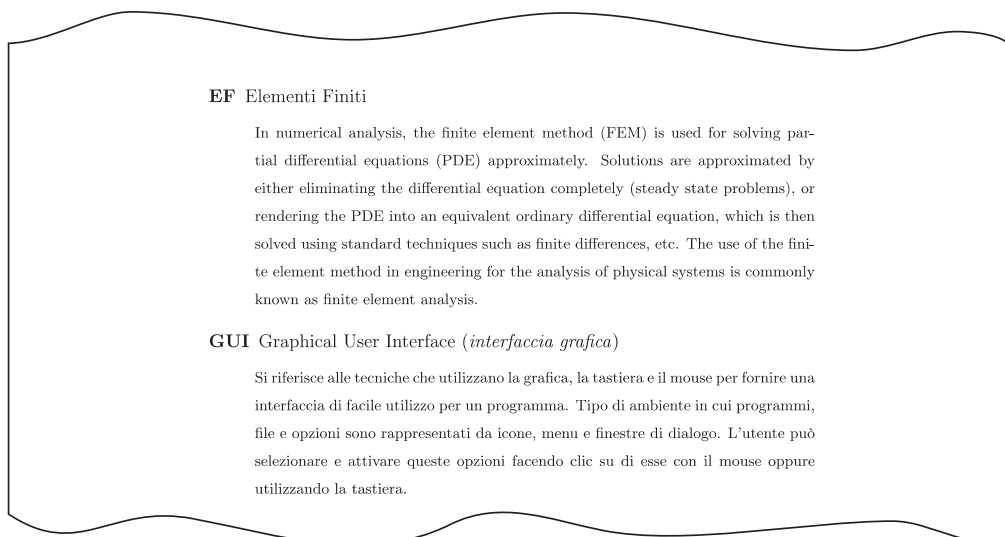


FIGURA 10: Esempio di elenco degli acronimi.

```
grazie all'eq.~\eqref{e2}
```

produce qualcosa del tipo

```
grazie all'eq. (3.6)
```

6.8 Revisione del codice

In fase di revisione del codice è molto utile, oltre ad un'attenta lettura del file di `.log`, l'utilizzo dei pacchetti `refcheck` e `showkeys` che controllano l'utilizzo dei `\label` e dei `\ref`. In aggiunta a questi è anche conveniente abilitare l'opzione `draft` per la `documentclass`: in questo modo i punti in cui il testo sborda dai margini verranno evidenziati con delle barre nere.

7 Siti utili

In aggiunta alle guide ed ai manuali citati nella bibliografia (vedi pag. 43 e seguenti), sono disponibili sul Web una serie di risorse utili per risolvere i problemi incontrati durante l'utilizzo di $\text{\LaTeX} 2_{\epsilon}$.

Riferimento primario per la comunità italiana di utenti $\text{\LaTeX}$ è il sito del Gruppo Utilizzatori Italiani di $\text{\TeX}$ (`GUI`)⁴¹ che ospita anche un forum sull'argomento.⁴²

Google, accanto al motore di ricerca tradizionale,⁴³ ne mette a disposizione anche uno dedicato al $\text{\LaTeX}$ ⁴⁴. Certamente utili sono il newsgroup Usenet inglese `comp.text.tex`⁴⁵ e quello italiano `it.comp.software.tex`.⁴⁶

Altro link di interesse è costituito da the Comprehensive TeX Archive Network (CTAN)⁴⁷ che

41. <http://www.guit.sssup.it/>

42. <http://www.guit.sssup.it/forum/>

43. <http://www.google.com/>

44. <http://directory.google.com/Top/Computers/Software/Typesetting/TeX/LaTeX/>

45. <http://groups.google.com/group/comp.text.tex>

46. <http://groups.google.it/group/it.comp.software.tex>

47. <http://www.ctan.org/>

ospita gran parte del materiale su $\text{\LaTeX}$ disponibile in rete ed è dotato di un motore di ricerca. Sarovar⁴⁸ è un catalogo molto completo di pacchetti e programmi legati a $\text{\TeX}$ e $\text{\LaTeX}$. Permette svariati tipi di ricerca, in particolare è estremamente utile la lista “topical” quando non si conosce il nome di un pacchetto ma solo “quello che deve fare”.

Molte università ed istituti di ricerca ospitano delle guide all'uso di $\text{\LaTeX}$ che generalmente sono pensate per neofiti. Si ricordano quella dell'Università di Cambridge,⁴⁹ quella dell'Università di Nottingham (principalmente sul setup del sistema)⁵⁰ quella dell'Università di Helsinki,⁵¹ quella della Emory University,⁵² ed infine quelle dell'Istituto Nazionale di Fisica Nucleare (INFN).⁵³ Si segnala inoltre che David R. Wilkins ha reso disponibile in formato `html` il suo libro *Getting Started with $\text{\LaTeX}$* .⁵⁴

Riferimenti bibliografici

5966, U. I. (1989). *Documentazione – Presentazione dei rapporti scientifici e tecnici*.

AMERICAN MATHEMATICAL SOCIETY (1995). *$\text{\LaTeX}$ User's Guide*. www.dd.chalmers.se/latex/Docs/PDF/amsl.doc.pdf.

48. <http://texcatalogue.sarovar.org/>

49. <http://www-h.eng.cam.ac.uk/help/tpl/textprocessing/>

50. <http://www.cs.nott.ac.uk/TSG/manuals/latex/intro/>

51. http://www.physics.helsinki.fi/~tfo_www/instr/latex-guide.html

52. http://www.emerson.emory.edu/services/latex/latex2e/latex2e_toc.html

53. <http://www.fi.infn.it/pub/tex/doc/html/latex.html> e <http://www.pi.infn.it/latex-intro/LaTeX.html>

54. <http://www.maths.tcd.ie/~dwilkins/LaTeXPrimer/>

- BAUDOIN, M. (1998). *Impara LATEX (...e mettilo da parte)*. http://www.mat.uniroma1.it/centro-calcolo/manuali/impara_latex.pdf.
- BECCARI, C. (1991). *LATEX, Guida a un sistema di editoria elettronica*. Hoepli.
- CARUSILLO, L. (2005). *Le tabelle in LATEX*. <http://www.webalice.it/lgrsll/pgl/latex/Tabelle.pdf.tar.gz>.
- CAUCCI, L. e SPADACCINI, M. (2005). *Gestione di Figure e Tabelle con LATEX*. http://www.guit.sssup.it/downloads/fig_tut.pdf.
- CEVOLANI, G. (2006). «Norme tipografiche per l'italiano in LATEX». *ArsTEXnica*, (1), pp. 29–42.
- DALY, P. (1998). *Graphics and Colour with LATEX*. <http://tex.loria.fr/graph-pack/grf/grf.pdf>.
- DILLER, A. (1999). *LATEX Line by Line: Tips and Techniques for Document Processing*. John Wiley & Sons.
- ECO, U. (1977). *Come si fa una tesi di laurea*. Bompiani, Milano.
- FAIRBAIRNS, R. (2005). *The UK TEX FAQ*. <http://www.ctan.org/tex-archive/help/uk-tex-faq/letterfaq.tex>.
- FEAR, S. (2005). *Publication quality tables in LATEX*. <http://www.ctan.org/tex-archive/macros/latex/contrib/booktabs/booktabs.pdf>.
- FLYNN, P. (2005). *A beginner's introduction to typesetting with LATEX*. <http://www.tug.org/tex-archive/info/beginlatex/>.
- GINI, R. (2004). *Generazione automatica di tabelle con LATEX e Stata*. http://www.guit.sssup.it/downloads/dispensa_gini.pdf.
- GOOSSENS, M., MITTELBACH, F. e SAMARIN, A. (1995). *The LATEX Companion*. Addison-Wesley.
- GOOSSENS, M., RAHTZ, S. e MITTELBACH, F. (1997). *The LATEX Graphics Companion: Illustrating Documents with TEX and PostScript*. Addison-Wesley.
- GRÄTZER, G. (1999). *First Steps in LATEX*. Springer Verlag.
- (2000). *Math into LATEX*. Birkhauser.
- GREGORIO, E. (2006). *LATEX: Breve guida ai pacchetti di uso più comune*. profs.sci.univr.it/~gregorio/breveguida.pdf.
- HAHN, J. (1993). *LATEX for Everyone: A Reference Guide and Tutorial for Typesetting Documents Using a Computer*. Prentice Hall.
- HIGHAM, D. e GRIFFITHS, D. (1997). *Learning LATEX*. Society for Industrial and Applied Mathematics.
- KNUTH, D. (1992). *The TEXbook*. Addison-Wesley.
- KOPKA, H. e DALY, P. (1995). *A Guide to LATEX – Document Preparation for Beginners and Advanced Users*. Addison-Wesley.
- LAMPORT, L. (1994). *LATEX: A Document Preparation System, User's Guide and Reference Manual*. Addison-Wesley.
- LEHMAN, P. (2004). *The Font Installation Guide*. <http://www.ctan.org/tex-archive/info/Typefonts/fontinstallationguide/fontinstallationguide.pdf>.
- LESINA, R. (1994). *Il nuovo manuale di stile. Guida alla redazione di documenti, relazioni, articoli, manuali, tesi di laurea*. Zanichelli, Bologna.
- MATRICCIANI, E. (2000). *La tesi scientifica*. Paravia, Torino.
- (2003). *Fondamenti di comunicazione tecnico-scientifica*. Apogeo, Milano.
- MORI, L. F. (2006). «Tabelle su LATEX 2_ε: pacchetti e metodi da utilizzare». *ArsTEXnica*, (2), pp. 31–47.
- OETIKER, T., PARTL, H., HYNÄ, I. e SCHLEGEL, E. (2000). *Una (mica tanto) breve introduzione a LATEX 2_ε*. <http://www.ctan.org/tex-archive/info/lshort/italian/itlshort.pdf>.
- PAKIN, S. (2003). *The Comprehensive LATEX Symbol List*. <http://www.ctan.org/tex-archive/info/symbols/comprehensive/symbols-a4.pdf>.
- PRATESI, M., LATINI, M. e ANTONECCHIA, M. (2003). *LATEX-PDF-Howto*. <http://www.telug.it/marco/LaTeX-PDF-HOWTO/LaTeX-PDF-HOWTO-1.1.1.pdf>.
- RECKDAHL, K. (1997). *Using Imported Graphics in LATEX 2_ε*. <http://www.ctan.org/tex-archive/info/epslatex.pdf>.
- TALBOT, N. (2004). *Creating a PDF document using PDFLATEX*. <http://theoval.cmp.uea.ac.uk/~nlct/latex/pdfdoc/pdfdoc-a4.pdf>.
- THE TUTORIAL TEAM (2000). *On-line Tutorial on LATEX*. Indian TEX Users Group. <http://www.tug.org.in/tutorials.html>.

- TRETTIN, M. (2005). *Elenco dei “peccati” degli utenti di $\text{\LaTeX}$ 2_ε*. <http://www.tug.org/tex-archive/info/l2tabu/italian/l2tabuit.pdf>.
- VOSS, H. (2003a). *Mathmode*. www.ctan.org/tex-archive/info/math/voss/mathmode/Mathmode.pdf.
- (2003b). *Rotating Text, Tabulars and Images*. <http://perce.de/LaTeX/rotating.pdf>.

▷ Lapo Filippo Mori
Department of Mechanical Engineering
Northwestern University
Evanston IL 60208
USA
mori@northwestern.edu

La tipografia nel taschino

Presentazione del sistema $\text{\texttt{OPs4GJr}}$

Salvatore Schirone

Sommario

$\text{\texttt{OPs4GJr}}$ è un sistema $\text{\texttt{L\TeX}}$ portatile per *PenDrive* USB utilizzabile sul sistema operativo Windows (9x, ME, XP) e ottenibile gratuitamente e in formato aperto dalla rete. $\text{\texttt{OPs4GJr}}$ permette di avere sempre a portata di mano un sistema $\text{\texttt{L\TeX}}$ completo e funzionante, e quindi di ricompilare i nostri sorgenti su qualsiasi macchina, anche se sprovvista di un'installazione $\text{\texttt{L\TeX}}$. In questo articolo presento per la prima volta $\text{\texttt{OPs4GJr}}$, descrivendone la struttura, l'installazione e la personalizzazione.

Abstract

$\text{\texttt{OPs4GJr}}$ is an *opensource* portable USB $\text{\texttt{L\TeX}}$ system for Windows (9x, ME, XP), freely available from internet. $\text{\texttt{OPs4GJr}}$ provides a fully working $\text{\texttt{L\TeX}}$ system always at hand, so that one can compile its own source `.tex` files on every computer. In the present article I will introduce $\text{\texttt{OPs4GJr}}$ for the first time, and I will describe its structure and how to install and customize it.

1 Presentazione di $\text{\texttt{OPs4GJr}}$

Chi usa il sistema $\text{\texttt{L\TeX}}$ per scrivere i propri documenti conosce bene alcuni problemi che si presentano nell'uso intensivo di questo programma:

- l'installazione di un sistema $\text{\texttt{L\TeX}}$ completo è relativamente complessa e occupa qualche centinaio di megabyte sul disco del calcolatore;
- anche per questo motivo, macchine diverse hanno spesso installazioni diverse, o ne sono completamente sprovviste, di conseguenza, risulta a volte impossibile ricompilare e gestire i propri documenti su architetture diverse dalla propria;
- il sistema $\text{\texttt{T\EXLive}}$, un sistema $\text{\texttt{L\TeX}}$ completo multiplatforma utilizzabile da cdrom è scomodo perché il supporto non riscrivibile ci costringe all'uso di un altro supporto (floppy, *PenDrive* USB ecc.) per i sorgenti e gli eventuali pacchetti aggiuntivi utilizzati, inoltre tale sistema non è aggiornabile se non a condizione di rimasterizzare una nuova versione (su un nuovo CD vergine).

Manca quindi, allo stato attuale, una versione di $\text{\texttt{L\TeX}}$ realmente e comodamente portatile, che permetta di avere a portata di mano sia i propri

documenti sia un sistema $\text{\texttt{L\TeX}}$ funzionante anche su macchine differenti. $\text{\texttt{OPs4GJr}}$ nasce esattamente per rispondere a questa esigenza.

La soluzione più comoda e semplice è quella di poter avere tutto ciò che ci occorre racchiuso all'interno di una pen-drive USB: il sistema $\text{\texttt{L\TeX}}$ già configurato, diversi suoi strumenti (tra cui visualizzatori, editor, ecc...) e i nostri documenti da compilare, con la possibilità di aggiornare la distribuzione, proprio grazie al fatto che il supporto USB è scrivibile.¹

L'installazione di un sistema $\text{\texttt{L\TeX}}$ su *PenDrive* USB potrebbe sembrare semplice e scontata da realizzare, tuttavia richiede un'analisi molto attenta in merito al software da installare sul supporto USB, così da confezionare una versione completamente portatile dell'intero sistema $\text{\texttt{L\TeX}}$. Il suo sviluppo è stato difficile, soprattutto per via della struttura tanto complessa del software in questione, poiché risulta strettamente legato alla macchina su cui deve funzionare.

Il caso che, a mio avviso, si verifica più di frequente è quello di trovarsi occasionalmente a lavorare su di una macchina con sistema operativo Windows, dunque naturalmente priva di una qualunque distribuzione $\text{\texttt{L\TeX}}$. A questo proposito mi sono dedicato alla realizzazione di una distribuzione portatile funzionante (per ora) solo su tale sistema operativo.

L'esperienza che ho maturato nell'ambito della preparazione di sistemi per programmi portatili in questi ultimi due anni, mi ha portato alla realizzazione di *oPenSuite*, ovvero di una versione specifica della mia *suite* per $\text{\texttt{L\TeX}}$.

L'occasione mi si è presentata dopo aver scoperto l'esistenza di un *how-to* scritto da Alexander Grahn (GRAHN, 2006), poi tradotto per $\text{\texttt{GJr}}$ da Stefano Caglio. Grahn spiegava come "portabilizzare" $\text{\texttt{L\TeX}}$, rendendo disponibile un set di *script batch* per la configurazione della distribuzione $\text{\texttt{MiKTeX}}$. Mi sono subito messo all'opera e il risultato è $\text{\texttt{OPs4GJr}}$ ²:

1. Abbiamo bisogno di una *PenDrive* di almeno 512 MB, cosa che oggi non è certo un problema, vista l'ampia diffusione e il costo contenuto di *PenDrive* con capacità pari o superiori ad 1GB.

2. La *suite* è prelevabile dal mio sito personale: <http://www.schirone.it> e il logo della *suite* è stato realizzato da Massimo Caschili.



2 La storia di oPs4GjT

È opportuno fare un breve *excursus* che inquadrì i termini della questione, chiarendo innanzitutto cosa si intende per “programma portatile”, e cosa sono le *PenSuite* e nella fattispecie, *oPenSuite* (chi è interessato solamente all'utilizzo di oPs4GjT, può passare direttamente alla sezione successiva).

2.1 Come creare programmi portatili

I programmi per Windows sono quasi sempre distribuiti in forma di pacchetti *autoinstallanti* (ovvero distribuiti sotto forma di un unico file, spesso chiamato *Setup.exe*). Questo significa che durante la sua installazione, e spesso in modo poco trasparente per l'utente, il pacchetto provvede a copiare una serie di file in cartelle diverse del nostro *filesystem* e a scrivere o modificare chiavi nel file registro di configurazione di Windows. Un programma portatile, invece, di solito viene distribuito o in forma d'archivio tipo *zip*, oppure in forma d'archivio *autoestraente* (con estensione *.exe*). L'estrazione dei files dall'archivio fa sì che tutti i files necessari al programma (eseguibili, librerie *dll*, file di configurazione, risorse varie, ecc...) vengano salvati all'interno di una cartella. L'esecuzione del programma avviene lanciando l'eseguibile principale, che di solito porta il nome del programma, *<programma>.exe*. Un programma con queste caratteristiche è portatile e può essere avviato direttamente da una *PenDrive* USB.

I programmi *opensource*, di solito, vengono rilasciati contemporaneamente sia sotto forma d'archivio, che sotto forma di pacchetto autoinstallante. Tuttavia, quando disponiamo solo del pacchetto d'installazione di un programma, renderlo portatile non è un'operazione semplice; poi, se non si dispone dei *file sorgenti*,³ tale operazione si può rivelare addirittura impossibile. Utilizzando particolari software, in grado di estrarre i file del programma dal file *.exe* del pacchetto d'installazione, si possono fare dei tentativi. Un software di questo genere, attualmente tra i più efficaci, è *Universal Extractor*. Il programma permette di saltare l'operazione d'installazione sul PC del pacchetto software che vogliamo rendere portatile, e ne estrae tutti i file come se l'autoinstallante fosse un normale archivio compresso.

Il procedimento illustrato finora non basta a rendere portatile un pacchetto software, poiché bisogna capire se il programma, in fase d'installazione, scrive o modifica una qualche chiave nel registro di configurazione di Windows. Per scoprirlo si possono usare alcuni programmi di monitoraggio, tra

3. Sono i file racchiusi all'interno del pacchetto autoinstallante.

i quali spicca *RegMon*.⁴ Contemporaneamente, ovvero sempre durante la sua fase d'installazione, è necessario scoprire se il programma salva dei dati in cartelle particolari del *filesystem*, come ad esempio in:

```
C:\Documents and Settings\
[user]\Dati applicazioni
```

e a tal fine ci vengono in aiuto altri programmi di monitoraggio, come ad esempio *FileMon*.⁵

Esistono alcuni piccoli eseguibili detti *launcher* che sono in grado di salvare nella stessa cartella del programma le rispettive chiavi di registro e i dati dell'utente. Uno tra questi è *Reg Rapper*.⁶

Attualmente, il progetto più completo e avanzato di *launcher* è in grado di far funzionare un programma e di salvare nella relativa cartella i dati e le sue chiavi di registro, così da rendere il programma stesso portatile al 100%. Si tratta di *X-Launcher*, presente su *WinPenPack.com*. Questo *launcher* è ormai in grado di “portabilizzare” una gran quantità di programmi; il suo utilizzo è reso difficile dalla sintassi particolare che occorre conoscere per impostarlo adeguatamente, dunque, prima di ogni altra cosa, bisogna analizzare bene il programma che si desidera rendere portatile.⁷

2.2 Nasce oPenSuite

Il fatto di poter disporre di svariati programmi portatili non è sufficiente da sé a realizzare una buona *distribuzione* portatile, poiché occorre anche un qualche *sistema di supporto* che consenta di gestire in modo pratico e comodo tali programmi: lanciarli direttamente dalla *PenDrive*, configurarli in base alle nostre esigenze, aggiornarli all'ultima versione disponibile, ecc... Queste operazioni vengono gestite da un cosiddetto *software launcher*. La gestione di tale programma e la composizione di una collezione di programmi portatili, che può essere ad esempio *base* o *plus*, è quello che noi chiamiamo *PenSuite*. Attualmente esistono diversi progetti attivi (italiani e non) che offrono gratuitamente una *suite* di programmi portatili per *PenDrive*.⁸ A mio avviso, il miglior progetto italiano e supportato da un'ampia comunità di utenti e sviluppatori, è il già citato *WinPenPack.com*.⁹

4. <http://www.microsoft.com/technet/sysinternals/utilities/regmon.mspx>

5. <http://www.microsoft.com/technet/sysinternals/FileAndDisk/Filemon.mspx>

6. <http://www.portasoft.org/e107/page.php?4>.

7. Per questo paragrafo mi sono servito ampiamente delle ricerche fatte da Andrea Luparia: <http://www.lupo73.altervista.org>.

8. Tra i più rilevanti segnaliamo: *WinInizio PenSuite* (<http://pensuite.wininizio.it/ita/index.php>), *MozUp* (<http://www.mozillaitalia.it/mozup/>), *PortableApps.com* (<http://portableapps.com/>); per una recensione cfr. (<http://www.lupo73.altervista.org/pro.htm>).

9. <http://www.winpenpack.com>.

La quasi totalità delle *suite* esistenti si basano su *launcher* proprietari, gratuiti, ma non *opensource*, quindi difficilmente personalizzabili e ricompilabili dagli utenti. Solo recentemente l'autore del *launcher* di *WinPenPack* ne ha reso disponibile il sorgente, ma la sua ricompilazione è risultata alquanto problematica e la documentazione molto scarsa.¹⁰ Nella ricerca di un progetto totalmente *opensource* mi sono imbattuto in *FramaKey*,¹¹ un progetto francese che sta portando avanti l'idea di una *suite* composta rigorosamente da programmi *opensource* ed è proprio a questo eccellente progetto che mi sono ispirato nella realizzazione della mia *suite*. In un primo momento ho contribuito direttamente alla sua localizzazione italiana, poi mi sono distaccato con un mio *fork*, cioè una versione modificata, perché, malgrado girasse intorno una attiva comunità, lo sviluppo dei programmi era ed è molto indietro rispetto ad altri progetti anche italiani; inoltre la comunità è risultata prevalentemente francese e non ho trovato il conforto di altri collaboratori italiani. In questo ultimo anno, *oPenSuite* si è evoluto modificando molti aspetti della *FramaKey*, divenendo di fatto un progetto completamente indipendente. La versione attuale, ancora in fase beta, è la 0.6.

Frequentando diversi forum sui siti di supporto alle varie *suite*, ho conosciuto Andrea Luparia, che ha sviluppato una sua *suite* in concomitanza con il mio progetto. Attualmente mi avvalgo della sua collaborazione nella *portabilizzazione* dei vari programmi, e si sta valutando insieme l'ipotesi di giungere ad una fusione dei nostri due progetti.¹²

Il nome "*oPenSuite*" contiene in se stesso la definizione e la finalità del progetto: una *suite* di programmi portatili per supporti rimovibili, *PenDrive*, completamente *opensource*¹³ e particolarmente personalizzabile ed estendibile dall'utente. L'obiettivo di *oPenSuite* è di dare agli utenti la massima libertà di personalizzazione e adattamento. Per questo la *suite* viene rilasciata solo nella versione base con tre programmi essenziali: il *browser* Firefox, l'*editor* Notepad++ e l'utilità di archiviazione 7zip. Gli eventuali programmi aggiuntivi possono essere prelevati da altre fonti, dal momento che il mio compito principale è quello di aggiornare la *suite*

e di sviluppare/localizzare particolari programmi tra cui, appunto, MiK_TE_X.

oPenSuite, nei suoi contenuti testuali e multimediali, viene rilasciata sotto licenza *Creative Commons*, "*Attribuzione – Condividi allo stesso modo 2.5 Italia*", mentre i programmi da cui dipende mantengono le loro licenze originali e sono elencati qui di seguito:

AeTrayMenu¹⁴ Una piccola ma potente applicazione che permette di inserire nella *traybar* di Windows un menù flessibile e configurabile, dal quale è possibile lanciare le applicazioni portatili. Viene rilasciata sotto licenza *Creative Commons Public Domain*;

SplashStart¹⁵ Programma di avvio della *suite* con un semplice *splash screen*. Utilizza un file *.ini* per la configurazione ed è distribuito sotto licenza *GPL*;

TiddlyWiki¹⁶ Un *Wiki* contenuto in una unica pagina. Distribuito sotto licenza *BSD*;

NSIS¹⁷ Programma utilizzato per la compilazione dell'installer della *suite* e distribuito sotto la *Common Public License*.

È proprio grazie alla disponibilità dei sorgenti e all'ampia libertà garantita da queste licenze che è stato possibile realizzare *oP_s4G_{IT}*.

2.3 Il MiK_TE_X portatile

La versione portatile di MiK_TE_X, ops-MiK_TE_X è basata sugli *script batch*¹⁸ messi a disposizione da Graham. Tali *script*¹⁹ permettono sostanzialmente di:

- inizializzare MiK_TE_X, Ghostscript e GSview sul dispositivo USB (*init.bat*);
- configurare ed aggiornare *MiK_TE_X* (*mo.bat* e *mpm.bat*);
- ripristinare tutte le configurazioni del PC ospitante (*reset.bat*).

Per rendere portatile MiK_TE_X è stato necessario installare su di un PC la sua distribuzione più aggiornata (prelevabile dal sito ufficiale del progetto²⁰) in versione "leggera", accompagnato dai programmi accessori Ghostscript e GSview. Il procedimento completo è illustrato qui di seguito.

Una volta creata la cartella ops-miktex, si creano al suo interno due nuove sotto-cartelle: *texmf*

10. Si tratta di un ottimo *launcher*, *ASuite* (<http://www.salvadorsoftware.com/>)

11. <http://www.framakey.org>.

12. L'accordo è ancora in fase di discussione poiché vi sono alcune diversità d'opinione in merito ai software da distribuire: per me la *suite* dovrebbe essere completamente *opensource*, mentre Andrea Luparia, pur adottando un *launcher opensource*, preferisce inserire anche programmi con licenze proprietarie *freeware*.

13. I programmi *OpenSource*, comunemente chiamati anche *FreeSoftware*, con una accentuazione più filosofica che tecnica, non sono da confondersi con i programmi *freeware*. Questi ultimi, benché siano rilasciati gratuitamente per l'utilizzo privato, non possono essere modificati (non si dispone del codice sorgente), e possono essere ridistribuiti solo a particolari e restrittive condizioni poste dall'autore.

14. <http://www.xs4all.nl/~broekroo/aetraymenu>.

15. <http://cdbrowserkit.sourceforge.net/SplashStart/SplashStart.html>

16. <http://www.tiddlywiki.com>, nella traduzione di Maurizio Pollio, <http://pollio.maurizio.googlepages.com/MiniGuidaTiddlyWiki.html>.

17. <http://nsis.sourceforge.net>.

18. Uno *script batch* è un vero e proprio programma costituito da linee di comando passate all'interprete shell di DOS (= *script*), in una determinata sequenza (= *batch*).

19. Per approfondimenti visitare la pagina: <http://www.guit.sssup.it/installazione/drive.php>.

20. <http://www.miktex.org>.

e `localtexmf.texmf` è la cartella dove andremo a copiare tutto ciò che è contenuto nella cartella principale della nostra distribuzione MiKTeX, che potrebbe essere, ad esempio

`C:\Program Files\MiKTeX 2.5`

La cartella `localtexmf` servirà a contenere ulteriori pacchetti aggiuntivi, creati da terze parti, come ad esempio `arstexnica` di `qJr`, che ho utilizzato per comporre questo articolo. Tutti i file batch devono essere nuovamente copiati nella *root* della cartella `ops-miktex`, assieme all'utility propria di Windows `reg.exe` (versione 3.0), reperibile nella cartella di sistema `c:\WINDOWS\system32`. A ciò occorre aggiungere le cartelle relative alle installazioni di Ghostscript e GSview: `gs` e `Ghostgum`. Una impostazione fondamentale riguarda lo script `versions.bat`, che deve contenere le esatte versioni dei programmi MiKTeX, Ghostscript e GSview, nelle rispettive variabili: `MiKTeX_VERSION`, `GS_VERSION` e `GSVIEW_VERSION`.

Per agevolare l'esecuzione dei tre script fondamentali, è stata approntata una interfaccia grafica, con `wxbasic`²¹, un compilatore basic che sfrutta le librerie `wx`. L'idea è stata presa da un articolo di Jerónimo Leal (LEAL, 2006). Sono bastate poche linee di codice per ottenere un discreto pannello di controllo (cfr. Figura 2). Il sorgente e il compilatore sono inseriti nella cartella `src` della *suite*, a disposizione di chi lo volesse modificare e ricompilare, così da espanderne le funzioni o semplicemente migliorare l'aspetto grafico.

Infine, per rendere il tutto più funzionale e pratico, è stato inserito nella *suite* l'editor *Texmaker*, anch'esso avviabile dal **Pannello di Controllo**. Grazie alla sua semplicità e leggerezza non ha richiesto nessun intervento di *portabilizzazione*. L'ultima fase del lavoro ha riguardato la compressione e l'archiviazione della cartella del progetto, `ops-miktex`.

3 Usare `oPs4qJr`

3.1 Installazione e uso

È molto semplice installare e utilizzare `oPs4qJr` e, come prima cosa, occorre prelevare il pacchetto `ops4guiltInstaller_base-0.1.0.0-it.exe` dal sito dell'autore. L'operazione potrebbe richiedere alcuni minuti, a seconda del tipo di connessione che si sta utilizzando, poiché si tratta di un archivio di ben 84.69MB. Il "doppio click" sull'eseguibile avvia la procedura d'installazione e si viene subito informati del fatto che il pacchetto è distribuito nei termini della licenza *Creative Commons* e che per poter proseguire, occorre accettare tali termini. Successivamente viene chiesto di indicare il supporto USB sul quale si desidera installare la

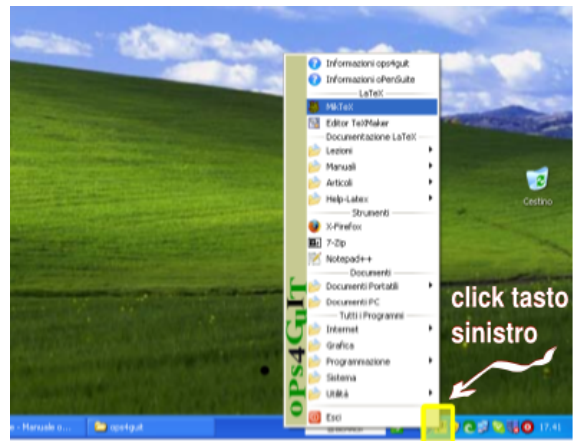


FIGURA 1: Menu tasto sinistro

suite (è possibile installarla anche in una cartella del nostro disco locale). Lo script dell'installer *NSIS* completerà la procedura d'installazione e, giunto al termine, ci chiederà se vorremo avviare l'applicazione.

Successivamente, grazie ad un sistema di *auto-run*, ogni qualvolta inseriremo la pen-drive nella porta usb del computer ci apparirà uno *splash screen* che ci pregherà di attendere qualche secondo. Trascorso questo tempo, nella *traybar* del *Desktop* di Windows troveremo l'icona `oPs4qJr`, dalla quale potremo lanciare le varie applicazioni ed applicare alla *suite* le nostre personalizzazioni.

Con il tasto sinistro del mouse, potremo accedere al menu contestuale principale (cfr. Figura 1), che consente di avviare le varie applicazioni, esplorare la cartella dei documenti portatili o accedere anche ai documenti presenti sul computer ospitante.

Le due voci del menù **Informazioni** consentono di consultare rispettivamente il manuale generale di *oPenSuite* e quello specifico di `oPs4qJr`, all'interno del browser predefinito (*X-Firefox*). Entrambi i manuali possono essere contestualmente modificati e aggiornati grazie a *Tiddlywiki*. Sele-

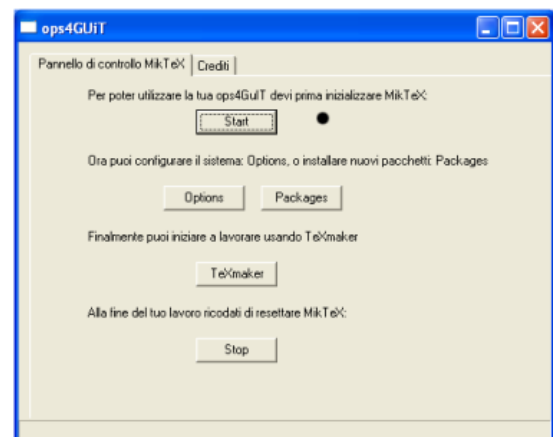


FIGURA 2: Pannello di controllo `oPs4qJr`

21. <http://wxbasic.sourceforge.net/>

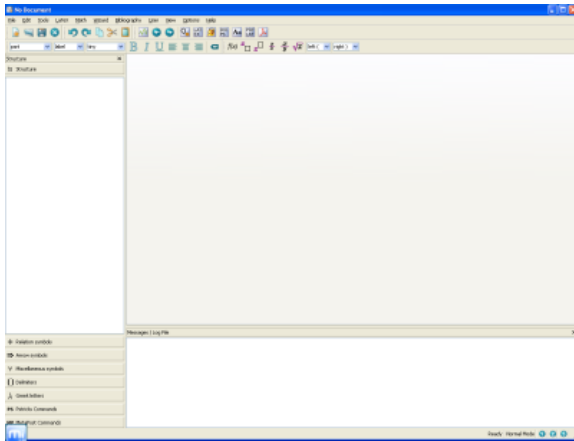


FIGURA 3: Texmaker

zionando **Esci**, termina l'esecuzione di **oPs4GUIT**. In aggiunta, dallo stesso menù potremo disporre di tutti gli strumenti messi a disposizione dalla *suite* e avviare tutti gli eventuali programmi che avremo installato durante la nostra fase di personalizzazione. In particolare, potremo consultare rapidamente alcuni tutorial e manuali distribuiti dal **GUIT**.

La sezione **LaTeX** conduce alla configurazione vera e propria della nostra distribuzione **TEX** portatile e, selezionando la voce **MiKTEX**, ne appare il pannello di configurazione (cfr. Figura n. 2). Il pulsante **Start** avvia, all'interno di una finestra DOS, lo script di configurazione generale di MiKTEX. Al termine dell'operazione, se il processo avrà avuto successo, avremo a disposizione il nostro sistema MiKTEX pronto per l'uso. Attraverso il pulsante **Opzioni** potremo effettuare diverse operazioni di configurazione, oppure installare altre estensioni per MiKTEX (*packages*). Il pulsante **Texmaker** avvierà l'omonimo editor (cfr. Figura n. 3), che potrà essere configurato ulteriormente attraverso le sue impostazioni di configurazione. *Texmaker* può anche essere avviato dal menù principale della *suite*. Quando avremo terminato il lavoro, dovremo ricordarci di arrestare il sistema tipografico premendo il pulsante **Stop**. Ciò avvierà lo script `reset.bat` che riporterà tutte le impostazioni del registro di sistema allo stato iniziale, così da non lasciare alcuna traccia fastidiosa dei programmi appena utilizzati sul computer ospitante e senza compromettere le impostazioni di una eventuale distribuzione MiKTEX preesistente nel sistema.

3.2 Personalizzazione

un click Con il tasto destro del mouse (cfr. Figura n. 4) si accede ad un altro menu dal quale possiamo gestire tutti i parametri di configurazione della *suite*, accedere al prompt di DOS e ad altre informazioni supplementari.

Selezionando la voce **Edita menu**, si avvia l'editor *Notepad++* con il file

`Framakey/FramaLauncher/FramaLauncher.ini`

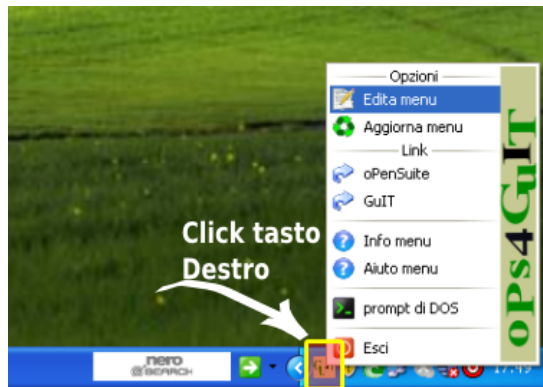


FIGURA 4: Menu tasto destro

precaricato al suo interno. Agendo su questo file si possono aggiungere e/o modificare le *voci* di menù (cfr. Figura n. 5). La sintassi da seguire è abbastanza intuitiva e le varie sezioni sono ampiamente commentate, dunque è sufficiente dare uno sguardo agli esempi già pronti per poi tentare di scrivere le proprie *voci*. Per fare un esempio, se si desiderasse utilizzare anche l'editor *Vim*, per scrivere i propri sorgenti **LATEX**, e lo si volesse far 'convivere' con *Texmaker*, dovremmo, prima di tutto, prelevare l'archivio dell'applicazione *GVim Portable* dall'indirizzo <http://portablegvim.sourceforge.net> e poi estrarne i file, ponendoli nella cartella *Apps* della nostra *suite*. Successivamente ci accerteremo dell'esatto percorso in cui si venisse a trovare l'eseguibile principale dell'applicazione,

`Apps/GVimPortable/GVimPortable.exe`

indi creeremo la relativa voce del menù come spiegato nei passi seguenti.

1. Avvieremo l'editor di configurazione e con il tasto destro del mouse selezioneremo la voce **Edita menu**.
2. Ci sposteremo alla sezione `[Menu.Left]` e, in coda alla parte di codice il cui commento si riferisce alla sottosezione **LaTeX**, aggiungeremo il seguente blocco di codice:

```
Type: item; Caption: "GVimPortable";
```

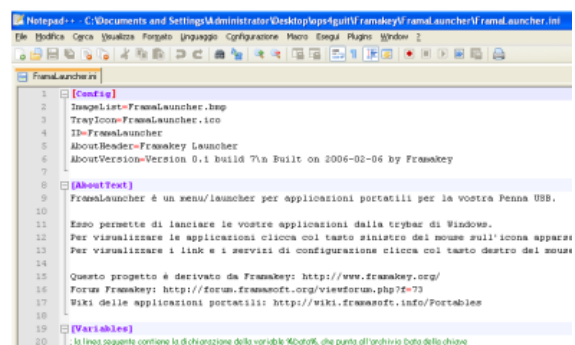


FIGURA 5: Configurazione del menu

```
Action: run;
FileName: "%Apps%\GVimPortable\
GVimPortable.exe";
Glyph: 26
```

La sintassi è particolarmente intuitiva, infatti si può notare esser composta da una serie di coppie del tipo `<elemento>: <valore>`²²:

Type:item annuncia che ci si appresta ad aggiungere o modificare una voce di menù;

Caption:<nome> specifica il nome della voce che apparirà scritto nel menù;

Action:run si riferisce all'azione da eseguire, in questo caso si tratta di avviare (*run*) una applicazione;

FileName:<eseguibile> specifica il percorso e il nome dell'eseguibile associato all'applicazione che si desidera avviare;

Glyph:<n> è il numero identificativo dell'icona associata al programma.

Per inserire l'icona di *Vim* nel suddetto menù, occorre agire sul file

```
Framakey\Fra...er\FramaLauncher.bmp
```

Bisogna aprirlo con un editor grafico e copiarvi al suo interno l'icona prescelta in formato **bmp** (e su sfondo rosa) entro uno dei riquadri liberi di coda. La posizione della nuova icona sarà associata ad un numero che identificherà proprio il valore di **Glyph**.

3. Salveremo il file e chiuderemo l'editor.
4. Per rendere efficaci le modifiche appena apportate, aggiorneremo il sistema con un semplice click destro del mouse su **Aggiorna menu**.
5. Infine, nel caso in cui avessimo commesso un errore di sintassi, riceveremmo un avviso immediato e dovremmo riaprire l'editor per correggerlo.

A questo punto, nel menù della *suite*, dovremmo trovare la nuova voce per poter avviare *Vim*. Se volessimo personalizzare questo editor, così da poterlo utilizzare al meglio con L^AT_EX, potremmo prelevare il *plugin* adatto da <http://vim-latex.sourceforge.net/index.php> e seguire le istruzioni per la sua installazione. In particolare, durante la procedura d'installazione del *plugin*, ogni qualvolta venga fatto riferimento alla cartella `~/vimfiles`, per la nostra *suite* dovremo riferirci alla cartella

```
Apps/GVimPortable/App/vim/vim70
```

Il menù a scomparsa, visualizzabile col tasto destro del mouse, contiene un elenco di collegamenti utili a siti internet, che possono essere ampliati o modificati a piacimento dall'utente. La voce **Info**

22. Il punto e virgola, che segue ciascuna voce, è necessario.

menu mostra una finestra con informazioni relative ad *AeTrayMenu*, ovvero il *launcher* della *tray-bar*, mentre **Aiuto menu** ne fornisce l'help in linea. **Prompt di Dos** esegue il comando `cmd.exe`, che mette a disposizione dell'utente la console di MS-DOS. **Esci** ha la stessa funzione del suo omonimo comando, attivabile col tasto sinistro del mouse.

4 Sviluppo futuro e ringraziamenti

Lo sviluppo del *core* del sistema dipenderà essenzialmente dall'evoluzione che avrà – e se l'avrà, *oPenSuite*. La *todolist*²³, comunque, è la seguente:

- implementare altri strumenti complementari a L^AT_EX;
- inserire anche *Emacs* come *editor* alternativo a *Termaker*;
- aggiungere altri manuali del G_LT_R;
- sviluppare il manuale wiki della *suite*;
- migliorare la grafica e il pannello di controllo.

Tutti gli utenti sono invitati a partecipare al progetto così da sostenerne lo sviluppo futuro. Ciò può essere fatto in vari modi, ad esempio scrivendo all'autore per segnalare eventuali errori, chiedere l'aggiunta di nuove funzioni, oppure aggiornando il manuale L^AT_EX attraverso *TiddlyWiki*. Più direttamente, è possibile partecipare al progetto anche tentando di personalizzare e sviluppare autonomamente la *suite* e mettendo a disposizione di tutti gli utenti il proprio lavoro, tramite il forum di discussione raggiungibile dal sito dell'autore.

Si ringraziano in particolare:

- Massimo Caschili per il logo ;
- Andrea Luparia per la fattiva collaborazione a *oPenSuite*;
- Stefano Caglio per la traduzione di Grahn;
- gli amici del G_LT_R per tutto il resto.

Riferimenti bibliografici

GRAHN, A. (2006). «Miktex+ghostscript+gsview+usb-drive-howto». <http://www.ctan.org/tex-archive/documentation/MiKTeX+Ghostscript+GSview+USB-drive-HOWTO.txt>.

LEAL, J. (2006). «Esperienze didattiche con L^AT_EX». *ArsTeXnica*, (2), pp. 71–74. URL <http://www.guit.sssup.it/arstexnica.php>.

▷ Salvatore Schirone
<http://www.schirone.it>
info@schirone.it

23. In gergo informatico, dall'inglese, è l'elenco dei compiti di cui ci si deve ancora occupare.

Questa rivista è stata stampata
su carta ecosostenibile Steinbeis Vision
prodotta dalla cartiera tedesca
Steinbeis Temming Papier GmbH & Co.



**Scuola Superiore
Sant'Anna**
di Studi Universitari e di Perfezionamento



Quarto convegno nazionale su T_EX, L^AT_EX e tipografia digitale

Pisa, 13 ottobre 2007

Aula Magna, Scuola Superiore Sant'Anna

Call for Paper

Sabato 13 ottobre 2007 presso l'Aula Magna della Scuola Superiore Sant'Anna di Pisa si terrà il quarto Convegno annuale su T_EX, L^AT_EX e tipografia digitale organizzato dal Gruppo Utilizzatori Italiani di T_EX. Il Convegno sarà un momento di ritrovo e di confronto per la comunità L^AT_EX italiana, tramite una serie di interventi atti sia a contribuire all'arricchimento sia a supportarne lo sviluppo. Maggiori informazioni sul Convegno e sulle modalità di presentazione degli interventi sono disponibili all'indirizzo:

<http://www.guit.sssup.it/guitmeeting/2007/>

ArsT_EXnica

Rivista italiana di T_EX e L^AT_EX

Numero 3, Aprile 2007

- 3 Editoriale
Massimiliano Dominici
- 4 Intervista a Frank Mittelbach
Frank Mittelbach, Gianluca Pignalberi, Dave Walden
- 13 L^AT_EX e la cesura delle parole in fin di riga
Claudio Beccari
- 21 L^AT_EXpedia: il futuro della documentazione su L^AT_EX
Lapo Filippo Mori
- 27 Scrivere la tesi di laurea con L^AT_EX 2_ε
Lapo F. Mori
- 46 La tipografia nel taschino
Salvatore Schirone