

---

## La classe TOPtesi

---





**POLITECNICO  
DI TORINO**

Manuale d'uso

## **La classe TOPtesi**

Per comporre le tesi al Politecnico di Torino  
e in molte altre università

CLAUDIO BECCARI

Classe toptesi versione v.5.92 del 2016/06/29

Questo testo è libero secondo le condizioni stabilite dalla L<sup>A</sup>T<sub>E</sub>X Project Public Licence (LPPL) riportata nell'appendice [A](#) alla pagina [91](#).

Composto con LuaL<sup>A</sup>T<sub>E</sub>X il 2 luglio 2016

# Sommario

Questo testo serve per descrivere come comporre tipograficamente la tesi di laurea o la monografia o la dissertazione di dottorato mediante il noto programma di composizione  $\text{\LaTeX}$ , o meglio, mediante le sue varianti  $\text{pdf\LaTeX}$ ,  $\text{Xe\LaTeX}$  o  $\text{Lua\LaTeX}$ ; per produrre con  $\text{Xe\LaTeX}$  il file finale in formato PDF archiviabile secondo la norma ISO 19005-1 bisogna procedere come descritto nel paragrafo [3.9.3](#).

# Summary

This text describes how to typeset a university master thesis, or the bachelor final report, or the PhD dissertation through the well known typesetting program  $\text{\LaTeX}$ , or rather through its variants  $\text{pdf\LaTeX}$ ,  $\text{Xe\LaTeX}$ , or  $\text{Lua\LaTeX}$ ; in order to produce the final document in a PDF archivable format according to the ISO regulation 19005-1 it's necessary to proceed as described in section [3.9.3](#).

# Ringraziamenti

Ringrazio gli studenti del Politecnico di Torino che mi hanno sollecitato a mettere la mia esperienza a loro disposizione per predisporre e rendere disponibile il software necessario per preparare le loro tesi, monografie o dissertazioni con la qualità che solo pdf $\LaTeX$ , X $\LaTeX$  o Lua $\LaTeX$  riescono a produrre<sup>1</sup>.

---

<sup>1</sup>Questa particolare composizione è stata eseguita con X $\LaTeX$ . Questo motore di composizione presenta numerosi vantaggi su pdf $\LaTeX$  per quel che riguarda l'uso dei font, anche per la matematica, ma non è ancora (2015) in grado di produrre l'uscita direttamente nel formato PDF, anche se apparentemente lo fa. In realtà la sua uscita è in un formato intermedio che viene poi trasformato automaticamente in PDF. L'esperienza mi insegna che in realtà le limitazioni di X $\LaTeX$  sono pochissime e, con la classe *toptesi*, quella forse più importante, ma facilmente risolvibile, riguarda il formato PDF archiviabile che non è ottenibile direttamente, tanto che bisogna procedere come indicato nel paragrafo [3.9.3](#).

# Indice

<b>Elenco delle tabelle</b>	<b>VIII</b>
<b>Elenco delle figure</b>	<b>IX</b>
<b>1 Guida rapida all'uso di questo manuale</b>	<b>1</b>
1.1 A cosa serve TOPtesi . . . . .	2
1.2 Cosa leggere . . . . .	3
1.3 Errori da evitare . . . . .	5
1.4 Pacchetti già caricati da TOPtesi . . . . .	5
1.5 Pacchetti che il laureando deve caricare personalmente . . . . .	7
1.6 Pacchetti da non caricare affatto . . . . .	8
1.7 Comandi e ambienti di TOPtesi . . . . .	8
1.8 Modelli di tesi e di frontespizi . . . . .	9
<b>2 Introduzione</b>	<b>11</b>
2.1 L <sup>A</sup> T <sub>E</sub> X e le tesi di laurea . . . . .	12
2.2 Installare L <sup>A</sup> T <sub>E</sub> X . . . . .	14
2.3 Ora siete pronti . . . . .	18
<b>3 L'uso di TOPtesi</b>	<b>21</b>
3.1 Impostazioni standard di TOPtesi . . . . .	21
3.1.1 Dove sono i file di TOPtesi? . . . . .	22
3.1.2 Il file di configurazione . . . . .	23
3.2 Pronti? Via! . . . . .	24
3.3 I file accessori . . . . .	24
3.3.1 Comporre il frontespizio con topfront . . . . .	25
3.3.2 I comandi accessori di topcoman . . . . .	30
3.4 La composizione del frontespizio . . . . .	32
3.4.1 Il logo dell'ateneo per l'uso con TOPtesi e TOPfront . . . . .	33
3.5 Comporre il frontespizio senza ricorrere a TOPfront . . . . .	35
3.5.1 Comporre il frontespizio con l'ambiente titlepage . . . . .	36
3.5.2 Comporre il frontespizio con il pacchetto <i>frontespizio</i> . . . . .	40



3.6	Come si comincia . . . . .	42
3.6.1	La scelta della codifica per il file sorgente . . . . .	45
3.6.2	La scelta della codifica per il file di uscita . . . . .	48
3.7	Come stampare la tesi . . . . .	50
3.8	Comporre la tesi in diverse lingue . . . . .	51
3.8.1	Configurazione iniziale . . . . .	51
3.8.2	Le lingue della tesi . . . . .	54
3.9	Il formato PDF/A . . . . .	57
3.9.1	pdf <sub>l</sub> atex e il formato PDF/A . . . . .	61
3.9.2	lua <sub>l</sub> atex e il formato PDF/A . . . . .	63
3.9.3	xelatex e il formato PDF/A . . . . .	63
3.9.4	Uso di ghostscript o di <i>pdfpages</i> . . . . .	64
3.10	Verifica della conformità . . . . .	68
<b>4</b>	<b>I comandi specifici introdotti da TOPtesi</b>	<b>71</b>
4.1	Introduzione . . . . .	71
4.2	Le opzioni . . . . .	71
4.3	Comandi di tipo generale . . . . .	73
4.4	Comandi per il frontespizio . . . . .	77
4.5	Altri comandi . . . . .	82
	<b>Conclusioni</b>	<b>89</b>
<b>A</b>	<b>The L<sup>A</sup>T<sub>E</sub>X Project Public License</b>	<b>91</b>
A.1	Preamble . . . . .	91
A.2	Definitions . . . . .	91
A.3	Conditions on Distribution and Modification . . . . .	92
A.4	No Warranty . . . . .	94
A.5	Maintenance Of the Work . . . . .	95
A.6	Whether and How to Distribute Works under This License . . . . .	96
A.6.1	Choosing This License or Another License . . . . .	96
A.6.2	A Recommendation on Modification Without Distribution . . . . .	97
A.6.3	How to Use This License . . . . .	97
A.6.4	Important Recommendations . . . . .	98
	<b>Indice analitico</b>	<b>99</b>

# Elenco delle tabelle

2.1	Le principali differenze fra pdf <code>latex</code> e x <code>elat</code> x . . . . .	19
3.1	Varie modalità di composizione del frontespizio . . . . .	30
4.1	Comandi di tipo generale . . . . .	74
4.2	Comandi ulteriori di <i>toptesi</i> . . . . .	83
4.3	Comandi ulteriori per il frontespizio e per il corpo della tesi definiti con l'opzione <i>classica</i> . . . . .	84
4.4	Comandi per il frontespizio della monografia di laurea, della tesi di laurea e della dissertazione di dottorato ( <i>continua</i> ) . . . . .	85
4.4	Comandi per il frontespizio della monografia di laurea, della tesi di laurea e della dissertazione di dottorato ( <i>continua</i> ) . . . . .	86
4.4	Comandi per il frontespizio della monografia di laurea, della tesi di laurea e della dissertazione di dottorato. Comando per comporre il verso del frontespizio. . . . .	87
4.5	Comandi per modificare le parole e/o le brevi frasi scritte nel frontespizio o per cambiare le parole italiane di default in altre parole diverse, o per esprimere gli stessi concetti in un'altra lingua . . . . .	88

# Elenco delle figure

3.1	Frontespizi composti con lo stile standard . . . . .	29
3.2	Un logo modificato con l'aggiunta del nome dell'ateneo . . . . .	30
3.3	Frontespizi composti con lo stile classica . . . . .	31
3.4	Frontespizio con l'orientamento landscape ottenuto con l'ambiente <i>title-</i> <i>page</i> . . . . .	39
4.1	I quattro frontespizi fondamentali . . . . .	80



# Capitolo 1

## Guida rapida all'uso di questo manuale

Nella pagina 18 c'è scritto in rosso quanto segue.

Dopo avere letto un po' di documentazione e aver giocato un poco con i programmi già predisposti sia dalla distribuzione del sistema  $\text{\TeX}$  sia dai vari editor ASCII, siete in grado di capire come funziona il tutto.

Attenzione: non si minimizzi la frase precedente: la documentazione va letta sempre e capita fino in fondo; se c'è qualcosa che non capite, provate ad esercitarvi con qualche piccolo esercizio, visto che la pratica permette di capire la teoria (e viceversa), ma non andate a cercare in rete aiuto nei vari forum dedicati a  $\text{\LaTeX}$  per farvi dire: “guarda nella pagina tal-dei-tali della documentazione”; per voi sarebbe una vera umiliazione...

Detto in altre parole, non cominciate nemmeno a leggere questo manuale e non usate TOPtesi se non avete ancora un minimo di conoscenza di  $\text{\LaTeX}$ . Non è solo una questione di esperienza pratica; è anche una questione di linguaggio; se non si conosce la terminologia, non si capisce nemmeno quello che si legge. Faccio solo un paio di esempi che mostrano l'ambiguità di certi termini, dei quali bisogna conoscere il significato e bisogna saperli distinguere in base al contesto.

**Pacchetto** In inglese esistono due termini usati nel sistema  $\text{\TeX}$ : *bundle* e *package*: il primo termine si riferisce ad una collezione di diversi file, per lo più di tipo *package*, ma non solo; i file di tipo *package* sono delle collezioni di definizioni, le cosiddette macro. In italiano i due termini vengono abitualmente tradotti entrambi con il nome *pacchetto*, per cui la parola italiana è ambigua.

**Formato** Nel mondo  $\text{\TeX}$  la parola *formato*, in inglese *format*, ha almeno tre significati; i principali significati sono i seguenti.

**Forma del mark up** Con questo significato ci si riferisce al file con estensione *.fmt* che contiene la traduzione in linguaggio macchina dell'insieme di macro che definiscono il mark up specifico del linguaggio usato; qui potrebbero

interessare i file di formato `pdflatex.fmt`, `xelatex.fmt`, `lualatex.fmt`; ne esistono diversi altri.

**Forma della pagina del testo composto** Le varie carte disponibili vengono vendute in diversi formati; per esempio A4 (210 mm per 297 mm), B5 (176 mm per 210 mm), eccetera. Si può usare questa parola anche per riferirsi alla forma della gabbia del testo e al layout della pagina composta tipograficamente.

**Codifica di registrazione delle immagini** La parola formato viene usata per specificare in realtà il modo di codificare una immagine, e si parla del formato JPEG, del formato PDF, eccetera. Esistono formati vettoriali e formati raster (o a matrici di punti, o bitmapped); i formati raster possono essere *lossless* oppure *lossy*; in questo caso la perdita, da cui il prefisso “loss”, si riferisce al modo di comprimere l’informazione dell’immagine: la compressione senza perdita permette di recuperare esattamente l’immagine di partenza, mentre la compressione con perdita permette di comprimere di più ma a spese di una approssimazione nel recupero dell’immagine di partenza; il formato PNG è di tipo lossless; il formato JPG (o JPEG) è lossy.

## 1.1 A cosa serve TOPtesi

Serve per comporre una tesi, sia essa la *monografia* (detta anche *elaborato finale*) preparata alla fine della laurea triennale, o la *tesi di laurea* predisposta alla fine della laurea magistrale o della laurea a ciclo unico, sia essa la *dissertazione di dottorato*. Sia essa da scrivere in italiano o un’un’altra lingua.

La tesi, in realtà, non è altro che il rapporto relativo allo studio, alla ricerca, alle sperimentazioni, al progetto, svolti come lavoro conclusivo di un periodo di studi universitari. Il contenuto della tesi non differisce sostanzialmente da qualunque altro rapporto scritto in una qualsiasi disciplina su cui verta principalmente la tesi.

Questa classe è già stata usata per comporre tesi di vario livello in ingegneria, in matematica, in fisica, in economia, in filologia greca, in filologia copta, in medicina, eccetera. Non è quindi destinata solo ai rapporti finali degli studi di ingegneria.

La “tesi” differisce da un generico rapporto, perché ha un valore legale come elaborato da presentare all’esame finale per un ciclo di studi superiori; deve quindi avere certi requisiti che permettano di soddisfare le richieste di tipo burocratico di ogni ateneo.

Questi requisiti riguardano principalmente il frontespizio; quindi in questo manuale si dedicherà molto spazio alla predisposizione del frontespizio da comporre con vari stili, e in diverse lingue.

TOPtesi serve anche per fornire alcune altre semplici estensioni che rendono più agevole la redazione del contenuto della tesi, ma in fondo non è questa la parte più importante di TOPtesi.

Come tutti i pacchetti che estendono la funzionalità di  $\LaTeX$ , TOPtesi definisce la geometria della pagina e dispone le informazioni accessorie come le testatine e i piedini. Queste impostazioni non sono modificabili né con TOPtesi né con la maggior parte degli altri pacchetti destinati alle tesi. Vale il concetto “prendere o lasciare”.

Prima di usare TOPtesi si esaminino i semplici esempi contenuti nei file accessori di questo pacchetto: `toptesi-example.pdf`, `toptesi-example-luatex.pdf`, `top-front.example.pdf`, `toptesi-example-con-frontespizio.pdf`. Si potrà vedere se l’impaginazione aggrada oppure se si desidera un’altra impaginazione; in questo secondo caso ci si rivolga ad altre classi e ad altri pacchetti: in rete non è difficile usare un motore di ricerca per trovare il pacchetto *frontespizio*, le classi *sapthesis*, *suftesi*, *TesiModerna*, *TesiClassica* e diverse altre classi prodotte specialmente nel mondo angloamericano e ancor meno adatte ad una personalizzazione multilingue. Alcune di queste classi sono dotate di eccellenti file di documentazione, altre sono usabili grazie a “template” o modelli di tesi scritte facendo uso di testo fittizio, ma tali da rendere immediatamente l’idea di come usare quei software.

Volendo, anche i file sorgente di questi esempi d’uso del pacchetto TOPtesi, elencati sopra con i nomi dei corrispondenti file PDF, possono essere usati come modelli; persino il file sorgente di questo manuale può essere usato come modello; basta cercare nelle cartelle dell’installazione di TOPtesi i file `.tex`, copiarseli in una propria cartella personale, cambiare loro il nome (attenzione, questo è importantissimo) e modificarne il contenuto a proprio piacimento. Con un minimo di attenzione si possono eliminare quelle parti che non servono; si possono modificare le opzioni della classe, si possono compilare con diversi motori di tipocomposizione del sistema  $\TeX$ . Non è forse fuori luogo sottolineare che l’uso dei modelli può essere molto utile per cominciare, ma fa perdere di vista il fatto che  $\LaTeX$  può fare molte più cose di quelle che si inseriscono solitamente negli esempi.

## 1.2 Cosa leggere

Un manuale di solito non è da leggere dalla prima pagina all’ultima. Se ne possono saltare diverse sezioni, ma è bene sapere che cosa si salta, quindi una rapida sfogliata delle pagine che si saltano non fa male.

Se avete già il sistema  $\TeX$  installato nel vostro PC, potete saltare buona parte del capitolo 2, ma almeno una volta conviene leggere la parte che precede il primo paragrafo numerato.

Del capitolo 3 è opportuno leggere il paragrafo 3.1.2, perché descrive l’uso di un file di configurazione, che non è obbligatorio usare, ma che risulta molto comodo.

Conviene leggere l’uso dei loghi nel paragrafo 3.4.1, perché TOPtesi consente di usare diversi loghi nel frontespizio e li può collocare in posti diversi della pagina.

Il paragrafo 3.6 espone come impostare inizialmente il documento principale della tesi e di come frazionarne il contenuto in diversi file. Non esiste un unico metodo e i vari

metodi presentano vantaggi e svantaggi. Conviene conoscere bene questi metodi e i loro pro e contro, anche perché nell’uso elementare di  $\text{\LaTeX}$  questi concetti non vengono mai affrontati.

Il paragrafo 3.6.1 è fondamentale; disporre di un editor che salva i file sorgente con una certa codifica e poi cercare di compilare quei file con  $\text{\LaTeX}$  impostato per una codifica diversa, vuol dire combinare pasticci inenarrabili; in questo manuale si consiglia di usare sia per l’editor sia per  $\text{\LaTeX}$  la codifica `utf8` ma è il laureando che deve sceglierla in base alle caratteristiche del suo software e ai programmi che intende usare per la compilazione. Tutti e tre i programmi principali di composizione,  $\text{pdf\LaTeX}$ ,  $\text{Xe\LaTeX}$ , e  $\text{Lua\LaTeX}$  funzionano bene con la codifica `utf8`;  $\text{pdf\LaTeX}$  funziona anche con altre codifiche; gli editor un po’ datati non funzionano con la codifica `utf8`, quindi è evidente che le varie situazioni richiedono impostazioni attente e accurate. Se fosse necessario, l’argomento codifiche può essere approfondito anche su altri testi liberi, per esempio la guida tematica che si trova nella sezione documentazione dell’associazione  $\text{\GUIT}$ , in <http://guitex.org/home/images/doc/GuideGuIT/introcodifiche.pdf> dal titolo *Introduzione alle codifiche in entrata e in uscita*.

Il paragrafo 3.8 descrive come impostare il file sorgente della tesi per comporre il testo in diverse lingue o per impostare una lingua principale diversa dall’italiano. È essenziale per comporre tesi in programmi di doppia laurea. Infatti TOPtesi è stato creato anche per poter soddisfare le esigenze che un numero sempre maggiore di laureandi incontrano quando partecipano a programmi di doppio titolo con i vari programmi europei Erasmus, Life Long Learning, Erasmus Mundus, eccetera; le tesi svolte in questi percorsi solitamente richiedono l’uso di altre lingue oltre o in sostituzione dell’italiano. TOPtesi viene incontro a queste esigenze sia grazie all’uso di  $\text{\LaTeX}$  o  $\text{Xe\LaTeX}$  o  $\text{Lua\LaTeX}$  che sono in grado di gestire una ottantina di lingue, sia perché quasi tutti i comandi che devono essere usati per il frontespizio e per molte strutture interne sono completamente configurabili in accordo con le lingue usate. Il paragrafo 3.8 provvede a spiegare come eseguire queste configurazioni.

Per la questione codifiche e la questione lingue, che sono due aspetti che hanno forti collegamenti, è opportuno tenere presenti le seguenti considerazioni.

TOPtesi non serve solo per comporre tesi di laurea in ingegneria. Sono al corrente che il pacchetto TOPtesi è stato usato per comporre almeno una tesi di filologia greca classica, e almeno una tesi di commento ad un testo copto altomedievale. Di queste sono sicuro, ma ho informazioni indirette che sono state composte altre tesi in lingue moderne e antiche che facevano uso di alfabeti diversi da quello cosiddetto “latino”. Al tempo di quelle due tesi sul greco antico e sul copto altomedievale non esistevano ancora i programmi del sistema  $\text{\TeX}$ ,  $\text{Xe\LaTeX}$  e  $\text{Lua\LaTeX}$ ; oggi che sono disponibili forse sarebbero state composte più facilmente con uno di questi due programmi; ma con questi è obbligatorio usare la codifica d’entrata `utf8`<sup>1</sup> e font codificati UNICODE, come i font OpenType.

---

<sup>1</sup>In realtà si potrebbero anche usare le codifiche a 8 bit tipiche di  $\text{pdf\LaTeX}$ , ma bisogna ricorrere ad



La codifica di entrata `utf8` è consigliabile anche se si usa `pdfLATEX`, ma non è imposta, quindi TOPtesi è “indifferente” alla codifica d’entrata purché glielo si dica: è quindi compito e responsabilità del laureando quello di specificare la codifica d’entrata. Similmente in relazione alla lingua o alle lingue usate è compito del laureando specificare la o le codifiche dei font da usare e di specificare i nomi dei font da usare.

Infine il capitolo 4 contiene tutti i comandi e gli ambienti introdotti da TOPtesi per la composizione del frontespizio, per la personalizzazione del frontespizio, per la composizione di strutture di testo o di figure in estensione a quelle normali di L<sup>A</sup>T<sub>E</sub>X, e via di questo passo.

## 1.3 Errori da evitare

Tuttavia si abbia anche cura di **non copiare nel preambolo di una tesi da comporre con TOPtesi un preambolo recuperato dalla tesi di un amico o, peggio ancora, dalla rete!**. Sarebbe una operazione fonte di molte delusioni, perché da una parte potrebbero venire caricati pacchetti incompatibili con TOPtesi, dall’altro alcuni comandi di TOPtesi ne potrebbero venire modificati con funzionalità diverse da quelle previste per TOPtesi. Potrebbero anche manifestarsi dei conflitti fra pacchetti caricati nel preambolo e quelli già caricati da TOPtesi.

## 1.4 Pacchetti già caricati da TOPtesi

Vale la pena di elencare i pacchetti già caricati dalla classe *toptesi*, al fine di evitare di ricaricarli nel preambolo.

*toptesi* è il file di macro di TOPtesi; potrebbe anche essere usato con una classe diversa da *toptesi*.

*graphicx* serve per gestire diverse funzioni grafiche e per l’inclusione di file grafici esterni, come fotografie, disegni, e simili.

*etoolbox* è un pacchetto di servizio, le cui funzionalità possono essere usate anche da un laureando molto competente; questo pacchetto agevola moltissimo la gestione dei file di classe e di quelli di estensione. Se il laureando ha sufficiente padronanza di L<sup>A</sup>T<sub>E</sub>X, può definirsi altri comandi specifici per la sua tesi, sfruttando le funzionalità avanzate di questo pacchetto.

---

artifici che non vale la pena usare quando sono disponibili i font OpenType; sarebbe stato necessario farlo per il copto altomedievale e per il copto liturgico, per i quali ho creato solo font con codifiche a 8 bit.

*topfront* contiene i comandi specifici per comporre il frontespizio; potrebbe venire anche usato da solo con un'altra classe; mediante l'opzione `noTOPfront` per la classe *toptesi* se ne può inibire il caricamento, così che l'operatore possa comporre il frontespizio con altri pacchetti esterni, o possa comporlo a modo suo sfruttando come meglio crede l'ambiente *titlepage*. La classe *toptesi* carica questo modulo solo durante l'esecuzione del comando `\begin{document}`, quindi nessun comando che abbia a che fare con il frontespizio può essere usato nel preambolo, in quanto *topfront* non è ancora stato letto e quindi i comandi non sono ancora stati definiti.

*topcoman* un piccolo pacchetto che contiene alcuni comandi utili in generale, non solo per l'uso con TOPtesi. Non è possibile inibirne il caricamento, ma spero che provveda da solo a non entrare in conflitto con altri pacchetti; finora non ho incontrato conflitti.

*ifxetex* serve per distinguere il motore di composizione con cui si compila la tesi.

`<nome del main file>.cfg` Il file di configurazione specifico per una data tesi viene caricato solo se esiste nella medesima cartella dove risiede il main file della tesi stessa.

*babel* viene caricato con le opzioni `english`, `italian`, cosicché l'italiano risulta svolgere le funzioni della lingua principale; questo pacchetto viene caricato solo se si compone la tesi con pdf $\LaTeX$ , e TOPtesi riconosce da solo quale sia il motore di composizione usato.

*polyglossia* viene caricato solo se la tesi viene composta con Xe $\LaTeX$  o Lua $\LaTeX$ . L'italiano viene specificato come lingua principale, e l'inglese come altra lingua.

*pdfx* deve venire caricato esplicitamente dall'utente, come spiegato nell'apposito paragrafo; nelle versioni precedenti esso veniva caricato specificando l'opzione `pdfa` per produrre un file possibilmente conforme al formato PDF/A. Ora, per compatibilità con il passato, l'opzione è ancora attiva, ma il suo scopo è solo quello di emettere un avviso di consultare la documentazione.

*hyperref* serve per comporre i collegamenti ipertestuali. Non è il caso di preoccuparsi di quando TOPtesi carica *hyperref* perché ci pensa lui a ritardarne la chiamata al momento di iniziare la composizione del documento. Si veda più avanti per sapere come e quando eventualmente configurare *hyperref* con le sue opzioni.

Il laureando è tenuto a documentarsi su ciascuno di quei pacchetti; normalmente egli dispone di tutta la documentazione di cui necessita già nella sua installazione del sistema TeX completo e aggiornato; basta che apra un terminale e vi scriva dentro;

`texdoc <nome del pacchetto>`

e, dopo aver premuto il tasto invio, sullo schermo del suo PC si apre la finestra che contiene la documentazione.

Conoscere l'elenco di questi pacchetti è importante proprio per non ricaricarli e per evitare conflitti quando si specificano opzioni diverse.

## 1.5 Pacchetti che il laureando deve caricare personalmente

Si noti: non sono precaricati i pacchetti *inputenc* per definire la codifica d'entrata; *fontenc* per definire la codifica dei font di uscita; e non è preimpostato nessun font particolare da usare per la composizione della tesi.

È voluto: *inputenc* non deve essere caricato se si usano  $\text{Xe}\text{L}\text{A}\text{T}\text{E}\text{X}$  o  $\text{Lua}\text{L}\text{A}\text{T}\text{E}\text{X}$ , ma in entrambi i casi l'editor che si usa per comporre il file sorgente *deve* essere configurato in modo che salvi i file sorgente con la codifica `utf8`. Con  $\text{pdf}\text{L}\text{A}\text{T}\text{E}\text{X}$  ci sarebbe una certa libertà nello scegliere la codifica d'entrata, ma, insisto, sarebbe meglio in ogni caso evitare di usare qualunque altra codifica diversa da `utf8`.

Per i font di uscita la o le codifiche da specificare dipendono dalle lingue usate; ripeto la raccomandazione di preferire i font espressamente confezionati per l'uso con  $\text{pdf}\text{L}\text{A}\text{T}\text{E}\text{X}$ , quando si compone con questo programma; se si usano i programmi  $\text{Xe}\text{L}\text{A}\text{T}\text{E}\text{X}$  o  $\text{Lua}\text{L}\text{A}\text{T}\text{E}\text{X}$  si abbia l'accortezza di usare il pacchetto *fontspec* specificandogli le opzioni giuste e caricando poi, mediante le sue funzionalità, i font OpenType di cui si sia accertata la presenza sulla propria macchina, e si sia verificato che contengano tutti i glifi che si intendono usare nella tesi. Tanto per citare l'importanza di questa verifica, questi programmi lavorano di default con i font Latin Modern, che, come dice il nome, contengono *solo* i caratteri latini. Se ci fosse bisogno di scrivere in greco o in cirillico, per esempio, allora i font OpenType UCM (distribuiti con il sistema  $\text{T}\text{E}\text{X}$  completo e aggiornato) contengono anche questi alfabeti.

Ecco quindi che la specificazione delle codifiche dei font di uscita è importante per  $\text{pdf}\text{L}\text{A}\text{T}\text{E}\text{X}$  perché con questo programma i font predefiniti sono i Computer Modern codificati in OT1, cioè mancano di qualunque segno accentato e non sono nemmeno completamente compatibili con la codifica ASCII. Questi font preimpostati potevano (forse) andare bene ai primordi dell'esistenza del sistema  $\text{T}\text{E}\text{X}$  ma non vanno bene oggi nemmeno per l'inglese, visto che anche in inglese si fanno citazioni di testi o di nomi di persone in lingue straniere che usano caratteri latini accentati.

Per motivi diversi non si sono caricati i pacchetti per la composizione della matematica estesa, *amsmath*, *amssymb* (che a sua volta carica *amsfonts*); né i pacchetti *amsthm* per la definizione di enunciati come teoremi, lemmi corollari, definizioni e simili, né il pacchetto *bm* per comporre in neretto simboli isolati di una espressione matematica.

Non si sono caricati nemmeno i pacchetti per il disegno programmato come, per esempio, *tikz*; a un laureando in ingegneria certamente servono, a un laureando in letteratura medievale difficilmente potrebbero servire, ma non venga in mente al laureando di caricare quei font che non gli permettono di comporre la matematica secondo le norme ISO-UNI; fra questi ci sono i font *euler*, certamente molto belli, ma non consentono di rispettare le norme ISO-UNI, perché il font non è inclinato come invece deve essere un buon corsivo matematico (*math italics*) e molti caratteri non sono facilmente distinguibili da quelli composti in tondo (*roman*).

## 1.6 Pacchetti da non caricare affatto

Non si devono assolutamente caricare pacchetti che modifichino l'aspetto della pagina, né caricare pacchetti che modifichino la composizione dei titoli o delle didascalie o delle note; eccezionalmente il pacchetto *caption* può venire caricato; *toptesi* verifica se tale pacchetto è stato effettivamente caricato e, nel caso, non definisce nemmeno la macro per le didascalie esplicitamente creata per la composizione delle tesi.

Se si vogliono fare questo genere di modifiche, stile della pagina, testatine, piedini, note, ... è meglio rivolgersi ad altri pacchetti diversamente configurabili; cito fra gli altri *suftesi*, *sapthesis*; ma quando si va a leggerne la documentazione si scopre che tutti vietano la modifica dell'aspetto della pagina o dei titoli o delle didascalie, a meno che quelle stesse classi non dispongano di comandi già predisposti per la personalizzazione di quegli elementi.

Se il laureando va a cercare in rete altri pacchetti per comporre tesi, si trova davanti allo stesso ostacolo. Se la propria università prescrive stili di pagina diversi o mette a disposizione file classe appositi (di solito piuttosto datati e costruiti male), si renderebbe necessario creare una classe apposita per soddisfare quelle esigenze. Non saprei cosa consigliare, se non ricorrere ad una classe generica come *book* e caricare tutti i possibili pacchetti di configurazione che si considerino necessari, per arrivare ad un risultato come quello richiesto dalla propria sede universitaria; i pacchetti preconfezionati come *TOPtesi*, e gli altri citati sopra, risparmiano questo lavoro, ma sono rigidi. In fondo anche *TOPtesi* è costruito così: parte dalla classe *report* e vi costruisce attorno quello che si è voluto fare in base a prescrizioni valide per il Politecnico di Torino, ma lungamente discusse e concordate con l'ateneo e poi rese compatibili con le prescrizioni di diverse altre università. Non è certo un pacchetto perfetto, ma è un buon compromesso.

## 1.7 Comandi e ambienti di TOPtesi

Nel capitolo 4 sono descritti i comandi specifici e gli ambienti introdotti da *TOPtesi* in aggiunta a quelli della classe *report* originali o modificati da *TOPtesi*.

La maggior parte di questi comandi si riferisce alla compilazione del frontespizio, ma è bene che il laureando li abbia sempre a portata di mano per poter eseguire le molteplici personalizzazioni che sono offerte da *TOPtesi*, in particolare dal suo pacchetto *topfront*. Quel capitolo contiene anche le figure che rappresentano otto tipici frontespizi in italiano, per elaborati finali della laurea triennale, per la laurea magistrale, per la dissertazione dottorale, sia svolte in singoli atenei, sia svolte in atenei associati. Sia con i loghi in testa alla pagina sia con questi loghi nella metà inferiore della pagina. Sono tante varianti che possono soddisfare molte esigenze, ma che evidentemente non le soddisfano tutte. Queste sono quelle previste e il pacchetto *topfront* non è abbastanza elastico per gestirne altre. Se si vogliono stili diversi esiste sempre il pacchetto *frontespizio* creato apposta da un altro docente universitario di un altro ateneo, quindi con un'altra visione d'insieme sull'aspetto

dei frontespizi. Infine esiste sempre la possibilità di usare l'ambiente *titlepage* per creare il frontespizio in modo assolutamente libero da ogni vincolo.

In quel capitolo sono anche rappresentati i quattro frontespizi fondamentali nella figura 4.1; è importante avere queste quattro figurine sotto gli occhi per sapere come comporre il proprio frontespizio in modo che corrisponda alle prescrizioni del proprio ateneo.

Il laureando non dedichi invece troppo tempo alla lettura del paragrafo 3.9 e seguenti, perché vi si parla di come comporre la tesi in modo che soddisfi alle prescrizioni di archiviabilità introdotte dalle norme ISO 19005 e successive. Quanto scritto in quei paragrafi serve solo se è richiesta la tesi in versione archiviabile secondo le norme ISO. In ogni caso, se non è richiesto, non è opportuno addentrarsi per questa strada; se è richiesto, le operazioni necessarie si possono eseguire a tesi completata come ultimo tocco finale.

## 1.8 Modelli di tesi e di frontespizi

Ricordo infine che il pacchetto TOPtesi contiene anche un certo campionario di modelli di tesi e di frontespizi di vario genere, che possono essere composti con diversi motori di composizione. Il laureando può servirsene in modo molto semplice: copia il file di esempio nella propria cartella di lavoro, gli cambia nome e poi modifica il modello commentando o de-commentando alcune righe, togliendo parti che non servono; per esempio vi ho messo un piccolo esempio di ringraziamenti, ma i ringraziamenti non andrebbero mai usati – vedi più avanti perché; vi ho messo un piccolo esempio di dedica, ma le dediche sono superflue nel 99% dei casi; eccetera.

Ripulito il preambolo delle cose che non servono, e scelte le righe ritenute necessarie, basta cambiare i nomi di fantasia che ho usato e i testi di fantasia che ho inserito per ottenere lo schema della tesi, che va poi riempito col contenuto relativo alla tesi che si vuole effettivamente comporre.



## Capitolo 2

### Introduzione

Si legge ancora nelle istruzioni per scrivere le tesi di molte università italiane e straniere:

Comporre la tesi con interlinea 2 e con righe di 60 battute; ...

Quelle università non si sono ancora accorte che le macchine da scrivere meccaniche o elettromeccaniche sono rimaste oggetti di sola curiosità, ammesso che ce ne sia ancora qualcuna disponibile e le poche superstiti non siano tutte nei musei.

Oggi si scrive con uno dei tanti sistemi di elaborazione di testi, detti anche “word processor”, che fanno parte più o meno di default di ogni dotazione iniziale di qualsiasi PC di qualunque marca e con qualunque sistema operativo.

Fra i vari programmi disponibili, uno in particolare spicca per la sua particolarità:  $\text{\LaTeX}$ . Veramente esso non è un word processor, anche se a prima vista lo sembrerebbe; il suo scopo non è finalizzato al testo in quanto tale, bensì alla sua composizione tipografica; esso è un programma di *tipocomposizione*.

$\text{\LaTeX}$  è nato come sovrastruttura di un altro programma,  $\text{\TeX}$ , che continua ad esserne il motore. La prima versione di  $\text{\TeX}$  è stata creata nel 1978, ma è ancora usata oggi, naturalmente molto aggiornata e ampliata, e questo fatto è una cosa insolita nel panorama turbolento di novità dell'informatica.

Secondo me, il suo successo è dovuto a due fatti: (a) esso è stato progettato e implementato da un matematico per comporre i suoi stessi libri di informatica matematica; (b) egli l'ha messo a disposizione di chiunque, sin dal primo momento, come software libero.

Oggi il software libero è piuttosto diffuso, ma nel 1978 parlare di software libero era quasi una bestemmia.

D'altra parte Donald E. Knuth non era soddisfatto della bassa professionalità che anno dopo anno manifestavano i compositori delle case editrici, i quali anno dopo anno si abituavano a quanto i programmi di elaborazione mettevano loro a disposizione, ma contemporaneamente perdevano le loro conoscenze professionali via via che si adattavano a quanto quei programmi consentivano loro di fare.

La cosa era o stava diventando insostenibile durante i vari anni in cui uscivano i successivi volumi dell'opera di Knuth *The Art of Computer Programming*, e così Knuth si dedicò alla creazione della tipografia elettronica realizzando il programma  $\text{\TeX}$  che, ripeto, è ancora in ottima salute e molto vivace dopo oltre trent'anni di onorato servizio.

L'uso di  $\text{\TeX}$  per eseguire direttamente la composizione era piuttosto difficile, e ogni utente doveva prima o poi imparare a scriversi delle macroistruzioni che gli consentissero di agevolare il suo lavoro.

Nel 1984  $\text{\TeX}$  era già così diffuso in tutto il mondo, specialmente in ambito accademico, che Leslie Lamport decise di produrre un sistema quasi completo di macro che consentisse agli utenti di usare  $\text{\TeX}$  lasciandolo dietro le quinte, in modo da potersi concentrare sul contenuto dei loro scritti e non sulla forma da dare a questo o a quel dettaglio.

Nel 1990 Knuth pubblicò la versione di  $\text{\TeX}$  che consentiva di comporre in diverse lingue simultaneamente; nel 1994 molti utenti di  $\text{\LaTeX}$  costituirono il  $\text{\LaTeX3}$  Team al fine di rendere gestibile la mole enorme delle estensioni di  $\text{\LaTeX}$  che in 10 anni utenti entusiasti avevano messo a disposizione della comunità degli altri utenti. Insomma è successo con  $\text{\TeX}$  e  $\text{\LaTeX}$  quello che succede normalmente con il software libero.

Attenzione: Knuth paga di tasca sua un assegno a chiunque trovi un errore nel suo software; a tutt'oggi non è andato in bancarotta, sia perché gli errori sono rarissimi, sia perché quelle poche persone che hanno segnalato errori veri e hanno ricevuto l'assegno di Knuth, non l'hanno incassato ma l'hanno incorniciato come una reliquia preziosa.

## 2.1 $\text{\LaTeX}$ e le tesi di laurea

Ovviamente  $\text{\LaTeX}$  serve per scrivere qualunque cosa; o meglio; serve per comporre tipograficamente qualunque testo.  $\text{\LaTeX}$  non è un programma di impaginazione, è un programma di composizione tipografica. Non aspettatevi quindi di poter fare qualunque acrobazia con le righe di testo, come per esempio piegarle, deformando i caratteri che vi sono appoggiati sopra, mettendovi attorno aloni di luce cangiante, sfumature, ombreggiature, evidenziandone i contorni, eccetera. Queste cose sono riservate ai creativi che si occupano di pubblicità.

Aspettatevi invece di comporre testi in cui ogni capoverso è ottimizzato per avere il minor numero di parole divise in sillabe in fin di riga, e di avere il minor numero possibile di “ruscelli” fra le parole grazie alla uniformità dello spazio interparola; aspettatevi di comporre formule complicatissime con il minimo di sforzo da parte vostra ma con la certezza che esse saranno composte come nessun altro programma riesce a fare. Se state componendo una tesi nel campo delle scienze umane aspettatevi il meglio in assoluto; se poi vi interessate di filologia di lingue antiche, non c'è altro programma che possiate usare a questo scopo. Aspettatevi uno stampato estremamente professionale.

Per questo motivo voi studenti che userete questo pacchetto di macroistruzioni chiamato  $\text{\LaTeX}$  dovreste astenervi dall'introdurre errori compositivi così da vanificare quanto di bello riesce a produrre  $\text{\LaTeX}$ .



Inizialmente vi troverete un po' a disagio perché vi siete abituati anche voi ai programmi commerciali che consentono la “composizione sincrona”; vi consentono di vedere subito sullo schermo il frutto del vostro lavoro. Per ottenere questo risultato questi programmi hanno necessariamente rinunciato a diverse funzioni, badando invece a presentare sullo schermo con la massima velocità il testo composto.

$\LaTeX$  richiede che voi scriviate un testo non formattato in puri caratteri ASCII<sup>1</sup>, ma marcato con un particolare sistema di *mark-up* che consente di sapere che cosa sia ogni parte del vostro scritto: una equazione, una citazione, una enumerazione, una elencazione, una descrizione, una poesia, una bibliografia, una epigrafe, eccetera. Ci pensa poi  $\LaTeX$  in un secondo tempo a dare forma al vostro testo e, in particolare, a dare la stessa forma a ogni elemento del vostro scritto a seconda di come lo abbiate “marcato”; elementi marcati nello stesso modo vengono composti nello stesso modo. Così si evitano quelle disuniformità compositive che si notano assai spesso quando si usa un word processor comune.

Attenzione: questa guida presuppone che voi abbiate già una conoscenza di base del linguaggio  $\LaTeX$ . Se non l'avete ancora, installate pure il programma, ma non usate questa guida per imparare a comporre testi con  $\LaTeX$ . Esistono diversi testi gratuiti in rete, da  *$\LaTeX$  per l'impaziente* a *L'Arte di scrivere con  $\LaTeX$*  a *Introduzione all'arte della composizione tipografica con  $\LaTeX$* ; basta cercare questi titoli in rete e si troverà da dove scaricarli.

Ricordatevi però che il vostro primo problema sarà quello di superare lo scoglio psicologico di non vedere subito il frutto delle vostre fatiche; abituati come siete ai word processor che seguono il paradigma *What you see is what you get* (Quello che vedi è quello che ottieni), che più realisticamente andrebbe scritto *What you see is all you can get* (Quello che vedi è tutto ciò che puoi ottenere), dovete passare al paradigma: *What you see is what you mean*, dove l'aspetto grafico di quel che si vede sullo schermo del PC in fase di scrittura iniziale non è importante, ma è importante quello che avete scritto; il suo aspetto grafico gli verrà dato da  $\LaTeX$  in un secondo tempo e in una maniera estremamente professionale.

Non pretendo di dire che non si possano scrivere tesi ancora più professionali di come si ottengono con TOPtesi, ma certo il risultato è molto migliore di quello che si può ottenere con qualsiasi word processor.

Vi sconsiglio fortemente di lasciarvi “attrarre” dall'usare programmi come LyX o TeXmacs; vi danno l'illusione di comporre come fareste con un comune word processor, anzi TeXmacs usa gli stessi font che userebbe  $\LaTeX$ ; è tutta una illusione; la comodità del comporre in modo da vedere sullo schermo qualcosa che vorrebbe essere molto simile a quanto  $\LaTeX$  produrrà, vi distrae dal vostro compito di fare attenzione al significato di quello che scrivete, e vi mettete a “giocare” con la sua forma. Non solo, ma quando vorrete estendere

---

<sup>1</sup>In realtà potete scrivere con qualunque set di caratteri e con qualunque *codifica*; oggi poi è possibile avere editor testuali che usano la codifica UNICODE, che consente, essendone capaci, di scrivere anche in cinese. Il documento che state leggendo è stato composto usando un file sorgente codificato in UNICODE; non contiene caratteri cinesi, ma avrei potuto farlo se conoscessi il cinese!

le capacità di presentare il testo che volete scrivere alla “ $\text{\LaTeX}$ ”, scoprirete ben presto che con LyX è molto difficile e con TeXmacs è impossibile.

Procuratevi invece un ottimo *shell editor* predisposto per lavorare con  $\text{\LaTeX}$ ,  $\text{\XeTeX}$  e  $\text{\LuaTeX}$ ; supererete ben presto l'imbarazzo di non poter vedere subito il risultato del vostro scrivere, ma sarete abbondantemente ricompensati dalle infinite possibilità compositive dei programmi del sistema  $\text{\TeX}$ . Più avanti ne parlerò diffusamente, ma qui ho voluto avvisarvi subito di non farvi incantare dal canto delle sirene come LyX e TeXmacs.

## 2.2 Installare $\text{\LaTeX}$

Per cominciare a lavorare con  $\text{\LaTeX}$  per prima cosa dovete installarvi il software; ovviamente se vi siete già installati l'intero sistema  $\text{\TeX}$  ed è già calibrato per comporre correttamente, potete saltare questo paragrafo.

Ci sono sostanzialmente tre situazioni.

**Windows** Gli utenti delle piattaforme Windows possono collegarsi in rete al sito [www.miktex.org](http://www.miktex.org) e scaricarvi ed installarsi la distribuzione di  $\text{\LaTeX}$ , o meglio, del sistema  $\text{\TeX}$  chiamata MiKTeX. Se si procede per questa via, allora si scarichi l'installazione completa, anche se si potrebbe installare la versione di base `small-miktex` che, seguendo le istruzioni di installazione, bisogna configurare in modo da consentirle di scaricare dalla rete ogni possibile pacchetto di estensione che possa via via essere necessario.

Ovviamente questo modo di installare e usare il sistema  $\text{\TeX}$  implica una connessione di rete sufficientemente veloce; personalmente, quando usavo una piattaforma Windows, ero solito scaricare la versione completa ed ero molto soddisfatto.

Oggi, però, anche su una macchina Windows consiglieri di installare la distribuzione  $\text{\TeX}$  Live che viene gestita nello stesso modo sia sulle piattaforme Windows, Linux e Mac. Il pregio è che  $\text{\TeX}$  Live viene aggiornata quasi quotidianamente sui server e comunque è la versione dalla quale il curatore di MiKTeX attinge per creare gli aggiornamenti della sua distribuzione; necessariamente, quindi, prima che il curatore sia riuscito a portare MiKTeX allo stesso livello di  $\text{\TeX}$  Live, passano di solito alcune settimane, a volte anche di più; non voglio togliere niente alla bravura del curatore, ma il progetto MiKTeX è il lavoro di un solo uomo, mentre  $\text{\TeX}$  Live è il frutto del lavoro di una squadra completa e attivissima.

**Linux** Gli utenti di Linux sono particolarmente fortunati perché il sistema  $\text{\TeX}$  è spesso parte integrante di qualunque variante di Linux, anche se non viene installato di default; basta inserire il disco di installazione o basta connettersi in rete e invocare uno dei vari programmi come `apt-get`, `rpm`, `yast` o ... per scaricare tutto quanto serve e per configurare l'installazione. Bisogna ricordarsi che Linux è un po' più “ruspante”

di Windows, e quindi la configurazione richiede un po' più di attenzione e di "smanettamento" con la riga di comando; ma a questo i "pinguini doc" ci sono abituati; in compenso si ha il beneficio di avere tutto quanto il software già predisposto fin dalla nascita per macchine UNIX e Linux e quindi si evitano tutti i (pochi) piccoli bug che si incontrano quando le cose sono tradotte per altri sistemi operativi.

Si faccia solo attenzione alle distribuzioni Debian; sono eccellenti e ne è garantita la compatibilità con ogni sistema operativo conforme ai dettami del consorzio Debian, ma solitamente sono in ritardo di alcuni mesi, in passato anche di un paio di anni, rispetto alle versioni aggiornate pubblicate dal  $\text{\TeX}$  Users Group! Appena si può si installi la distribuzione  $\text{\TeX}$  Live completa scaricata dal sito <http://www.tug.org/ctan.html> ufficiale. Esiste in rete un testo intitolato *texlive-ubuntu.pdf*<sup>2</sup>, che si trova con qualunque motore di ricerca; specifica come installare  $\text{\TeX}$  Live fresco di giornata (e aggiornabile sistematicamente come detto sopra per le macchine Windows) a fianco della distribuzione Debian di  $\text{\TeX}$  Live; quest'ultima serve per soddisfare le dipendenze di altri programmi Debian; la prima, invece, serve per lavorare davvero.

**Mac** Gli utenti delle piattaforme Macintosh e possessori di un portatile o di un desktop che funziona con il sistema operativo Mac OS X hanno anche loro a disposizione una distribuzione che si chiama Mac $\text{\TeX}$ , che viene installata e configurata con un particolare software adatto alla specificità del sistema operativo. Si cerchi il nome Mac $\text{\TeX}$  con un qualunque motore di ricerca che indicherà un sito dal quale si può scaricare il pacchetto di installazione (piuttosto grosso); alla fine del download viene chiesto se continuare con l'installazione; rispondendo affermativamente, il software viene scaricato sul disco che si sarà indicato, ma quel che è più comodo, esso è già completo, come ogni distribuzione e installazione  $\text{\TeX}$  Live. Il sistema Mac $\text{\TeX}$  produce il suo output essenzialmente in formato PDF; se ne tenga conto leggendo attentamente la documentazione. Il pacchetto è già dotato dello shell editor  $\text{\TeX}$ Shop che fa ricorso ad un suo previewer interno per il formato PDF; questo previewer consente di eseguire sia l'*inverse search* sia di fare la *forward search*. Questo è estremamente comodo durante la fase di editing del documento.

Merita di segnalare che per tutte e tre le piattaforme è disponibile e, talvolta, è già installato di default anche l'editor  $\text{\TeX}$ works (multipiattaforma) che, con il suo visualizzatore interno, consente di eseguire la ricerca diretta e inversa. Ad alcuni, abituati a interfacce grafiche fornite di molte barre cariche di icone per eseguire il possibile e l'impossibile,  $\text{\TeX}$ works piace poco perché la sua schermata è minimale, ma c'è un motivo: nei moderni

---

<sup>2</sup>Questo testo è stato scritto nel 2010 per installare  $\text{\TeX}$  Live su Ubuntu, che è di tipo Debian, ma ci sono anche le istruzioni per Fedora e OpenSuse; con piccole varianti sono istruzioni che vanno bene per qualunque macchina Linux.

schermi larghi, con rapporto di forma 16:9, lo schermo contiene accostate e senza sovrapposizioni sia la finestra di editing sia quella del testo composto; questo è molto comodo, più di quanto si possa immaginare, per “lavorare” agevolmente il documento da comporre. Benché anche Texmaker e TeXstudio siano in grado di accostare le due schermate, essi consumano molto spazio per le barre superiori, inferiori e laterali, cosicché le vere aree destinate all’editing o alla visualizzazione ne risultano corrispondentemente ristrette, tanto da far preferire T<sub>E</sub>Xworks sui laptop e ancor di più sui netbook.

Poi sono necessari i programmi accessori per visualizzare sullo schermo e/o stampare su carta i prodotti della composizione. Ognuna delle tre piattaforme tipo può avere già installati sia i visualizzatori dei file in formato `.dvi`, `.ps` o `.pdf`. Il formato `.dvi` è il formato nativo del sistema T<sub>E</sub>X quindi il software per visualizzare e stampare arriva insieme alla distribuzione già installata, però oggi è un formato che non si usa quasi più. Per il formato `.ps` bisogna disporre di qualcosa come ghostscript e/o GSView o altri simili software che con Linux sono solitamente già disponibili insieme al sistema. In ogni caso non è difficile trovare in rete i luoghi da dove scaricarli. Per il formato `.pdf` Linux, come al solito è già attrezzato, ma non è male per tutti e tre i tipi di piattaforma il programma Adobe Reader che la Adobe mette a disposizione di chiunque gratuitamente e per tutte le possibili piattaforme<sup>3</sup>. Naturalmente l’Adobe Reader è una specie di programma dimostrativo, per altro eccellente; ma credo che la Adobe lo metta a disposizione per far venire l’acquolina in bocca e per invogliare a comperare il prodotto commerciale completo Adobe Acrobat; per gli studenti non costa molto in accordo con il programma Education di quell’azienda. Secondo me vale ogni dollaro che costa, ma ovviamente questo giudizio dipende dall’uso che se ne fa.

Oggi, invece, è molto importante disporre di visualizzatori PDF integrati con l’editor, in modo che siano predisposti per lavorare in tandem sia per mostrare costantemente a fianco della finestra di editing la finestra del file composto in formato PDF, sia per fare la ricerca diretta e inversa; oltre ai già citati T<sub>E</sub>Xshop (solo Mac) e T<sub>E</sub>Xworks (multipiattaforma) posso citare gli editor TeXstudio e Texmaker (multipiattaforma e molto simili fra loro); altri *shell editor* non dispongono di un visualizzatore integrato, ma possono venire “sincronizzati” con visualizzatori esterni; per esempio su piattaforme Windows il visualizzatore sincronizzabile è SumatraPDF; su Linux è Okular; su Mac ne posso citare altri due: TeXnicle (gratuito) e Texpad (commerciale, ma con un costo accessibilissimo) che non solo hanno il visualizzatore integrato, ma le loro “finestre” sono in realtà due parti di una stessa finestra; ingrandendola a pieno schermo si ottiene una comodità di composizione difficilmente ottenibile con altri sistemi.

Anche Emacs (multipiattaforma), arricchito del plug-in Auctex, che lo rende particolarmente adatto per gestire i file del sistema T<sub>E</sub>X, è sincronizzabile con vari visualizzatori

---

<sup>3</sup>Sembra che con Linux le cose non stiano più così, ma in rete si trovano ancora delle installazioni non recentissime, ma funzionanti, anche per Linux.

PDF; avendo la pazienza di imparare ad usarlo in modo non superficiale, Emacs assieme ad Auctex rendono il lavoro con il sistema  $\text{\TeX}$  particolarmente comodo.

Per le piattaforme Windows il programma di installazione di MiK $\text{\TeX}$  offriva la possibilità di installare TeXnicCenter, ma lo sconsiglio vivamente, perché non è all'altezza delle distribuzioni moderne di MiK $\text{\TeX}$  e di  $\text{\TeX}$  Live; oggi mi pare che MiK $\text{\TeX}$  venga distribuito con TeXstudio. C'è anche lo *shell editor* WinEdt (shareware), ottimo e dalla versione 8 in poi sembra che sia dotato di un visualizzatore integrato PDF (comunque è sincronizzabile con SumatraPDF).

Per le piattaforme Mac esistono diversi programmi; a me sembra che il migliore di tutti sia  $\text{\TeX}$ shop, automaticamente installato quando si usa la distribuzione Mac $\text{\TeX}$ . “Migliore” significa qui il giusto compromesso fra la semplicità e l'efficienza e la validissima integrazione con un suo visualizzatore interno che consente di eseguire con un semplice click di mouse il passaggio da un punto della finestra di composizione del file sorgente al punto corrispondente nella finestra del documento composto in formato PDF, e viceversa. Suo “figlio”  $\text{\TeX}$ works (multipiattaforma) sembra avere qualche funzionalità in meno (non è così vero), ma ha una interfaccia comodissima per scoprire la codifica di un file `.tex` e per convertire il file in un'altra codifica;  $\text{\TeX}$ shop e  $\text{\TeX}$ works sono autoconfigurabili per ciascun file `.tex` grazie ad alcune righe di commenti speciali da scrivere in testa al file, cosa che rende il loro uso incredibilmente comodo. TeXstudio dalla versione 2.5 in poi è in grado di interpretare le stesse righe speciali di  $\text{\TeX}$ shop e di autoconfigurarsi di conseguenza (settembre 2012). Emacs con Auctex è in grado di usare righe di autoconfigurazione che però hanno una sintassi diversa da quelle di  $\text{\TeX}$ shop. Questa affermazione vale anche per Aquamacs, che è una applicazione per Mac che integra direttamente emacs e Auctex.

Per tutte e tre le piattaforme principali TeXStudio offre notevoli vantaggi, compresa la visualizzazione di parti del testo da comporre che richiedano più interazione fra il compositore e il software. Permette anche di disporre di una finestra laterale che contiene tutta la struttura ad albero del documento da comporre; cliccando su ogni ramo o rametto di questo albero, il programma sposta la finestra sul punto del file sorgente dove quella sezione comincia. Quasi tutti gli editor citati *non* consentono di fare la ricerca inversa con il formato di uscita PDF, ma solo con il formato DVI. TeXShop per Mac, e TeXworks, TeXStudio e TeXmaker per tutte le piattaforme permettono di eseguire nativamente la ricerca inversa anche con il formato PDF, e questa particolarità è estremamente comoda. Per le piattaforme Windows esiste il visualizzatore PDF SumatraPDF che può essere configurato per interagire con molti editor per poter essere usati assieme sia con la ricerca diretta sia con quella inversa. Va da sé, che se non ci sono esigenze diverse, il formato di uscita PDF è sicuramente quello da preferire.

Mi sono ripetuto diverse volte nel descrivere gli editor per lavorare con il sistema  $\text{\TeX}$ ? L'ho fatto apposta, a costo di essere noioso. Lavorare con editor efficienti, ben adattati al lavoro che si deve fare e che consentano la ricerca diretta e inversa fra la finestra di editing e quella del file composto in formato PDF è talmente importante, che le ripetizioni non sono mai abbastanza.

## 2.3 Ora siete pronti

Ora che avete scaricato tutto il software gratuito o commerciale di cui avete bisogno siete pronti per cominciare.

Dopo avere letto un po' di documentazione e aver giocato un poco con i programmi già predisposti sia dalla distribuzione del sistema  $\text{\TeX}$  sia dai vari editor descritti sopra, siete in grado di capire come funziona il tutto.

Attenzione: non si minimizzi la frase precedente: la documentazione va letta sempre e va capita fino in fondo; se c'è qualcosa che non si capisce, si provi con qualche piccolo esercizio, visto che la pratica permette di capire la teoria (e viceversa), ma non andate a cercare in rete aiuto nei vari forum dedicati a  $\text{\LaTeX}$  per farvi dire: “guarda nella pagina tal-dei-tali della documentazione”; per voi sarebbe una vera umiliazione... I forum vanno benissimo, ma per rispondere a domande serie, non a cose che si trovano già documentate.

A qualcuno può venire in mente: “Ma non sarà mica che ci siano in giro dei programmi che permettono di fare tutto questo in modo WYSIWYG?” Come noto, WYSIWYG è l'acronimo che si forma con le iniziali di “what you see is what you get”.  $\text{\LaTeX}$  dovrebbe essere classificato con l'acronimo WYSIWYM che sta per “what you see is what you mean”. Certo per ottenere esattamente quello che si vuole comunicare bisogna lavorare (apparentemente) di più; in realtà bisogna usare di più la testa e di meno il mouse.

Tuttavia là fuori nei negozi ci sono diversi prodotti che consentono di usare  $\text{\LaTeX}$  praticamente in modo WYSIWYG; da Scientific Word a LyX a TeXmacs a Textures ce ne è per ogni piattaforma; LyX e TeXmacs sono freeware mentre gli altri costano attorno ai 500\$. Poi c'è la soluzione gratuita di OpenOffice e di LibreOffice con l'estensione 1.2 di Writer2LaTeX. Io le sconsiglio tutte, come ho già avuto modo di dire, e qui ne ripeto i motivi.

Per poter operare in modo “sincrono” così da avere immediatamente sullo schermo una cosa molto simile a quello che si otterrà sulla carta, il programma deve essere velocissimo ad eseguire il rendering grafico di quanto viene via via immesso nel testo; per questo motivo deve rinunciare a non poche funzionalità del sistema  $\text{\TeX}$ . Però tutti questi software hanno la possibilità di salvare i file in formato `.tex`, cioè nel formato sorgente del sistema  $\text{\TeX}$ , per cui una volta finito l'editing si può eseguire la composizione finale con il programma vero, e non tramite le funzionalità del programma di editing.

Io ho cominciato a lavorare con  $\text{\LaTeX}$  a metà degli anni '80 e non ho mai usato editor sincroni. Negli anni '90 ho esaminato Textures per aiutare un collega statunitense che stava scrivendo un libro con quel software, ma non sapeva come fare per disporre di macro adatte per la composizione della matematica di cui aveva bisogno; ma dopo poco ho lasciato perdere perché dovevo lavorare su una piattaforma Mac altrui e non potevo seccarlo in continuazione per chiedergli come si fa questo, come si fa quello; allora il sistema operativo era molto diverso dall'attuale Mac OS X e solo gli addetti ai lavori sapevano come usarlo al meglio. Però non ne ero rimasto particolarmente impressionato, anche perché allora i font vettoriali venivano gestiti in modo molto più complesso di oggi.

Tabella 2.1. Le principali differenze fra pdf $\LaTeX$  e X $\LaTeX$ 

	pdf $\LaTeX$	X $\LaTeX$
Formati di uscita	DVI e PDF	XDV e PDF
Font	Font di 256 caratteri con codifiche OT1, T1, T2, LY1, LGR, eccetera	Font OpenType con codifica UNICODE
Alfabeti diversi	Solo mediante pacchetti esterni	Font OpenType
Lingue retrograde	Solo mediante pacchetti esterni	Font OpenType
Ideogrammi	Solo mediante pacchetti esterni	Font OpenType
Gestione Lingue	circa 80 lingue	circa 80 lingue
Formati immagini	PDF, JPG, PNG, EPS(†)	PDF, JPG, PNG, EPS
Possibilità di scontornare	SI	Ridotta ma può essere migliorata attraverso i comandi primitivi del motore xetex
Microgiustificazione	Completa	Solo protrusione
Formato archiviabile	PDF/A-1b	SI, ma... (*)

(\*) X $\LaTeX$  con la distribuzione 2016 di  $\TeX$  Live, consente la produzione diretta del formato PDF/A, ma richiede delle attenzioni particolari che verranno descritte nel seguito.

(†) Dalla versione del 2010 pdf $\LaTeX$  converte automaticamente in formato PDF i file EPS, conservandone quindi il carattere vettoriale; provvede anche a scontornarli.

Disponendo di una piattaforma Linux ho anche verificato il funzionamento sia di LyX sia di TeXmacs, ma alla fine sono tornato a usare i miei semplici editor ASCII con i quali non ho assolutamente nessuna limitazione per quel che riguarda la composizione dei miei testi. Bisogna dire che io faccio un grande uso di macroistruzioni adatte al mio modo di comporre; in generale le macroistruzioni personali non sono interpretabili da quei sistemi che sono un po' chiusi in se stessi, proprio per poter massimizzare la velocità di rendering.

Il difetto maggiore di questi programmi di composizione sincrona è che distraggono l'autore con l'aspetto del testo composto più o meno fedelmente a quello che si potrà ottenere davvero. Scrivere in modo WYSIWYM significa concentrarsi sul messaggio e non sul suo aspetto. Inoltre quando con quei programmi si esporta il documento in formato  $\LaTeX$ , il codice generalmente è penoso; dipende dal contenuto, ma generalmente è richiesto un pesante lavoro di pulizia e di riscrittura di alcune parti per renderle veramente scritte e marcate come si deve. Se poi bisogna fare delle modifiche, queste vanno comunque fatte sul file  $\LaTeX$ , perché in generale quei programmi non accettano macro personali o pacchetti

per i quali non siano già predisposti. In sostanza è una gran perdita di tempo e farne uso non vale assolutamente la pena, nemmeno se si è principianti e si trova comodo ricorrere a qualcosa che ricorda l'uso dei word processor a cui si è già abituati; il principiante che cominci con questi software, non imparerà mai a usare  $\text{\LaTeX}$  come si deve.

**Nel seguito partirò dal presupposto che si abbia già una certa conoscenza di  $\text{\LaTeX}$  e che si conosca la differenza fra  $\text{\LaTeX}$ ,  $\text{\pdf\LaTeX}$ ,  $\text{\Xe\LaTeX}$  e  $\text{\Lua\LaTeX}$ .**

È possibile che qualcuno non abbia conoscenze sufficientemente approfondite a proposito del programma e mark-up  $\text{\Xe\LaTeX}$ ; è un tipo di mark-up molto simile a quello di  $\text{\LaTeX}$  e di  $\text{\pdf\LaTeX}$ , ma ha una gestione dei font diversa e può usare anche i font del sistema operativo, senza dover fare nessuna acrobazia per installarli e configurarli. Esso ha bisogno di attenzioni particolari per quel che riguarda la creazione dei file in formato PDF archiviabile, anche se l'uscita finale sia in formato PDF; gli si possono dare in pasto le figure nei formati accettati sia dal programma latex (formato EPS), sia dal programma pdflatex (formati PDF, PNG, JPG). Per la gestione delle lingue dispone di un suo pacchetto *polyglossia* che è specifico per questo programma di composizione. L'uso di  $\text{\Xe\LaTeX}$  sta guadagnando terreno specialmente fra i linguisti; certamente è il motore di composizione più adatto alle tesi di carattere letterario, specialmente se contengono estesi brani composti con “lettere” non appartenenti all'alfabeto latino. Oggi  $\text{\Lua\LaTeX}$  può sostituire completamente  $\text{\Xe\LaTeX}$  anche se talvolta sembra leggermente più lento nel suo lavoro; in compenso la sua integrazione con il linguaggio di scripting Lua gli permette di fare cose impossibili con gli altri programmi citati.

Merita qui segnalare con una tabellina le principali differenze fra  $\text{\pdf\LaTeX}$  e  $\text{\Xe\LaTeX}$ , tabella 2.1 nella pagina 19.



# Capitolo 3

## L'uso di TOPtesi

La maggior parte delle macro definite nel pacchetto TOPtesi servono per comporre il frontespizio; siccome l'utente potrebbe desiderare di comporre il frontespizio in modo diverso da quello preimpostato in questa classe, prima esporrò come e perché la classe si comporta in un certo modo per comporre questa prima e importante pagina della tesi, poi esporrò che cosa bisogna fare se si usano altri pacchetti o altre tecniche per comporre il frontespizio.

Ci si ricordi comunque che l'aspetto generale dalla pagina, i font usati o gli altri font alternativi che si possono usare, il frontespizio, e altri elementi che costituiscono il “look” della tesi composta con questa classe, sono abbastanza rigidi; non dico che questa classe sia rigorosamente del tipo “prendere o lasciare”, ma sicuramente non è una classe generica da poter personalizzare a piacere in ogni dettaglio. Ci sono altre classi disponibili già distribuite con il sistema  $\text{\TeX}$  (aggiornato e completo); ne cito alcune: la classe *sapthesis*, impostata sulle prescrizioni dell'università di Roma La Sapienza, la classe *suftesi*, che in realtà non serve solo per comporre tesi, ma serve anche per comporre diversi altri tipi di documenti in diversi formati e con diversi stili. Nessuna di queste classi è configurabile a piacere, ma solo nei limiti delle personalizzazioni previste. Quella più configurabile mi pare sia la classe *suftesi*, ma anche questa, esplicitamente predisposta per tesi nel campo delle scienze umane, forse è meno flessibile per le scienze sperimentali.

### 3.1 Impostazioni standard di TOPtesi

L'insieme di macro contenute in TOPtesi realizza esattamente le specifiche di composizione per le monografie, tesi di laurea e dissertazioni di dottorato da comporre presso il Politecnico di Torino; l'acronimo TOP sta per **T**Orino **P**olitecnico e naturalmente gioca un po' anche sull'altro significato della parola inglese “top”.

Tuttavia il pacchetto non è stato creato solo per il Politecnico di Torino, il cui nome è quello preimpostato per l'Ateneo; invece la stringa “Facoltà di ” e il nome della facoltà sono vuote. Questo dipende dall'ultima riforma universitaria approvata in Italia, che

ha obbligato gli atenei a riformulare gli statuti in modo da eliminare sostanzialmente le facoltà; in alcuni casi strutture didattiche equivalenti alle facoltà sono sopravvissute sotto altro nome, in altri sono sparite definitivamente. In alcuni casi il coordinamento delle attività didattiche è passato ai dipartimenti. Lo studente deve quindi informarsi presso la sua segreteria didattica competente per sapere come debba essere intestato il frontespizio delle monografie, tesi magistrali o dissertazioni di dottorato. I comandi predisposti in questo pacchetto consentono di personalizzare la composizione del proprio lavoro finale per molti atenei e molte strutture didattiche e, oserei dire, in molte lingue.

Presso il Politecnico di Torino, l'ateneo preso come riferimento, le facoltà non esistono più; perciò il valore preimpostato per la stringa "Facoltà di " è nullo; un test per la composizione del frontespizio verifica la presenza di questa stringa nulla e omette completamente di indicare qualsiasi informazione sulla struttura didattica; in altri atenei potrebbe essere utile inserire nel file di configurazione qualcosa come:

```
\StrutturaDidattica{Dipartimento_di_}
\struttura{Ingegneria_Strutturale}
\corsodilaurea{Ingegneria_di_Ponti_e_Strade}
```

Il nome del corso di laurea potrebbe essere facoltativo se la struttura didattica competente cura la didattica di un solo corso di laurea, ma quasi sicuramente questo corso non ha lo stesso nome della struttura didattica, quindi è opportuno specificarlo anche in questi casi; se invece la segreteria competente non richiede il nome di nessuna struttura didattica, allora il corso degli studi deve essere necessariamente indicato. Si noti: per facilitare l'inserimento di queste informazioni sono stati predisposti i comandi `\StrutturaDidattica`, equivalente a `\FacoltaDi`, e `\struttura`, equivalente a `\facolta`.

### 3.1.1 Dove sono i file di TOPtesi?

La distribuzione di TOPtesi contiene molti file; ma ogni distribuzione moderna del sistema  $\text{\TeX}$  li carica tutti senza che dobbiate intervenire a mano. Tuttavia è importante sapere dove trovare questi file.

L'installazione del sistema  $\text{\TeX}$  prevede che i suoi numerosissimi file siano installati in un certo numero di strutture di cartelle, chiamate "alberi"; ogni albero ha una radice; normalmente la radice della distribuzione si chiama `texmf-dist`, ma sulla vostra macchina e con il vostro sistema operativo potrebbe avere un altro nome. Esiste anche un albero radicato nella vostra "home"; nei sistemi di tipo UNIX essa si indica simbolicamente con la tilde e la radice del vostro albero personale potrebbe chiamarsi `~/texmf`; sulle macchine Windows invece dovete cercare in `C:\Documents and settings` seguito dal vostro *user name*, oppure da `All users`, oppure da `Users` seguito dal vostro nome; magari ci sono ancora altri rami di albero da percorrere, ma poi si trova `texmf` o `localtexmf`. Nei sistemi Mac la radice del vostro albero personale è sotto la vostra `~/Library`.

L'albero personale non è mai inizialmente predisposto con l'installazione del sistema  $\text{\TeX}$ ; ve lo dovete creare voi. In esso creerete una struttura di rami identica a quella degli

alberi di sistema, magari sarà un albero più semplice e non così ramificato come gli alberi di sistema, ma i gruppi di cartelle presenti dovranno essere innestati nello stesso modo. Qui metterete i vostri file personali di classe, di stile, di definizioni, eccetera. Quando aggiornate la vostra distribuzione del sistema  $\text{\TeX}$  le cartelle degli alberi di sistema potranno essere completamente riscritte, mentre il vostro albero personale non verrà assolutamente modificato.

Tutti i file del pacchetto TOPtesi vengono caricati nell'albero della distribuzione; i file della documentazione lungo il ramo `doc`; i file sorgente, lungo il ramo `source`; i file “eseguibili” in una cartella lungo il ramo `tex`; fra questi file “eseguibili”<sup>1</sup> c'è anche il file `toptesi.cfg`. Copiate questo file nella vostra cartella di lavoro come spiegato qui di seguito, cambiategli il nome ma non l'estensione, e modificalo secondo le vostre necessità.

### 3.1.2 Il file di configurazione

È comodo, ma non è obbligatorio, disporre di un file di configurazione. Esso serve essenzialmente per contenere i comandi con i loro argomenti necessari per comporre il frontespizio della tesi. Ma se usate qualche altro pacchetto o qualche altro metodo per creare il frontespizio, il file di configurazione, anche se presente, non viene usato.

Il pacchetto TOPtesi contiene già alcuni file con configurazioni di default da usare come modello; chiunque se ne può copiare uno in un altro file mantenendo l'estensione `.cfg` e con il nome identico a quello del file principale della tesi. Questo file, se esiste, verrà letto durante l'esecuzione del programma e verranno eseguiti tutti i comandi che esso contiene; se quindi il laureando ne fa uso, deve servirsi del modello per cambiarne i dati e metterci le informazioni che ritiene utili per la sua tesi. Se non vuole servirsi del file di configurazione deve solo inserire nel suo file principale tutte le informazioni necessarie al frontespizio (e al retrofrontespizio, se lo vuole usare) prima di specificare l'ambiente `frontespizio` o il comando `\frontespizio`.

Il laureando perciò può usare un diverso file di configurazione, sempre chiamato `\mainfile.cfg`, collocato nella stessa cartella dove risiede il materiale da comporre. Per esempio, potrebbe comporre la monografia di laurea creando da qualche parte la cartella `/monografia` e in questa cartella si crea `\mainfile.cfg` nel quale scrive quello che gli è necessario; in questa cartella sistema anche i file per comporre la monografia. Quando due anni dopo compone la sua tesi di laurea magistrale, crea da qualche parte la cartella `/tesi` e ci mette dentro un altro file `\mainfile.cfg` con la sua configurazione adattata alla tesi magistrale; in questa stessa cartella mette i file relativi alla sua tesi magistrale. Se dopo tre anni prende il dottorato e si scrive la sua dissertazione dottorale, si crea da qualche parte la

---

<sup>1</sup>Chiamare “eseguibili” i file che vengono usati da  $\text{\LaTeX}$  è molto improprio: si tratta di file necessari durante l'esecuzione del programma, ma non sono quei file ad essere eseguiti nel senso informatico del termine.

cartella /dissertazione e vi mette dentro un altro file  $\langle mainfile \rangle$ .cfg che configura per la sua dissertazione; sempre in questa stessa cartella mette i file necessari per comporre la sua dissertazione. Ovviamente nei tre casi  $\langle mainfile \rangle$  è il nome del file principale che, altrettanto ovviamente, sarà diverso per ciascuna delle tre tesi.

Nota bene: il modulo *topfront* che contiene i comandi per la composizione del frontespizio oltre al comando per caricare l'eventuale file di configurazione, chiama quest'ultimo file col nome `\jobname.cfg`. La macro `\jobname` è la stessa usata da  $\LaTeX$  per conservare il nome senza estensione del main file del documento che si sta componendo; quindi se il main file si chiamasse `GiorgioRossiTesiMagistrale.tex` il file di configurazione associato a questo main file si deve chiamare obbligatoriamente `GiorgioRossiTesiMagistrale.cfg`. Siccome i sistemi operativi di tipo UNIX distinguono le lettere maiuscole dalla minuscole nei nomi dei file, se il file di configurazione si chiamasse `giorgiorossitesimagistrale.cfg` non verrebbe letto da Linux o da Mac OS X perché il nome proprio del file è scritto in tutte lettere minuscole, e non sarebbe uguale al nome proprio del main file che contiene anche delle lettere maiuscole. Per evitare problemi si consiglia di rispettare le maiuscole e le minuscole anche agli utenti dei sistemi operativi Windows, anche se questi sistemi operativi non distinguono i nomi dei file in base al fatto che contengano lettere maiuscole o minuscole.

## 3.2 Pronti? Via!

Ora siete pronti per comporre la vostra tesi o monografia o dissertazione. Ricordate solo di non giocare con quei pochi comandi di  $\LaTeX$  che permettono di fare pasticci alla WYSIWYG;  $\LaTeX$  compone da solo benissimo; al massimo, alla fine, quando tutto sarà finito e non sarà più necessario apportare correzioni, potrete anche inserire qualche spazio fine positivo o negativo per fare degli aggiustamenti di seconda o terza approssimazione. Ricordatevi però che, anche se non c'è limite al meglio (ma nemmeno al peggio!),  $\LaTeX$  lavora benissimo da solo purché non lo si disturbi con interventi non professionali; se non siete un tipografo professionista, lasciate perdere ogni modifica tipografica;  $\LaTeX$ , lo ripeto, lavora generalmente meglio di molti tipografi; è proprio il motivo per il quale Knuth ha realizzato il sistema  $\TeX$ .

## 3.3 I file accessori

Qui verranno descritti alcuni usi del modulo *topfront* e del modulo *topcoman*. Il primo serve per comporre il frontespizio; il secondo mette a disposizione alcuni comandi utili. Né l'uno né l'altro sono indispensabili, ma sono utili. Questi due pacchetti sono sempre caricati da TOPtesi, tranne in alcuni casi specificati meglio più avanti. Ma il fatto di venire o non venire caricati dipende dall'utente, cioè dalle opzioni che egli specifica alla classe *toptesi* e dai pacchetti aggiuntivi che egli vuole usare; in altre parole l'utente non ha mai bisogno di

caricare esplicitamente questi due pacchetti ma può solo esplicitamente specificare azioni che ne impediscono il caricamento.

### 3.3.1 Comporre il frontespizio con *topfront*

Il pacchetto TOPtesi contiene fra le sue parti il file *topfront* che serve solo per comporre il frontespizio; può essere usato per comporre il solo frontespizio separatamente dalla tesi, ma in questo caso, talvolta, può essere meglio ricorrere a quanto esposto nel paragrafo 3.5.

In questo paragrafo, comunque spiego come comporre il frontespizio indipendentemente dal fatto che rappresenti il frontespizio isolato della tesi o sia quello della tesi intera. Questo pacchetto in effetti, oltre a leggere l'eventuale file di configurazione, contiene solo i comandi per definire gli elementi del frontespizio e per comporlo.

Comporre il frontespizio isolatamente può tornare utile in diverse circostanze, per esempio quando si deve ancora modificare la tesi quasi ultimata, ma è necessario cercare il o i relatori per far loro firmare alcune copie del solo frontespizio.

Basta predisporre un piccolo file come questo, salvandolo con un nome a piacere, per esempio, con grande fantasia, `myfile.tex`:

```
% !TEX TS-program = XeLaTeX
% !TEX encoding = UTF-8 Unicode
\documentclass[12pt]{toptesi}
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% Impostazioni per comporre con XeLaTeX
\setmainfont[Ligatures=TeX]{TeX Gyre Termes}% o altro font
\setotherlanguage{french}% o altra lingua
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
```

*<il resto del preambolo>*

```
\begin{document}

\begin{frontespizio*}
\ateneo{Università di Marconia}
\logosede{logouno}
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%% Tesi magistrale
\corsodilaurea{delle Telecomunicazioni}
\titolo{Titolo della tesi\\ di laurea magistrale}
\sottotitolo{Sottotitolo della tesi di laurea magistrale}
\relatore{prof.\ Enrico Rosa}
\sedutadilaurea{Dicembre 2114}
\candidata{Susanna Rossi}
\secondacandidata{Laura Bruni}
```

```
%\retrofrontespizio{...}  
\end{frontespizio*}
```

*⟨il resto della tesi⟩*

```
\end{document}
```

La parte che ho indicato con *⟨il resto del preambolo⟩* e *⟨il resto della tesi⟩* sono facoltative; certo sono necessarie quando si compone l'intera tesi; per comporre il solo frontespizio si possono omettere.

Perciò bisogna fare attenzione: se si compone il frontespizio separatamente dalla tesi, la riga `\documentclass` deve essere identica alla corrispondente riga del vostro main file della tesi; siccome *toptesi* può usare diversi formati di carta, è necessario che il frontespizio sia composto su carta dello stesso formato. Non è invece necessario usare come font normale un font dello stesso corpo della tesi; con carte particolarmente piccole potrebbe essere più opportuno usare il corpo di 10 pt invece che quello di 12 pt. Se si deve scrivere in un'altra lingua diversa dall'italiano e dall'inglese lo si specifica con `\setotherlanguage` con le modalità che verranno espone più avanti. Si noti che l'esempio precedente è impostato per comporre il frontespizio con  $\text{\XeTeX}$ , ma se si lavora con  $\text{\pdfTeX}$ , allora vanno specificati i font da usare, e le codifiche di entrata e di uscita; bisogna anche impostare la lingua nella maniera specifica di *babel*. Non scendo nei dettagli, perché fra i vari modelli forniti con il pacchetto TOPtesi, ce ne sono alcuni predisposti per lavorare con  $\text{\pdfTeX}$  e altri con  $\text{\XeTeX}$  o  $\text{\LuaTeX}$ .

Comunque, se si vuole usare `pdflatex`, la parte fra le prime due righe `%%` va sostituita, per esempio, con:

```
\usepackage[utf8]{inputenc}  
\usepackage[T1]{fontenc}  
\usepackage{newtxtext}% o altro font
```

Ma se si deve scrivere il frontespizio in un'altra lingua bisogna agire come spiegato poco più avanti.

In realtà il pacchetto *topfront* lavora bene in italiano e distingue i singolari dai plurali, i maschili dai femminili; non fa nulla di particolare se la lingua di composizione del frontespizio è l'inglese o un'altra lingua; ricordate che l'inglese è già impostato ma non attivato; lo si attiva nel main document con il comando `\english`. Se si vuole comporre il frontespizio in un'altra lingua, allora, usando `xelatex`, la si specifica mediante il comando `\setotherlanguage`; invece usando  $\text{\pdfTeX}$  la si indica fra le opzioni di `\documentclass` e poi, prima di usare l'ambiente *frontespizio* o `\frontespizio*` o i comandi corrispondenti, la si imposta con `\selectlanguage`. Poi si inseriscono i seguenti comandi, modificando gli argomenti in accordo con le restrizioni burocratiche per i frontespizi delle tesi nella vostra università:

```
\retrofrontespizio{This work is subject to the Creative Commons Licence}  
\DottoratoIn{PhD Course in\space}  
\NomeMonografia{Bachelor Degree Thesis}  
\TesiDiLaurea{Master Degree Thesis}  
\NomeDissertazione{PhD Dissertation}  
\InName{in}  
\CandidateName{Candidate}% or Candidates  
\AdvisorName{Supervisor}% or Supervisors  
\TutorName{Tutor}  
\NomeTutoreAziendale{Internship Tutor}  
\CycleName{cycle}  
\NomePrimoTomo{First volume}  
\NomeSecondoTomo{Second Volume}  
\NomeTerzoTomo{Third Volume}  
\NomeQuartoTomo{Fourth Volume}  
\logosede{logouno,logodue}% one logo or a comma separated list of logos
```

Passando uno di questi file a xelatex o a pdflatex, si ottiene un documento con il frontespizio e, facoltativamente, il retrofrontespizio, che si possono stampare in varie copie per gli scopi detti sopra. Ricordate che *toptesi* precarica *babel* o *polyglossia* a seconda che usiate pdflatex oppure xelatex o lualatex; non avete quindi l'obbligo di invocare di nuovo questi pacchetti se componete il solo frontespizio usando *toptesi*, ma dovete specificare la lingua che volete usare nei modi descritti sopra.

Il retrofrontespizio non è necessario in senso assoluto quando volete produrre il frontespizio isolato. Non è nemmeno necessario quando volete comporre la tesi. Tuttavia il comando è disponibile e permette di aggiungere la dichiarazione legale in merito alla licenza; il retrofrontespizio di questa documentazione è stato composto con i comandi seguenti<sup>2</sup>:

```
\retrofrontespizio{Questo testo è libero secondo le condizioni  
stabilite dalla \LaTeX\ Project Public Licence (LPPL) riportata  
nell'appendice~\ref{ch:LPPL} alla pagina~\pageref{ch:LPPL}.
```

```
\bigskip
```

```
\noindent Composto con \XeLaTeX\ il \today  
\vspace*{5\baselineskip}}
```

Non dovete dimenticare che, se volete usare xelatex o lualatex, l'editor con cui scrivete i vostri file deve essere configurato per registrarli sul disco fisso con la codifica UTF-8.

---

<sup>2</sup>Questo codice contiene dei comandi `\ref` perché è servito per comporre l'intero documento dove le etichette invocate sono definite. Se si vuole usare questo codice per comporre il retrofrontespizio assieme al frontespizio, ma isolatamente dalla tesi (lo sconsiglio), bisogna sostituire quei comandi e i loro argomenti con i valori reali.

Nell'esempio di frontespizio presentato sopra si è usato l'ambiente *frontespizio\** che è disponibile a partire dalla versione del pacchetto TOPtesi 5.85. I comandi `\frontespizio` e `\frontespizio*` funzionano ancora, ma suggerisco di usare gli ambienti.

Vediamo le differenze: nella figura 3.1 si vedono sempre i loghi all'inizio della pagina, non ci sono filetti di nessun genere, il nome dell'università ci può essere o può mancare, le diverse tipologie di tesi hanno informazioni diverse. I loghi sono tutti in testa alla pagina ma potrebbero essere anche nella parte inferiore; la differenza dipende dall'usare l'ambiente *frontespizio* che mette i loghi in testa, o l'ambiente *frontespizio\** che mette i loghi nella parte inferiore della pagina; la figura 4.1 nella pagina 80, infatti, mostra i frontespizi di sinistra che sono stati composti con l'ambiente *frontespizio* e quelli di destra composti con l'ambiente *frontespizio\**. Il nome dell'ateneo appare o non appare a seconda che il comando `\ateneo` contenga un argomento vuoto o un argomento con un nome esplicito. Le altre informazioni dipendono da quali comandi si sono usati per caratterizzare la tesi triennale, magistrale o dottorale. Questi comandi verranno descritti dettagliatamente più avanti.

Nella figura 3.3 sono mostrati invece i quattro frontespizi che si ottengono quando alla classe (o anche al solo modulo *topfront* quando lo si usa con una classe diversa da *toptesi*) viene specificata l'opzione *classica*. Si nota che il nome dell'ateneo è separato dal resto della pagina da un filetto orizzontale; analogamente l'anno accademico in calce alla pagina è separato da un filetto orizzontale. I candidati sono chiamati "Laureandi". Il blocco contenente i nomi dei relatori e correlatori e quello contenente i nomi dei laureandi sono allineati superiormente e non sono sfasati come nello stile standard. La seduta di esame è indicata con la dicitura "Anno accademico" in maiuscoletto e l'anno, o l'intervallo di anni è indicato con le cifre arabe minuscole (old style). Con questo stile più classico, il logo o i loghi sono collocati fra i titoli e i blocchi dei relatori e dei laureandi.

Si noti infine che in entrambi gli stili esistono esempi con due loghi, per i quali i nomi dei due atenei vanno scritti in forma un poco ellittica, ma piuttosto antiestetica; in generale potrebbero formare una riga così lunga da non entrare nella pagina fisica.

Lo si potrebbe considerare una "feature", una particolarità del pacchetto TOPtesi. In realtà non conosco altre classi o moduli di estensione dove sia possibile fare riferimento a diversi loghi e a diversi nomi.

Quello che consiglierei in questi casi sarebbe di aggiungere il nome dell'ateneo al logo, se già non lo contenesse, e lo farei un con un font di contrasto, per esempio con un font senza grazie maiuscolo o maiuscoletto con iniziali maiuscole, magari su due righe collocato sotto il logo vero che non contenga a sua volta già il nome dell'ateneo; come per esempio nel logo modificato mostrato nella figura 3.2.

Fatta questa modifica per tutti i loghi privi del nome dell'università che è necessario usare, si compone il frontespizio con lo stile standard usando semplicemente un nome vuoto come argomento di `\ateneo`. Si veda poco più avanti come usare una pluralità di loghi.

Vale la pena di ricordare che la retrocompatibilità con i comandi `\frontespizio` e





Figura 3.1. Frontespizi composti con lo stile standard. In questo stile mancano i filetti e i loghi delle università possono essere messi in posti diversi della pagina; il nome dell'università può essere presente esplicitamente oppure può essere compreso nel logo.



Figura 3.2. Un logo modificato con l'aggiunta del nome dell'ateneo

`\frontespizio*` è conservata.

Perciò merita ricordare la molteplicità di configurazioni offerta dal modulo *topfront* mediante la tabella comparativa 3.1.

Tabella 3.1. Varie modalità di composizione del frontespizio

Comando o ambiente	Stile standard	Stile classica
<i>frontespizio</i>	loghi in testa	loghi in basso
<code>\ateneo vuoto</code>	senza nome dell'ateneo	messaggio d'errore
<code>\ateneo non vuoto</code>	Nome dell'ateneo in testa	nome dell'ateneo in testa
filetti	assenti	presenti
<i>frontespizio*</i>	loghi in basso	loghi in basso
<code>\ateneo vuoto</code>	senza nome dell'ateneo	messaggio d'errore
<code>\ateneo non vuoto</code>	nome dell'ateneo in testa	nome dell'ateneo in testa
filetti	assenti	presenti
<code>\frontespizio</code>	come <i>frontespizio</i>	come <i>frontespizio</i>
<code>\frontespizio*</code>	come <i>frontespizio*</i>	come <i>frontespizio*</i>

### 3.3.2 I comandi accessori di topcoman

L'altro file accessorio *topcoman* contiene diversi utili comandi, alcuni già incorporati nella definizione della lingua italiana del sistema T<sub>E</sub>X (solo se si usa *babel*). Tuttavia se dovete scrivere in una lingua diversa dall'italiano quei comandi non sono disponibili e qui viene utile avere quei comandi in un pacchetto distinto da *babel* e non soggetto a test sulla lingua in uso. Alcuni di questi comandi non definiti da *poliglossia* quando si scrive in italiano, si rendono particolarmente utili quando si compone con xelatex o lualatex.



Figura 3.3. Frontespizi composti con lo stile classica. Con questo stile si ha una disposizione diversa delle varie informazioni e i loghi sono sempre posti nella parte inferiore della pagina. Il nome dell'università è sempre presente.

Il file contiene anche un comando per stampare in modo verbatim il contenuto di un file esterno, per esempio il listato di un programma; ci sono altri pacchetti  $\text{\LaTeX}$  che consentono di stampare il listato di un programma in modo anche migliore, per esempio il pacchetto *listings*, ma, visto che può servire per la tesi e, visto che una volta che lo si conosce potrebbe piacere e poi si vorrebbe continuare ad usarlo, è meglio disporne come pacchetto separato. Non sarebbe difficile per esempio stamparsi in pulito il listato del proprio programma contenuto nel file `myprogram.c`: si scrive semplicemente un file del tipo

```
\documentclass{report}
\usepackage{topcoman}
\begin{document}
\pagestyle{plain}
\chapter*{Il programma myprogram.c}
\listing{myprogram.c}
\end{document}
```

e lo si passa a `pdflatex` per ottenere poi una stampa ben composta da conservare per documentazione; per evitare che le righe fuoriescano dalla larghezza normale del testo, o, peggio ancora, fuori dalla carta, sarebbe opportuno che le righe del programma non fossero più lunghe di una ottantina di caratteri per la carta A4, proporzionalmente di meno per formati di carta più piccoli.

## 3.4 La composizione del frontespizio

Non è generalmente necessario inserire nel frontespizio il logo dell'ateneo; controllate presso la vostra segreteria competente le regole fissate per la composizione del frontespizio. Se il logo è richiesto, allora la segreteria vi dice anche quale logo usare e da dove scaricarlo il file che ne contiene la descrizione grafica; **siate precisi nel richiedere le limitazioni legali d'uso del logo dell'ateneo, perché il logo è proprietà esclusiva dell'ateneo e l'uso indebito potrebbe essere fonte di problemi legali non indifferenti**. Siccome tali limitazioni esistono, il pacchetto TOPtesi non viene più distribuito con i file dei loghi di diversi atenei, come veniva fatto nelle prime edizioni.

Controllate se si tratta di un file vettoriale (estensione `.pdf` oppure `.eps`) oppure bitmapped (estensione `.jpg`, `.png`, `.bmp`, `.wmf`, `.tiff`,...). Ricordate che `pdflatex`, `xelatex` e `lualatex` possono incorporare solo i formati `.pdf`, `.eps`<sup>3</sup>, `.jpg`, `.png` o `.mps`<sup>4</sup>. Se il file

---

<sup>3</sup>Dall'edizione di  $\text{\TeX}$ Live 2010 anche `pdflatex` è in grado di incorporare file in formato `.eps`, mentre prima era necessario convertirli preventivamente in formato `.pdf`.

<sup>4</sup>I file vettoriali in formato `.mps` sono ottenuti con il programma METAPOST; l'estensione normale delle figure create con questo programma di default hanno una estensione numerica; si può impostare il programma per assegnare l'estensione `.mps` oppure si può cambiare il nome e l'estensione.

bitmapped viene fornito in un formato diverso da questi cinque, è necessario convertirlo in uno dei formati accettati.

Vale la pena di ricordare che il formato .jpg va bene per immagini a colori sfumate come le fotografie, mentre il formato .png è più adatto per disegni al tratto con righe fortemente contrastate rispetto allo sfondo. In generale, quindi, sarebbe meglio convertire nel formato .png un logo bitmapped fornito in un formato diverso da quelli elencati sopra, ma il formato .png potrebbe contenere delle “ trasparenze ” che non sono accettabili con le prescrizioni PDF/A dei documenti archiviabili e in questo caso bisogna ricorrere al formato .jpg. Potendo disporne, sarebbe sempre meglio usare un disegno veramente vettoriale del logo.

Merita segnalare che le tesi svolte nell’ambito di collaborazioni fra atenei possono richiedere i loghi dei diversi atenei partecipanti all’accordo; segnalo la Scuola Interpolitcnica di Dottorato condotta nell’ambito di un accordo fra i politecnici di Bari, Milano e Torino. Ricordo i programmi di doppia laurea condotti nell’ambito di accordi fra l’ateneo di appartenenza e un altro ateneo, generalmente europeo. Ricordo il programma europeo Erasmus Mundus dove il titolo di studio viene rilasciato da consorzi di almeno quattro atenei di nazioni europee diverse. In queste circostanze è spesso necessario inserire nel frontespizio i loghi di tutti gli atenei partecipanti alla convenzione.

### 3.4.1 Il logo dell’ateneo per l’uso con TOPtesi e TOPfront

Il pacchetto *topfront* contiene un comando molto potente, `\logosedede`, che accetta la sintassi seguente:

```
\logosedede[⟨altezza⟩]{⟨file del logo⟩}
```

oppure

```
\logosedede[⟨altezza⟩]{⟨lista di nomi di file di loghi separati da virgole⟩}
```

dove il valore prefissato per l’`⟨altezza⟩` del logo è 25 mm per la carta A4, ed è scalato proporzionalmente per carte di formato diverso; con l’argomento facoltativo si può specificare un’altezza leggermente maggiore o minore del valore predefinito. L’argomento `⟨file del logo⟩` o `⟨lista di nomi di file di loghi separati da virgole⟩` è costituito da un solo nome di file o da una lista di nomi di file separati da virgole, corrispondenti ciascuno al logo di un diverso ateneo. Questi file devono avere i formati corrispondenti a una delle cinque estensioni consentite, ma non è necessario specificare il nome del file completo di estensione; questa specificazione potrebbe essere necessaria solo nel caso in cui si disponga del logo di uno stesso ateneo in diversi formati grafici (o, cosa da non fare mai, immagini o loghi diversi con il nome del loro file uguale e l’estensione diversa; questo genere di pasticci è fonte di numerosi errori, facilmente evitabili se si mantiene in ordine la scelta dei nomi e delle estensioni).

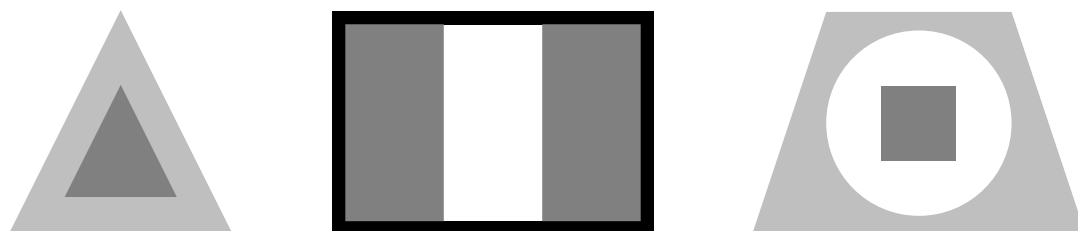
Il programma codificato nel pacchetto *toptesi* provvede a scalare tutti i loghi all'⟨*altezza*⟩ specificata e li compone affiancati su una sola riga distanziati mediante uno spazio predefinito di 3 em; lo si può impostare ad un valore leggermente maggiore o minore tramite il comando `\setlogodistance`, per esempio:

```
\setlogodistance{2em}
```

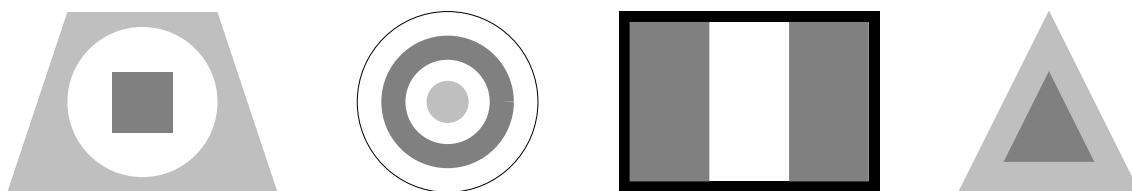
Se la riga di loghi, eventualmente scalati in altezza e distanziati dello spazio predefinito o esplicitamente specificato, risultasse più larga della giustezza della pagina del titolo, il programma provvede ad uno scalamento dell'intera riga di loghi in modo che non superi la giustezza della pagina. Un solo logo viene composto centrato; due o più loghi vengono ugualmente composti centrati ma distanziati come si è detto. Seguono alcuni esempi, ma si tenga presente che i loghi usati per questi esempi sono solo dei disegni creati in modo tale da avere la forma di un logo; loghi veri di diverse università non sono distribuiti insieme a TOPtesi, a causa delle restrizioni sulla loro proprietà legale; solo il vero logo del Politecnico di Torino appare nel frontespizio, perché questa è una pubblicazione ufficiale di quell'ateneo. Un solo logo appare come nel frontespizio di questo manuale; due loghi appaiono così:



Tre loghi appaiono così:



Quattro loghi, nonostante sia stata conservata l'⟨*altezza*⟩ predefinita, producono una riga di loghi di altezza ridotta per consentire che appaiano tutti e quattro nella stessa riga:



Si noti che il logo in testa alla pagina del frontespizio di questo documento è una esplicita richiesta espressa dal Politecnico di Torino; molte altre sedi come la quasi totalità delle case editrici che mettono il loro logo nel frontespizio, lo mettono nella metà inferiore della

pagina; ma la tesi di laurea o di dottorato è un documento particolare, il cui frontespizio deve soddisfare ad alcune esigenze burocratiche, e non può, quindi, soddisfare a criteri estetici molto stringenti. Per questo motivo, lo si vedrà meglio avanti, il modulo *topfront* consente una variante del comando per creare i frontespizi: essa intesta la pagina col nome dell'ateneo e mette i loghi nella metà inferiore della pagina. Se, come per il Politecnico di Torino, il nome fa parte del logo<sup>5</sup> non è il caso di ripetere il nome dell'ateneo nella parte alta della pagina perché vi appare già nel logotipo. Comunque, volendo, si può inserire in testa alla pagina sia il logo, sia il nome generico dell'ateneo sia il nome proprio dell'ateneo; come fare lo si è già descritto in precedenza e lo si è riassunto nella tabella 3.1 nella pagina 30.

### 3.5 Comporre il frontespizio senza ricorrere a TOPfront

Potrebbe essere necessario comporre il frontespizio in modo diverso da come viene composto tramite i comandi contenuti nel pacchetto *topfront*, sia pure ricorrendo alle due varianti dell'ambiente *frontespizio* e *frontespizio\**.

Esistono alcune strade: una è quella di usare l'ambiente *titlepage* e scriverci dentro quello che si vuole in accordo con le prescrizioni del proprio ateneo; l'altra è quella di usare altri pacchetti espressamente creati per comporre frontespizi, in particolare il pacchetto *frontespizio*.

Ci sono diversi motivi per i quali un laureando o dottorando vorrebbe comporre un frontespizio diverso da quello predefinito da TOPtesi con il suo modulo *topfront*.

1. Le specifiche per le tesi di una data università sono inconciliabili con quelle preimpostate da TOPtesi.
2. Le informazioni da inserire nel frontespizio sono diverse da quelle preimpostate da TOPtesi.
3. Il laureando non deve soddisfare a specifiche particolari ma vuole comporre un frontespizio personalizzato a suo piacimento.

A seconda dei motivi per i quali il laureando non vuole o non può usare il layout e la strutturazione del frontespizio predisposto da TOPtesi, le due strade offrono alcuni vantaggi e svantaggi.

La prima via, quella di usare l'ambiente *titlepage*, permette il massimo della libertà e consente di comporre il frontespizio come il laureando lo vuole, ma ha lo svantaggio che il laureando dentro quell'ambiente deve specificare ogni dettaglio di composizione.

---

<sup>5</sup> Il logo di questo Politecnico è fatto di due parti: quella tonda di sinistra prende il nome di “marchio”, mentre quella letterale a destra prende il nome di “logotipo”; l'Ufficio Comunicazione e Immagine che si occupa di questi aspetti formali ha emesso una normativa specifica sui colori e le forme sia del marchio sia dei font usati per il logotipo.

Se conosce bene L<sup>A</sup>T<sub>E</sub>X e ha buone conoscenze tipografiche può ottenere un risultato gradevole, ma può anche cadere negli errori tipografici più comuni e produrre un frontespizio impresentabile.

La seconda via può dare risultati buoni, grazie alla possibilità di personalizzazione che offre, in particolare con il pacchetto *frontespizio*. Ma non si può fare sempre tutto; anche questo pacchetto è preimpostato per produrre il frontespizio nel formato di carta A4 e con due stili, “standard” ed “elements” specifico per l’uso con la classe *suftesi*. Più avanti indicherò come produrre frontespizi con altri formati di carta.

### 3.5.1 Comporre il frontespizio con l’ambiente *titlepage*

Per seguire questa strada, qui posso dare solo delle indicazioni generiche.

1. Aperto l’ambiente *titlepage* sarebbe opportuno cambiare localmente la geometria del layout della pagina, specificandolo con

```
\thispagestyle{<stile>}
```

dove *<stile>* è stato precedentemente definito usando il pacchetto *geometry* o altri pacchetti. Sarebbe opportuno ridefinire i due margini interno ed esterno in modo che siano uguali. Qualcuno desidera apportare un piccolo spostamento verso l’esterno di non più di 5 mm per compensare la curvatura della prima pagina; questa curvatura dipenderebbe dal modo di confezionare/legare la tesi; di solito con una buona brossura ben eseguita o una legatura a filo, non è necessario apportare questa correzione, con una tesi spillata con punti metallici, probabilmente una piccola correzione non guasta; questa correzione è concepita per le pagine interne ma, siccome il frontespizio è la prima pagina della tesi, la curvatura della pagina è irrilevante e questo piccolo spostamento è del tutto superfluo.

2. Non è necessario che i margini laterali siano complessivamente equivalenti ai margini del corpo della tesi, ma non guasta se lo sono.
3. Per la disposizione verticale del testo, sarebbe preferibile che i margini superiore e inferiore siano otticamente equivalenti; questo dipende dal fatto che il primo elemento in alto sia, per esempio, il nome dell’ateneo scritto con caratteri relativamente alti e neri, oppure sia il logo dell’università, che può essere alto, ma di solito non è tanto largo. Inoltre lo stile della pagina non deve presentare né testatina con il suo spazio di separazione del testo, né piedino, con il suo spazio di separazione dal testo, e non deve contenere il numero della pagina.
4. Il laureando deve provvedere autonomamente alle spaziature verticali fra un elemento e l’altro, deve decidere se certe parti siano da comporre in modo centrato, oppure in bandiera col palo a sinistra o a destra, se certe parti debbano essere affiancate e come debbano essere distanziate.
5. Il laureando deve naturalmente scegliere collezione, codifica, famiglia, serie, forma, corpo e scartamento dei font che usa per le varie parti del testo. Una possibile scelta



per un font senza grazie da usare per il nome dell’ateneo, potrebbe essere quella di riferirsi alla collezione Iwona, scegliendo la famiglia *iwona* con codifica T1 nella serie m con la forma n, nel corpo di 15 pt, e con lo scartamento poco interlineato di soli 17 pt. Dovrebbe quindi chiamare il pacchetto *iwona* nel preambolo e per usare il font dovrebbe usare la serie di comandi

```
\fontsize{15}{17}%  
\usefont{T1}{iwona}{m}{n}
```

quando compone quell’informazione; dovrebbe curare di fare questa operazione dentro un gruppo al fine di mantenere queste impostazioni solo per la scrittura di quella informazione. Sarebbe meglio definirsi un comando personale per evitare di dover ogni volta introdurre quella serie di istruzioni, con il rischio di commettere anche il più piccolo errore di battitura e quindi di generare fastidiosi messaggi d’errore.

Non scendo in ulteriori dettagli; l’utente che volesse comporre un frontespizio molto personalizzato per questa via, deve sapere quello che sta facendo e deve farlo con la dovuta competenza tipografica e del linguaggio L<sup>A</sup>T<sub>E</sub>X. I risultati estetici dipendono fortemente da queste sue capacità.

Un esempio di un frontespizio costruito per questa via è quello che un utente ha voluto creare con la pagina in landscape; la sua tesi era veramente originale nel formato “panoramico” dovuta al fatto che conteneva molti disegni. Egli ha poi scritto questo codice:

```
\documentclass [titlepage]{article}  
\usepackage[a4paper,left=3cm,bottom=1.5cm,right=3cm,top=1.5cm,%  
landscape]{geometry}  
\usepackage{graphicx}  
\usepackage[T1]{fontenc}  
\usepackage{iwona}  
\usepackage[italian]{babel}  
\usepackage[utf8]{inputenc}  
\nofiles  
  
\newcommand {\institutionfont}{\fontsize {22}{30}\scshape}  
\newcommand {\divisionfont}{\fontsize {16}{20}\rmfamily}  
\newcommand {\pretitelfont}{\fontsize {16}{16}\rmfamily}  
\newcommand {\titlefont}{\fontsize {18}{22}\usefont{T1}%  
iwona}{bx}{n}}  
\newcommand {\fixednamesfont}{\fontsize {14}{17}\mdseries}  
\newcommand {\namesfont}{\fontsize {15}{18}\bfseries}  
\newcommand {\footfont}{\fontsize {15}{18}\rmfamily}
```

```

\begin{document}

\begin{titlepage}

\centering

\begin{minipage}[c]{.475\textwidth}\centering
\includegraphics[width=30mm]{logouno}\\[\baselineskip]
{\institutionfont UNIVERSITÀ DI PAPEROPOLI\par}
\end{minipage}
\hfill
\begin{minipage}[c]{0.475\textwidth}\centering
{\divisionfont FACOLTÀ DI PENNUTOLOGIA\\[1ex]
Corso di~laurea in~ingegneria del~volo~battente\par}
\end{minipage}\vspace{\stretch{1}}

{\pretitelfont Tesi di laurea\par}\vspace{\baselineskip}
{\titlefont STUDIO ANALITICO DELLE PROPRIETÀ DELLE PENNE,
INCLUDE QUELLE DEGLI UCCELLI ACQUATICI, IN PARTICOLARE QUELLE
DEI~CIGNI NERI AUSTRALIANI\par}
\vspace{\stretch{0.7}}

\makebox[\textwidth]{\null\hfill\def\arraystretch{1.5}%
\begin{minipage}[t]{.375\textwidth}\raggedright
\begin{tabular}[t]{@{}l@{}}
\fixednamesfont Relatore\\
\namesfont Prof. ing. Mario Rossi\\[1ex]
\fixednamesfont Correlatore\\
\namesfont Dott. ing. Giuseppe Scapigliati
\end{tabular}
\end{minipage}
\hfill
\begin{minipage}[t]{.375\textwidth}\raggedleft
\begin{tabular}[t]{@{}l@{}}
\fixednamesfont Candidato\\
\namesfont Alfredo Bianchi
\end{tabular}
\end{minipage}\hfill\null}

\vspace{\stretch{1}}

```

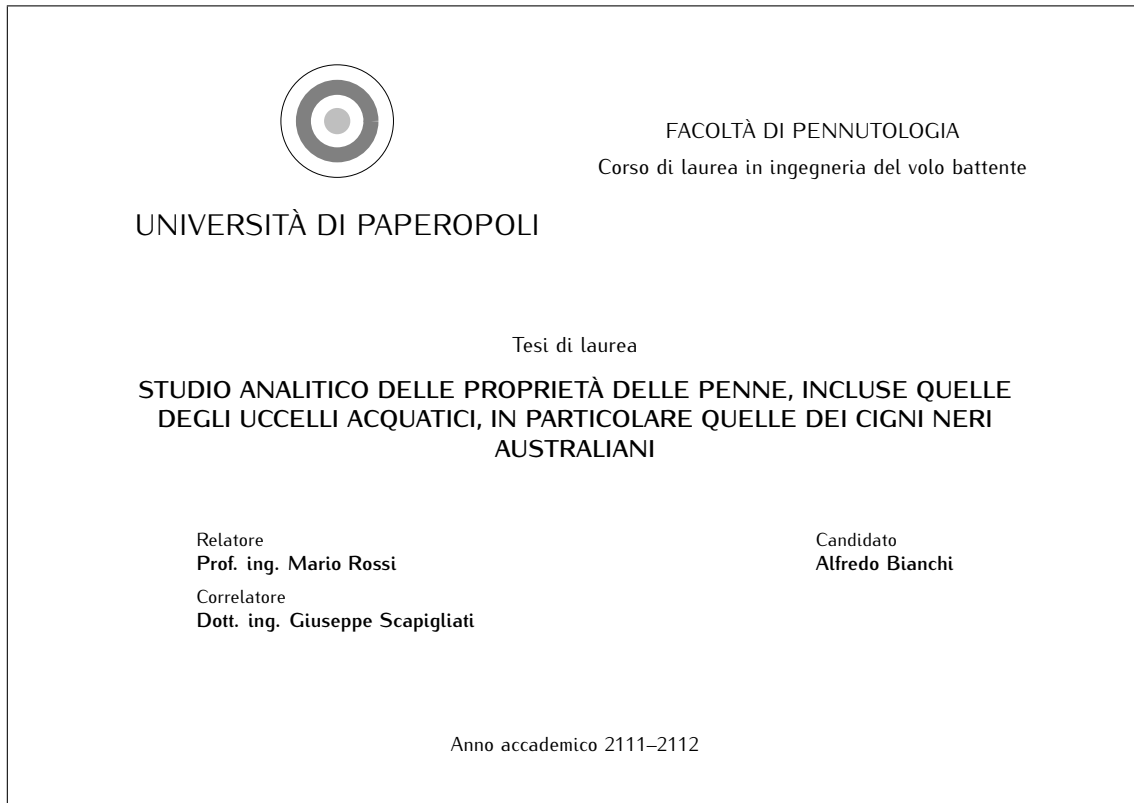


Figura 3.4. Frontespizio con l’orientamento landscape ottenuto con l’ambiente *titlepage*

```
{\footfont Anno accademico 2111--2112\par}  
  
\end{titlepage}  
\end{document}
```

Nel codice precedente appaiono alcuni comandi per i font i cui nomi per le corrispondenti dichiarazioni sono ispirati a quelli contenuti nel pacchetto *frontespizio*, ma i nomi sono irrilevanti, nel senso che qualunque altro nome sarebbe stato altrettanto valido; è invece utile che i nomi dei font siano scelti in modo che ricordino la funzione per la quale sono stati definiti.

La pagina che l’utente ha ottenuto è contenuta nella figura 3.4. Per inserirla come frontespizio della tesi egli ha poi usato il pacchetto *pdfpages* che permette di inserire pagine scelte di un file PDF in un documento da comporre con L<sup>A</sup>T<sub>E</sub>X. Se ne veda la documentazione aprendo un terminale o un prompt dei comandi, scrivendoci dentro `texdoc pdfpages` e poi premendo il tasto `invio`.

### 3.5.2 Comporre il frontespizio con il pacchetto *frontespizio*

Esistono altri pacchetti, oltre a *frontespizio*, per comporre pagine di titoli; ma per le tesi questo pacchetto sembra il più indicato. Definisce comandi per inserire virtualmente qualsiasi informazione necessaria per un frontespizio di testi di laurea triennale, magistrale o dottorale; per impostazione predefinita tutte le informazioni fisse sono in italiano, ma ciascuna può essere modificata come si vuole in qualsiasi lingua.

Fra le opzioni della classe *toptesi* si deve inserire l'opzione `noTOPfront` per evitare conflitti fra i comandi del modulo *topfront* e quelli del pacchetto *frontespizio*. **Dimenticarsi di specificare questa opzione vuol dire andare incontro a messaggi d'errore causati dai conflitti fra i due pacchetti.**

Questo pacchetto *frontespizio* si avvale di un file ausiliario `-frn.tex` per comporre il frontespizio vero e proprio; questo implica che per disporne come frontespizio del documento finito, bisogna comporre anche questo file ausiliario.

Precisamente tutte le informazioni riguardanti il frontespizio devono essere messe dentro un ambiente *frontespizio* collocato dopo l'istruzione `\begin{document}`. Qui si metteranno i comandi per definire il nome del o dei candidati, il nome del o dei relatori e degli eventuali correlatori, il nome dell'università, il nome del corso di laurea, il nome del tipo di tesi, il titolo e il sottotitolo della tesi, eccetera.

Eseguendo `pdflatex` oppure `xelatex` o `lualatex` viene scritto il file ausiliario; se la tesi ha il main file che si chiama `MarioRossiTesiMagistrale.tex`, dopo averlo compilato, nella cartella di lavoro ci sarà anche il file ausiliario `MarioRossiTesiMagistrale-frn.tex`; bisognerà dunque compilare questo file ausiliario, e poi ricompilare il documento che *riscriverà il file ausiliario* con i nuovi eventuali dati modificati, ma intanto ingloberà il file PDF appena prodotto. La sequenza delle compilazioni sarà dunque:

```
pdflatex MarioRossiTesiMagistrale
pdflatex MarioRossiTesiMagistrale-frn
pdflatex MarioRossiTesiMagistrale
```

Se il frontespizio deve essere composto con font diversi da quelli di default, con codifiche particolari, con altre impostazioni particolari è meglio che il file ausiliario sia dotato di un preambolo adeguato; quindi il file principale avrà la struttura seguente;

```
\documentclass[<opzioni-classe>]{toptesi}
% Preambolo del main file
...
\usepackage[<codifiche>]{inputenc}
...
\usepackage[<opzioni-frontespizio>]{frontespizio}
...
\begin{document}
% Per il preambolo del file ausiliario
```

```
\begin{Preambolo*}
  \usepackage{<opzioni>}{<pacchetto>}
  \newcommand...
  \renewcommand...
  ...
  \renewcommand...
\end{Preambolo*}
% Informazioni per il frontespizio
\begin{frontespizio}
  \Universita{<nome corto dell'università>}
  \Logo[<altezza-logo>]{<logo>}
  \facolta{<nome della facoltà>}
  ...
  \NCorrelatore{<singolare>}{<plurale>}
  \Correlatore{<Nome-Cognome di un correlatore>}
\end{frontespizio}
% Corpo del main file
...
\end{document}
```

Il contenuto dell'ambiente *Preambolo\** può essere molto vario e dipende dal programma usato per la compilazione del frontespizio, da scelte particolari dei font, da comandi da ridefinire, eccetera. La documentazione di *frontespizio* espone diversi esempi che danno un'ottima visione d'insieme per tutte le operazioni che si possono fare e dei risultati che si possono ottenere.

Però, come già detto, questo pacchetto compone di default il frontespizio in una pagina A4, e in due stili, quello “standard” e quello per lo stile “elements” della classe *suftesi*.

Se si volesse comporre la tesi in formato B5, per esempio, bisogna sfruttare adeguatamente il contenuto dell'ambiente *Preambolo\**.

La procedura da seguire, dunque dovrebbe essere modificata come segue. Assumiamo che non si debbano apportare modifiche al contenuto del frontespizio; in caso contrario bisogna ripetere la procedura dopo ogni modifica del suo contenuto all'interno dell'ambiente *frontespizio*.

```
\documentclass [11pt,b5paper]{book}
% Preambolo del documento
\usepackage[norules,swapnames]{frontespizio}
\usepackage[T1]{fontenc}
\usepackage{iwona}
\usepackage[italian]{babel}
\usepackage[utf8]{inputenc}
\nofiles
\begin{document}
```

```
% Per il preambolo del file ausiliario
\begin{Preambolo*}
  \usepackage[11pt,b5paper,top=25mm, bottom=25mm,
             left=15mm,right=15mm]{geometry}
\end{Preambolo*}

\begin{frontespizio}
  \Istituzione{...}
  ...
  \Correlatore{...}
\end{frontespizio}
\end{document}
```

Si noti che nel corpo dell'ambiente *Preambolo\** si può inserire un intero preambolo, non semplicemente la chiamata al pacchetto *geometry* con le sue opzioni; questo ambiente può contenere qualsiasi cosa sia ragionevole in un preambolo; l'importante è che l'opzione relativa al formato della carta sia identica sia fra le opzioni della classe, sia fra quelle di *geometry* nell'ambiente *Preambolo\**.

Compilando questo documento si ottiene un'unica pagina contenente il frontespizio composto, che può poi venire incorporato nella tesi mediante l'uso del pacchetto *pdfpages* che, come si è già detto, consente di inserire pagine scelte di un file PDF dentro un documento da comporre con pdfL<sup>A</sup>T<sub>E</sub>X. In questo modo la composizione del frontespizio e quello della tesi procedono separatamente e non c'è bisogno, ogni volta che si cambino i dati per il frontespizio, di eseguire le tre compilazioni.

## 3.6 Come si comincia

Non vi dico niente di come si scriva una tesi; so per esperienza che all'inizio ci si sente sperduti davanti allo schermo vuoto. Ma i laureandi possono scaricare dal portale della didattica del sito del Politecnico <http://didattica.polito.it> l'opuscolo *Saper Comunicare — Cenni di scrittura tecnico-scientifica* dove c'è scritto più o meno tutto quel che bisogna sapere per impostare e scrivere un rapporto tecnico-scientifico qual è la tesi, la monografia o la dissertazione.

Qui dirò solo come si comincia a scrivere il file sorgente della tesi. Ci sono essenzialmente due vie, ognuna delle quali offre vantaggi e svantaggi, quindi non si può dire a priori quale via sia la più indicata.

1. Si compone un unico file come esemplificato con i file `toptesi-example.tex`, `toptesi-example-xetex.tex` e `toptesi-example-con-frontespizio.tex` allegati alla documentazione di TOPtesi. Questi file permettono di sperimentare diverse cose semplicemente mettendo o togliendo dei segni di commento all'inizio delle righe del preambolo che contengono alcuni comandi particolari. Essi sono

completi di bibliografia e quell'unico file permette di comporre una tesi completa; ovviamente può essere usato come modello per una tesi; basta cambiare il contenuto dei comandi che contengono i nomi dei candidati, dei relatori, del titolo, dell'ateneo, della eventuale facoltà, eccetera. Ovviamente bisogna cambiare anche il contenuto dei capitoli e della bibliografia.

2. Si spezza il file sorgente in un numero adeguato di file parziali da gestire come specificato qui di seguito; ogni file parziale conterrà solo una parte della tesi, per esempio un solo capitolo; siccome il tutto è gestito dal master file, ciascun file parziale non dovrà più contenere la dichiarazione della classe né il preambolo né le righe `\begin{document}` e `\end{document}`, ma solo il materiale relativo ad uno specifico capitolo cominciando appunto con la riga che ne specifica il titolo: `\chapter{Il titolo di questo capitolo}`.

Il file che contiene il capitolo può venire chiuso con il comando `\endinput`, dopo il quale il file può contenere qualunque cosa che non verrà mai letta dal programma di tipocomposizione, ma per il laureando potrebbe essere molto utile per scrivere alcune annotazioni personali.

Avendo già predisposto una scaletta da seguire per comporre la tesi e avendo dato un nome a ciascun capitolo (nome un po' più espressivo di quelli che scriverò io qui a titolo di esempio), si predispone un master file con un nome "diverso" da `tesi.tex`; bisogna sbizzarrire la fantasia, ma che il nome sia mnemonico e ricordi subito a che cosa ci si riferisce. Questo master file sarà dunque composto, per esempio, così:

```
\documentclass[b5paper,11pt]{toptesi}
\includeonly{%
    preliminari,%
    capitolo1,%
    capitolo2,%
    capitolo3,%
    appendici,%
    bibliografia%
}
\begin{document}
\ateneo{Università di Bengodi}
\nomeateneo{La Saggezza}
\titolo{Studi di tricotetratomia applicata}
\sottotitolo{Cosa succede quando si spacca un capello in quattro}
\candidata{Maria Chiomafolta}
\relatore{prof.\ Piero Scapigliati}
\sedutadilaurea{Maggio 2125}
%
```

```
\include{preliminari}  
\include{capitolo1}  
\include{capitolo2}  
\include{capitolo3}  
\appendix  
\include{appendici}  
\include{bibliografia}  
\end{document}
```

È da osservare in particolare che l'aver scritto l'argomento di `\includeonly` nel modo esemplificato sopra, permette di inserire dei segni di commento all'inizio di alcune delle righe in modo da eseguire, per esempio, la compilazione selettiva di un solo capitolo conservando tutte le informazioni incrociate relative agli altri capitoli già compilati. Con i PC di oggi la compilazione è rapidissima, quindi lo scopo di risparmiare tempo di compilazione non è il motivo per il quale raccomando di seguire questa procedura; faccio invece notare che tenere separate le varie parti della tesi permette di lavorare con più ordine e di risparmiare non poco tempo per la gestione dei vari file sorgente e per le loro correzioni.

Non sarebbe male nemmeno tenere sgombra la cartella di lavoro, cercando di confinare tutto il materiale ausiliario (per esempio disegni, fotografie, brani di codice, eccetera) in sottocartelle della cartella di lavoro; ma queste raccomandazioni sono quelle che valgono per ogni lavoro da eseguire con  $\text{\LaTeX}$  e il lettore con un minimo di esperienza sa già come gestire. All'occorrenza vale la pena di ripassare l'argomento nella guida *L'Arte di scrivere con LaTeX* da scaricare dalla sezione Documentazione del sito <http://www.guitex.org>.

Il file `preliminari.tex` conterrà presumibilmente il comando per comporre il frontespizio, il sommario, molto facoltativamente i ringraziamenti<sup>6</sup> e gli indici.

Il file `capitolo1.tex` conterrà l'introduzione e in pratica esporrà lo stato dell'arte oltre allo scopo del lavoro e ai risultati che si vorrebbero ottenere.

I file `capitolo2.tex`, `capitolo3.tex`, eccetera, conterranno la descrizione del lavoro svolto e l'analisi critica dei risultati via via ottenuti; l'ultimo file prima delle appendici conterrà i commenti finali, quali ad esempio un sunto dei risultati raggiunti, una breve esposizione degli ulteriori sviluppi, possibili approfondimenti, eccetera.

---

<sup>6</sup>I ringraziamenti in una tesi non sono quasi mai necessari; lo sono quando bisogna ringraziare una istituzione o una o più persone *esterne* alla propria università che hanno messo a disposizione le loro strutture o le loro competenze per lo svolgimento della tesi. Non si ringraziano genitori, fidanzati/e, amici, e tutti i parenti di ogni grado, il quali saranno più che contenti di sapervi laureati, piuttosto che vedere il loro nome scritto sulla tesi.... Inoltre rappresenta un atto di piaggeria ringraziare il o i propri relatori; essi hanno svolto un loro compito istituzionale, non vi hanno fatto un piacere. Naturalmente questo non vieta che la consuetudine del lavoro assieme non abbia dato origine a sentimenti di reciproca stima e anche di amicizia, ma questa è un'altra faccenda.



Nelle varie appendici verranno raccolte le misure eseguite, gli sviluppi matematici, i listati dei programmi, le note critiche, le lunghe citazioni e tutte le altre cose che rovinerebbero la fluidità del testo, se fossero inserite direttamente nel corpo della tesi.

La bibliografia conclude la tesi. A seconda del campo disciplinare in cui si svolge la tesi può essere importante scegliere un metodo di citazione piuttosto di un altro; la collezione di pacchetti del sistema  $\text{T}_{\text{E}}\text{X}$  vi consente un'ampia scelta di personalizzazioni per gli stili di composizione e di citazione. Da alcuni anni, poi, è disponibile il pacchetto di estensione *biblatex* che consente una gestione diretta della bibliografia mediante comandi  $\text{L}^{\text{A}}\text{T}_{\text{E}}\text{X}$ ; in ogni caso per queste personalizzazioni è necessario creare un database bibliografico (con estensione *.bib*) in un formato particolare, e poi è necessario elaborare questo database mediante il programma *bibtex* o il programma *biber* (predefinito con *biblatex* dal 2012), sempre facente parte della distribuzione del sistema  $\text{T}_{\text{E}}\text{X}$ . I risultati che si possono ottenere sono estremamente professionali.

### 3.6.1 La scelta della codifica per il file sorgente

Si noti che per comporre i file del corpo della tesi si possono usare direttamente le lettere accentate della tastiera italiana o quelle che si possono ottenere con combinazioni di tasti sia sotto Windows sia sotto Linux o Mac OS X; per default  $\text{TOP}_{\text{tesi}}$  non specifica nessuna particolare codifica per l'input; ma perché il programma di tipocomposizione elabori correttamente i file sorgente, o si usa *xelatex* o *lualatex* ricordando di salvare i file con la codifica UNICODE o UTF-8, oppure non si specifica nessuna particolare codifica ma si usano solo i caratteri ASCII a 7 bit; oppure ancora si usano i caratteri nazionali direttamente da tastiera ma si specifica la codifica che l'editor ha impiegato per salvare nei file i codici dei caratteri non ASCII, per esempio le lettere accentate; a meno che non usiate *xelatex* o *lualatex*, vi suggerisco di specificare esplicitamente la codifica mediante il comando

```
\usepackage[utf8]{inputenc}
```

che va bene sia per le macchine Windows, sia per quelle basate su UNIX.

Gli editor  $\text{T}_{\text{E}}\text{X}$ Shop,  $\text{T}_{\text{E}}\text{X}$ works,  $\text{TeX}$ studio,  $\text{Tex}$ maker, e qualcun altro, consentono di usare la codifica *utf8* che permette di inserire qualunque segno fra le decine o centinaia di migliaia che la codifica UNICODE consente – se usate *xelatex* o *lualatex* questa codifica è necessaria ma non bisogna esplicitarla nel file sorgente. Il problema, in effetti, non è la codifica, ma è la tastiera. Con sistemi operativi diversi si sono sviluppati programmi di vario genere per scrivere in alfabeti diversi da quello latino esteso, qual è quello invocato, per esempio, con l'opzione *latin1* oppure *utf8*. Ma se dovete scrivere la vostra tesi con brani in greco o in russo o in arabo o in lingue con segni sillabici o ideogrammi, questo problema si presenterebbe qualunque sia l'editor e il programma di composizione che usate, ed è un problema che dovete risolvere da soli.

Bisogna notare che in italiano vanno accentate anche le lettere maiuscole; siccome la tastiera italiana non le prevede, consiglio di inserire nel preambolo la definizione del

comando `\E` per scrivere È. Credo che questa sia la maiuscola accentata più frequente in italiano e le altre non capitano mai all'inizio di un periodo; possono capitare nei titoli e allora bisogna arrangiarsi con codici numerici o con tabelle di caratteri. Il comando:

```
\usepackage{xspace}
\newcommand*\E{\`E\xspace}
```

inserisce direttamente una È all'inizio di un periodo e inserisce direttamente lo spazio successivo, per cui, senza inserire esplicitamente nessun comando di spaziatura dopo il comando `\E`, si può scrivere semplicemente:

```
\E conveniente rilevare che\dots
```

per ottenere:

È conveniente rilevare che...

L'uso del pacchetto *xspace* fa sì che esso, tramite il comando `\xspace` inserisca uno spazio, a meno che il comando `\E` non sia seguito da qualunque cosa che non sia uno spazio, per esempio un segno di interpunzione, o una tilde di legatura; anche il comando esplicito di spazio `\_` impedisce l'inserimento di un ulteriore spazio da parte di `\xspace`.

Se il vostro editor vi consente di selezionare le maiuscole accentate da una tabella oppure vi consente di usare combinazioni di tasti per inserire direttamente le maiuscole accentate nel file sorgente, fatene uso, perché la lettura del file sorgente ne guadagna moltissimo.

Se volete o dovete usare *pdflatex*, dovete caricare nel vostro preambolo tutto quello che è necessario per gestire quelle lingue scritte con alfabeti diversi, ma non chiedetemi come si fa, perché so dirvelo solo per il greco. Per questa lingua, se usate l'ortografia monotonica, basta inserire l'opzione della lingua *greek* fra le opzioni *della classe*; se usate l'ortografia politonica<sup>7</sup>, allora dovete richiedere per *babel* lo specifico attributo seguendo questo procedimento (per *polyglossia* si veda più avanti):

```
\documentclass[... ,greek.ancient,...]{toptesi}
...
\begin{document}
...
\end{document}
```

oppure

---

<sup>7</sup>Il greco moderno può essere politonico o monotonico; quello antico è sempre e solo politonico; però le regole di sillabazione sono diverse; con *polyglossia* non ci sono problemi; con *babel* a tutt'oggi (ottobre 2015) esiste l'attributo *ancient* per selezionare la sillabazione antica oltre alla grafia politonica, ma non è ancora possibile disporre effettivamente della sillabazione antica completa.

```
\documentclass[... ,greek,...]{toptesi}
...
\languageattribute{greek}{ancient}
\begin{document}
...
\end{document}
```

Al posto di `ancient` si può specificare `polutoniko`; `polutoniko` si riferisce al greco moderno scritto con tutta la varietà di accenti e diviso in sillabe con i criteri adatti alla lingua moderna; `ancient` si riferisce al greco classico, che oltre ad usare la solita varietà di accenti, usa altre regole per la sillabazione di questa varietà di lingua.

Per una tesi che tratti il greco antico in forma filologica, consiglio l'uso del pacchetto *teubner*; visto che l'ho scritto io, un po' di *réclame* non guasta ... Bisogna però leggerne la documentazione con attenzione. In particolare conviene ricordare che *teubner* non viene caricato se non è stato precedentemente specificata la lingua greca fra le opzioni, ma poi provvede da solo ad impostare l'attributo `ancient` e tutte le altre caratteristiche necessarie per comporre in greco con il *mark-up* filologico. A tutt'oggi (2016) il pacchetto *teubner* non è stato adeguatamente controllato sotto  $\text{\LaTeX}$ . Probabilmente non è completamente compatibile, ma non sono ancora in grado di dire niente né in positivo né in negativo.

Immagino che per usare le lingue che si scrivono in cirillico, basti specificare l'opzione della lingua, come si fa per il greco. Per alcune altre lingue bisogna invece caricare l'apposito pacchetto, il quale provvede a tutte le impostazioni necessarie; in questi casi bisogna leggere con attenzione la documentazione di quei pacchetti.

Suggerisco di riferirsi al documento *L'Arte di scrivere in diverse lingue con (Xe)LaTeX*, liberamente scaricabile dalla sezione Documentazione del sito [www.guitex.org](http://www.guitex.org). Questa guida tratta molti argomenti utili per comporre in lingue straniere comprese alcune di quelle che si scrivono con insiemi di caratteri diversi da quelli latini; descrive anche molte caratteristiche delle tradizioni tipografiche in uso in nazioni diverse dall'Italia oltre, ovviamente, la descrizione delle tradizioni italiane.

Ma, tornando alla codifica UTF-8, il problema non è tanto l'editor che consenta o non consenta di usare quella codifica; quasi tutti i sistemi operativi consentono di installare diversi driver per la tastiera del PC, cosicché con un semplice click del mouse si può passare da un alfabeto all'altro; il problema sono i tasti della tastiera sulla cui faccia è riportato il segno che si ottiene premendo quel tasto quando è attivo il driver corrispondente a quella specifica tastiera. Non è ovviamente possibile cambiare il disegno che appare sui tasti, semplicemente cliccando per cambiare driver; forse lo si può fare con le tastiere virtuali dei PC *touch screen*, ma con i PC normali questo non è possibile. Ecco quindi che la difficoltà di scrivere con un alfabeto diverso richiede una memoria ferrea per ricordare a quale segno dell'altro alfabeto corrisponda il tasto della nostra tastiera.

Per il greco moderno e classico, se si usano i font di default con la codifica LGR (Local GReek), che vengono installati praticamente con ogni installazione del sistema  $\text{\TeX}$  aggiornata e, soprattutto, completa, le corrispondenze con la tastiera latina sono studiate

in modo da semplificare molto questo processo di “memorizzazione”. Tuttavia ad una persona abituata a leggere il greco scritto (giustamente) in greco, vederlo scritto con caratteri latini fa un certo effetto; la documentazione di babel e quella di teubner ricordano questa corrispondenza fra i tasti latini e i segni greci (lettere, accenti, spiriti, segni speciali, ecc.). La corrispondenza fra greco e latino è abbastanza semplice perché il greco ha 24 maiuscole e 25 minuscole, mentre il latino ne ha 26 di ciascun tipo. Non so che cosa succeda con il cirillico che ha un alfabeto composto di più di 30 segni, per cui la corrispondenza diretta con l'alfabeto latino è impossibile.

Con le macchine Macintosh il sistema operativo Mac OS X mette a disposizione una specie di tastiera virtuale sui tasti della quale si può cliccare con il mouse; si possono installare diverse tastiere che possono venire scambiate con un semplice click sulla icona disegnata nella barra superiore; per cui scrivere in greco politonico o in russo o in ebraico, o in..., non è un problema. Temo che lo sia con altre macchine e altri sistemi operativi.

Mi dicono che sulle macchine Linux l'editor Vim e, su tutte le macchine, l'editor emacs prevedano delle combinazioni di tasti che consentono di scrivere direttamente in greco sullo schermo, nella finestra del file `.tex`. Ho letto qualcosa in merito, ma non sono in grado di confermare per esperienza diretta.

### 3.6.2 La scelta della codifica per il file di uscita

Solo per scrivere in inglese non è necessario specificare nessuna codifica per i font da usare per comporre la tesi; questo è vero in quanto l'inglese, fra le lingue più comuni nel mondo occidentale, è l'unica che non fa uso di accenti o altri segni diacritici. Perciò la codifica di default OT1 per i font di uscita adatti a scrivere in inglese va benissimo e non è necessario specificare nulla. Ma attenzione: come ho già detto, se si deve inserire anche una sola parola diversa dall'inglese e che contenga un accento, o se si deve scrivere in “middle English”, che usava gli accenti, questa vecchia codifica non va assolutamente bene per produrre file PDF/A compatibili. Quindi anche per l'inglese sarebbe opportuno usare una codifica più avanzata.

Inoltre con questa codifica OT1 la presenza di accenti impedisce la divisione in sillabe o ne riduce molto l'efficacia; per l'italiano il problema è relativamente modesto perché normalmente gli accenti compaiono solo sull'ultima vocale delle parole *tronche*. Quindi con questa codifica la parola *qualità* viene divisa in “qua-lità” invece che in “qua-li-tà”. Invece in francese una parola come *électricité* non viene divisa per niente a causa dell'accento sulla prima vocale.

Allora per tutte le lingue diverse dall'inglese, ma anche per l'inglese, è quanto mai opportuno specificare per i font di uscita la codifica T1. C'è però un problemino, facilmente risolvibile, ma bisogna pensarci: fino ad oggi (2015) i font di default usati da pdfL<sup>A</sup>T<sub>E</sub>X sono i font della collezione Computer Modern con codifica OT1; e ogni installazione del sistema T<sub>E</sub>X li può usare sia nella versione “bitmapped” sia in quella PostScript Type 1. Quest'ultimo formato dei font dovrebbe essere quello da usare sempre con qualunque font

per il formato PDF del file di uscita, perché questo formato è portabile da una macchina ad un'altra, indipendentemente dal sistema operativo usato. Se poi il file deve essere archiviato, esso deve essere in formato PDF, quindi, di fatto non ci sono scelte.

I file della collezione Computer Modern con codifica T1, conosciuti anche come font EC, sono invece distribuiti solo nella forma bitmapped, la quale, oltre ad essere vietata nei file PDF archiviabili, si legge molto male sullo schermo, se il programma di visualizzazione non è molto ben adattato ai font bitmapped. Esiste la collezione di font cm-super per ovviare a questo inconveniente (questa collezione è installata di default in ogni installazione completa del sistema  $\text{\TeX}$ ), ma l'uso di questi font implica un aumento non trascurabile delle dimensioni del file di uscita. Per aggirare questi inconvenienti è meglio usare i font della collezione Latin Modern, per cui nel preambolo del file da gestire con `pdflatex` si specificherà:

```
\documentclass[...]{toptesi}
...
\usepackage[T1]{fontenc}
\usepackage{lmodern}
...
\begin{document}
...
\end{document}
```

Se il file viene gestito con `xelatex` o `lualatex`, oltre alla codifica `utf8` di default *che non va esplicitata*, basta richiedere l'uso del pacchetto *fontspec* senza ulteriori opzioni e viene automaticamente caricata ed usata la collezione dei font Latin Modern nella loro versione OpenType; a questo provvede direttamente la classe *toptesi*. Usando `xelatex` o `lualatex`, gli appositi comandi per la selezione dei font sia testuali sia per la matematica può essere eseguita nel preambolo senza bisogno di ricaricare il pacchetto *fontenc*. Volendo, invece di usare i font Latin Modern, si possono impostare facilmente altre famiglie di font con comodi comandi che fanno riferimento ai font OpenType installati sulla propria macchina.

Naturalmente per gli utenti esperti di  $\text{\LaTeX}$  ci sono anche altri pacchetti usabili per impiegare altri font diversi da quelli della collezione Latin Modern, che comunque, secondo me, sono i migliori per la loro completezza; grazie al diverso disegno dei corpi minori la lettura dei pedici e degli apici ne risulta agevolata.

Qui, per la composizione della tesi con `pdflatex` segnalo solo la collezione dei font Times eXtended, richiamabili con i pacchetti *newtxtext* e *newtxmath*, e la collezione dei font Palatino eXtended, richiamabili con i pacchetti *newpxtext* e *newpxmath*. I Times sono font più stretti di quelli della collezione Latin Modern e sono forse indicati per rendere più compatta una tesi un po' voluminosa. I Palatino sono un po' più larghi dei font della collezione Latin Modern e allungano la tesi di qualche pagina. I nuovi pacchetti *newtx...* e *newpx...* sono configurabili con diverse opzioni; li consiglio a coloro che vogliono usare i font Times o Palatino, ma se devo dare un consiglio assoluto, direi che sia sempre preferibile usare i

font Latin Modern dalle prestazioni molto migliori grazie ai corpi ottici; volendo si può usare il pacchetto *cfr-lm* che consente di personalizzare fortemente le proprietà dei font Latin Modern da usare, per esempio permettendo l'uso delle cifre minuscole 1234567890 nel testo, e le cifre maiuscole 1234567890 nella composizione della matematica.

Tanto per confronto, questo testo di documentazione su TOPtesi è composto con il motore di composizione *lualatex* usando i font OpenType TeX Gyre Termes con codifica UNICODE presenti nella distribuzione del sistema T<sub>E</sub>X; come si vede la divisione delle parole in fin di riga non presenta inconvenienti particolari; forse con un testo di altro genere, con composizione in colonne strette o all'interno di tabelle, si potrebbe verificare la presenza di righe molto lasche, dove è stato allargato troppo lo spazio interparola. Per questo, quando si usa il motore di composizione *pdflatex*, si consiglia sempre di usare la codifica T1; non si sa mai, ma certamente con questa codifica si hanno meno inconvenienti e le lettere accentate sono disegnate meglio. Se si usa come motore *pdflatex*, è conveniente usare anche il pacchetto di estensione *microtype* che consente di comporre ancora meglio le righe del testo; con *xelatex*, invece, le funzionalità di questo pacchetto sono solo parzialmente usabili; comunque si può usare *microtype* anche compilando con *xelatex* sebbene i miglioramenti siano meno “vistosi” che con *pdflatex*; con *lualatex* il pacchetto *microtype* produce gli stessi benefici che si ottengono con *pdflatex*, senza i piccoli inconvenienti che si manifestano con *xelatex*.

## 3.7 Come stampare la tesi

Il pacchetto TOPtesi contiene anche l'opzione di composizione *in bianca e volta*, ma di default compone su una facciata sola della pagina.

Qual è la differenza? È quella che si osserva nei libri composti meglio, cioè nei libri con i fogli stampati davanti e di dietro: quando il libro è aperto la striscia bianca formata dai due margini interni delle pagine affacciate è circa uguale a ciascuno dei margini esterni. Invece quando si stampa su una facciata sola il corpo del testo è centrato e i margini laterali sono uguali. Inoltre nelle testatine delle pagine stampate solo in bianca compare sempre e solo il titolo del capitolo corrente, mentre nelle testatine delle pagine *pari* stampate in bianca e volta compare il titolo del capitolo, mentre nelle pagine *dispari* compare il titolo del paragrafo con cui comincia la pagina.

Molti laureandi ritengono di fare una cosa utile stampando fronte e retro, o, come dicono i tipografi, in bianca e volta. Certamente è una buona idea, sia perché il fascicolo della tesi viene ad assumere un aspetto più professionale, sia perché, ecologicamente parlando, si consuma meno carta.

Per questo ho cambiato le impostazioni di default della classe *report*, sulla quale è basata la classe *toptesi*, per far sì che comunque la stampa della pagina, sia essa solo in bianca o in bianca e volta, appaia sempre con i margini laterali uguali.

Ho deciso per questa soluzione perché quando si rilega la tesi, necessariamente dal lato della cucitura o dell'incollatura le pagine sono piegate e l'insieme dei due margini interni

delle pagine affacciate appare visivamente più piccolo di ciascuno dei margini esterni. Se si stampa solo in bianca invece sarebbe il caso di aumentare il margine sinistro; tuttavia non ho realizzato questo spostamento non solo perché il margine destro resta di una certa ampiezza e può accogliere le note marginali, se si decide di farne uso, ma anche perché, almeno al Politecnico di Torino, la copia da depositare nella segreteria didattica non deve essere realizzata su carta, ma deve essere registrata su un CD-ROM in formato .pdf e chi legge la tesi sullo schermo preferisce senz'altro avere il testo centrato piuttosto che spostato alternativamente da un lato o dall'altro dello schermo.

Questa è una  
nota marginale

Però questa soluzione potrebbe non piacere ad alcuni laureandi, specialmente se compongono tesi piuttosto voluminose e/o se le rilegano con una costola piuttosto rigida. In questo caso la classe *toptesi* accetta l'opzione cucitura mediante la quale le facciate di sinistra e di destra vengono spostate all'esterno di 7 mm, per compensare la rigidità della cucitura e la forte curvatura delle pagine causate dallo spessore e/o dal tipo di rilegatura. Questo implica che la copia da memorizzare sul CD-ROM dovrà essere compilata senza specificare questa opzione, che invece sarà specificata per ricomporre il file prima della stampa su carta.

## 3.8 Comporre la tesi in diverse lingue

Il pacchetto TOPtesi usa di default il pacchetto *babel* o il pacchetto *polyglossia* per gestire la composizione in diverse lingue. Tuttavia la composizione in lingue diverse dall'inglese (che è la lingua di default per una installazione “vergine” del sistema T<sub>E</sub>X) richiede che il sistema sia configurato per gestire diverse lingue; perciò le operazioni da fare sono essenzialmente due.

1. Bisogna configurare il sistema T<sub>E</sub>X per gestire diverse lingue;
2. Bisogna specificare nel comando di apertura del master file di quali altre lingue ci si vuole servire **oltre all'italiano e all'inglese che sono già automaticamente invocati da TOPtesi**. Si ricorda che non si può più invocare nel preambolo il pacchetto *babel* una seconda volta, quando esso è già stato invocato una prima volta nel corpo della classe. Si veda nel seguito come bisogna specificare le ulteriori lingue oltre all'italiano e all'inglese.

### 3.8.1 Configurazione iniziale

Per la prima parte, cioè per configurare il sistema T<sub>E</sub>X per gestire diverse lingue, subito dopo l'installazione lanciate L<sup>A</sup>T<sub>E</sub>X su un file di prova qualunque, per esempio *sample.tex* (facente parte della distribuzione del sistema T<sub>E</sub>X), poi apritene il file *.log*, nell'esempio *sample.log*; nelle prime righe di questo file troverete le lingue per le quali il vostro sistema è già configurato.

Per esempio, componendo con `pdflatex` questo file di documentazione, il file `toptesi-it-xetex.log` nel 2012 conteneva nelle prime righe quanto segue:

```
This is pdfTeX, Version 3.1415926-2.4-0.9998 (TeX Live 2012)
  (format=xelatex 2012.7.10)  9 SEP 2012 16:04
entering extended mode
\write18 enabled.
file:line:error style messages enabled.
%&-line parsing enabled.
**toptesi-it-xetex.tex
(./toptesi-it-xetex.tex
LaTeX2e <2011/06/27>
Babel <v3.8m> and hyphenation patterns for english, dumylang,
nohyphenation, german-x-2012-05-30, ngerman-x-2012-05-30, afrikaans,
ancientgreek, ibycus, arabic, armenian, basque, bulgarian, catalan,
pinyin, coptic, croatian, czech, danish, dutch, ukenglish,
usenglishmax, esperanto, estonian, ethiopic, farsi, finnish, french,
friulan, galician, german, ngerman, swissgerman, monogreek, greek,
hungarian, icelandic, assamese, bengali, gujarati, hindi, kannada,
malayalam, marathi, oriya, panjabi, tamil, telugu, indonesian,
interlingua, irish, italian, kurmanji, latin, latvian, lithuanian,
mongolian, mongolianlmc, bokmal, nynorsk, polish, portuguese,
romanian, romansh, russian, sanskrit, serbian, serbianc, slovak,
slovenian, spanish, swedish, turkish, turkmen, ukrainian,
uppersorbian, welsh, loaded.
```

Dal 2014, purtroppo l'elenco delle lingue è sostituito dalla seguente laconica frase:

```
Babel <3.9k> and hyphenation patterns for 81 languages loaded.
```

Non c'è bisogno di preoccuparsi; nella mia macchina sono installate alcune lingue in più di quelle di default, ma se il numero è così alto, vuol dire che sono installate tutte le lingue che i programmi di composizione del sistema  $\text{\TeX}$  sono capaci di gestire; fra queste c'è l'italiano, tre versioni di greco, tre versioni di latino, eccetera. L'elenco completo può venire letto nel file `xelatex.log` (o, componendo con `pdflatex`, nel file `pdflatex.log`) che si trovano in uno degli alberi del sistema  $\text{\TeX}$ . La lettura è un po' criptica, anche perché il file è piuttosto lungo, ma basta cercare la stringa `l@english`, e di lì in avanti è registrato tutto quello che serve conoscere per l'elenco delle lingue installate.  $\text{\Lua\TeX}$ , diversamente da  $\text{\pdf\TeX}$  e da  $\text{\X\TeX}$ , carica le impostazioni, sillabazione compresa, per le lingue del documento solo al momento di compilare quel dato documento; per questo compilatore solo l'inglese è precaricato, ma al momento opportuno esso si carica tutto ciò che è necessario per comporre in ciascuna dell'ottantina di lingue che riesce a gestire, o meglio in ciascuna delle lingua specificate per la composizione di ogni particolare documento, massimo un'ottantina (bastano?...).



Quel listato o quella riga è quanto appare sempre quando si usa la distribuzione T<sub>E</sub>X Live completa o quando si usa la distribuzione MiK<sub>T</sub>E<sub>X</sub> 2.9 (o successiva) completa; serve per dare un'idea di quello che contengono le prime righe del file in questione. Se, come è possibile, dopo la prima installazione il vostro primo file `.log` non contiene fra le lingue elencate anche l'italiano (oppure sono elencate solo una mezza dozzina di lingue, oppure si dice che le lingue installate sono in un numero molto inferiore a 80), allora questa è la cosa più urgente da fare; non vorrete mica che la vostra tesi, scritta in italiano, abbia le parole divise in sillabe in fin di riga con le regole angloamericane? Se poi siete su un programma di doppia laurea e dovete scrivere parte della tesi nella lingua dell'università straniera di cui prendete l'altro titolo, allora è necessario avere disponibile almeno anche quella lingua. Questo inghippo avviene quando si scarica una versione ridotta delle distribuzioni del sistema T<sub>E</sub>X, oppure quando si usa la distribuzione T<sub>E</sub>X Live/Debian (sempre sconsigliabile per vari motivi, non escluso il fatto che l'installazione di default prevede solo la sillabazione per l'inglese).

Le regole per inizializzare le lingue gestibili differiscono da distribuzione a distribuzione del sistema T<sub>E</sub>X. Bisogna quindi leggere la documentazione che accompagna la vostra distribuzione; qui darò alcuni cenni relativi alle distribuzioni che io conosco.

## MiK<sub>T</sub>E<sub>X</sub>

Questa distribuzione è forse quella che più sovente viene installata nella forma “basic” (errore! mai installare la versione di base, perché è tutt'altro che completa) e perciò è fra quelle che hanno bisogno di essere gestite. Si apre il wizard da Avvio|Programmi|MiK<sub>T</sub>E<sub>X</sub>|MiK<sub>T</sub>E<sub>X</sub>\_Settings. Esso apre una finestra di dialogo con diverse linguette; si clicca sulla linguetta Languages e viene aperta un'altra finestra con l'elenco di tutte le lingue gestibili, alcune già con il segno di spunta alla loro sinistra nell'apposito quadratino, altre senza il segno di spunta. Potete cliccare per togliere il segno di spunta sulle lingue già spuntate ma che non userete mai, ma ve lo sconsiglio, e potete aggiungere il segno di spunta cliccando nel quadratino corrispondente alle lingue desiderate, ma vi consiglio di spuntarle tutte.

Poi uscite dal dialogo per la scelta delle lingue cliccando OK e tornate sulla linguetta General; qui trovate due bottoni il primo dei quali serve per aggiornare il database dei nomi dei file, mentre il secondo serve per re-inizializzare i file di formato, cioè quei file che contengono già la traduzione in linguaggio macchina di tutte le operazioni che la vostra distribuzione del sistema T<sub>E</sub>X è capace di compiere con i suoi vari applicativi. La divisione in sillabe è una di quelle operazioni che deve essere inizializzata, perché richiede strutture dati particolari che sarebbe troppo lungo generare di volta in volta (Lua<sub>L</sub>T<sub>E</sub>X, però ci riesce). Cliccate sul bottone per re-inizializzare i file di formato; alla fine di questa operazione potete chiudere il wizard e controllare, dopo aver ricomposto, per esempio, `sample.tex` che le prime righe del file `.log` contengano tutte le lingue che volete usare.

## **T<sub>E</sub>X Live**

La distribuzione T<sub>E</sub>X Live completa distribuita sul circuito [ctan](#) nasce già configurata per tutte le lingue che L<sup>A</sup>T<sub>E</sub>X è capace di gestire, quindi anche l'italiano; bisogna ricorrere ai suoi comandi di configurazione solo se si vuole aggiungere una lingua la cui sillabazione non sia distribuita insieme a T<sub>E</sub>X Live. Questo evento è talmente raro che non vale la pena di insistervi sopra. Tuttavia un laureando che svolga la sua tesi sull'ostrogoto altomedievale e disponesse solo della sillabazione per l'ostrogoto classico e per l'ostrogoto contemporaneo, si troverebbe in difficoltà; infatti, prima ancora di inserire le regole per l'ostrogoto altomedievale, dovrebbe scriversi le regole di sillabazione e codificarle nel linguaggio specifico richiesto dal sistema T<sub>E</sub>X; questo è tutt'altro che facile ed è riservato a specialisti molto avanzati.

## **MacT<sub>E</sub>X**

Dal 2007 la distribuzione di MacT<sub>E</sub>X sostanzialmente coincide con T<sub>E</sub>X-live, salvo che contiene anche programmi accessori specifici per le macchine Macintosh; è preconfigurato per gestire tutte le lingue di cui il sistema è capace, compreso l'italiano. Ma per l'ostrogoto altomedievale...

## **Distribuzioni commerciali**

Le distribuzioni commerciali differiscono da quelle gratuite essenzialmente per i programmi accessori che accompagnano la distribuzione del sistema T<sub>E</sub>X che, di sua natura, è gratuito; il costo delle distribuzioni commerciali corrisponde al valore delle parti accessorie, all'assistenza per i clienti, eccetera.

Fra le parti accessorie in generale ci sono anche i programmi realizzati attraverso comode interfacce grafiche per la inizializzazione e l'aggiornamento del software.

Perciò con queste distribuzioni bisogna leggere il manuale di istruzioni e provvedere corrispondentemente.

### **3.8.2 Le lingue della tesi**

Di default TOPtesi compone la tesi in italiano e per scrivere la tesi in italiano non occorre altro.

Se si devono inserire nella tesi brani di testo in lingua straniera, ma lasciando la struttura della tesi in italiano, se si usa pdf<sub>l</sub>atex basta elencare fra le altre opzioni nel comando di apertura del master file i nomi (inglesi) delle lingue da usare; per esempio, per inserire brani in francese si scriverà:

```
\documentclass[12pt,twoside,french]{toptesi}
```

Se invece si vuole scrivere la tesi in inglese, lingua già inserita di default in TOPtesi, ma non attivata, bisogna dare l'indicazione esplicita dopo l'inizio del documento mediante il comando `\english`; così:

```
\documentclass[11pt,cucitura]{toptesi}
...
\begin{document}
\english
...
```

Per commutare dall'italiano all'inglese e viceversa basta alternare le dichiarazioni `\italiano` e `\english`. Attenzione: queste dichiarazioni alterano anche le parole come Chapter o Capitolo, Table o Tabella, quindi per inserire brani nell'altra lingua è opportuno servirsi dei comandi di *babel*, in particolare l'ambiente *otherlanguage*, per esempio:

```
... disse:
``\begin{otherlanguage}{english}
  Mr Livingstone, I suppose\dots
\end{otherlanguage}''
e si strinsero la mano.
```

L'ambiente *otherlanguage* è adatto per citazioni relativamente lunghe; per citazioni brevi, come quella dell'esempio, sarebbe meglio usare così il comando `\foreignlanguage`:

```
disse: ``\foreignlanguage{english}{Mr Livingstone, I suppose\dots}''
e si strinsero la mano.
```

Con i programmi xelatex e lualatex le lingue ausiliarie, oltre l'italiano e l'inglese già preinstallate in *toptesi*, basta specificare nel preambolo, per esempio:

```
\setotherlanguages{french,spanish}
```

Oppure, se la lingua richiede un alfabeto speciale, se ne può specificare il font specifico e prescrivere, per esempio:

```
\setotherlanguage[variant=ancient]{greek}
\newfontfamily{\greekfont}{GFS Didot}
```

Ciò fatto si dispone per ogni lingua, tranne quella principale, di un ambiente dal nome uguale alla lingua, che all'occorrenza seleziona anche il font specifico, la cui famiglia abbia un nome che comincia con lo stesso nome della lingua. Per esempio:

```
\begin{greek}
Οἰόνται τινές, βασιλεῦ Γέλων, τοῦ ψάμμον τὸν ἀριθμὸν ἄπειρον εἶμεν
τῷ πλήθει· λέγω δὲ οὐ μόνον τοῦ περὶ Συρακούσας τε καὶ τὰν ἄλλαν
Σικελίαν ὑπάρχοντος, ἀλλὰ καὶ τοῦ κατὰ πᾶσαν χώραν τάν τε οἰκημέναν
καὶ τὰν ἀοίκητον.
\end{greek}
```

Anche usando *xelatex* o *lualatex* bisogna definire i comandi `\sommario` e `\ringraziamenti` nella stessa maniera illustrata qui di seguito per comporre con il programma *pdflatex*.

Invece per comporre la tesi in una lingua diversa dall'italiano e dall'inglese bisogna lavorare un pochino di più perché bisogna ridefinire alcune cose; per esempio, per scrivere la tesi in spagnolo bisogna agire così:

```
\documentclass[12pt,spanish]{toptesi}% <--- la lingua come opzione
                                     %      della classe!
\ExtendCaptions{spanish}{Resumen}{Agradecimientos}
\newcommand*{\spagnolo}{\selectlanguage{spanish}}%
...
\begin{document}
\spagnolo
...
\end{document}
```

Ovviamente le parole “Resumen” (Sommario) e “Agradecimientos” (Ringraziamenti) andranno scelte accuratamente; io ho indicato solo ciò che ho trovato sul vocabolario.

È anche possibile servirsi del file di configurazione; per esempio le definizioni dei comandi suddetti possono venire scritte nel file di configurazione e alternando i comandi `\italiano`, `\english` e `\spagnolo` si possono alternare le regole di sillabazione oltre a cambiare le parole *infix*, come Chapter, Capítulo, Capítulo, eccetera.

Se si usa *xelatex* o *lualatex* ci si ricordi che il pacchetto *polyglossia* è già caricato dalla classe *toptesi*, che provvede già a dichiarare l'italiano come lingua principale e l'inglese come lingua accessoria; si possono nominare nel preambolo della tesi diverse altre lingue accessorie, per esempio:

```
\setotherlanguages{french, spanish}
```

e per le lingue che richiedono un trattamento particolare si usa un comando simile (al singolare), per esempio:

```
\setotherlanguage[variant=ancient]{greek}
```

Per scrivere in lingue che implicano alfabeti diversi da quello latino, ovviamente, bisogna aver curato di disporre di una configurazione del sistema  $\text{\TeX}$  completa anche dei font che servono. Non dovrebbero esserci problemi con il cirillico e con il greco; per altri alfabeti e per le lingue che si scrivono da destra a sinistra bisogna ovviamente disporre dei pacchetti necessari. Si tenga presente che le versioni moderne del sistema  $\text{\TeX}$  *non* usano come interprete il programma originario di Knuth, ma la sua estensione che in origine si chiamava  $\epsilon\text{\TeX}$ ; la variante che produce il file del documento composto in formato .pdf si chiama *pdftex*, ma questa versione comprende tutte le estensioni di  $\epsilon\text{\TeX}$ . Bene, queste estensioni servono anche per gestire le lingue con scrittura retrograda pur di disporre dei pacchetti relativi a queste lingue.

Anche *xelatex* e *lualatex* sono in grado di gestire le lingue retrograde e sono capaci con modesti adattamenti di alcuni comandi di comporre in verticale (cinese, giapponese,...).

## 3.9 Il formato PDF/A

Il Politecnico di Torino, come anche molte università italiane e straniere, intende “smaterializzare” le copie di archivio delle tesi di laurea. Questo intendimento è cominciato nel 2008 ma non sono al corrente dell’esito di questa operazione a cui ho collaborato parzialmente prima di andare in pensione.

Il problema dell’archiviazione elettronica è che il materiale archiviato deve essere reperibile, leggibile e stampabile per un tempo indefinito, anche secoli! Per questo è necessario che esso sia archiviato in un formato standard e che nel futuro continui ad esistere i programmi per la visualizzazione e la stampa di questo formato, non necessariamente quelli che esistono oggi.

L’International Standards Organization (ISO) ha pubblicato nel 2005 uno standard per l’archiviazione dei file con la norma ISO 19005-1:2005 (nel 2011 e poi ancora nel 2012 sono stati approvati due aggiornamenti dello standard, ma per quello che interessa qui non è necessario vederne le modifiche in dettaglio). Secondo questa norma i file archiviabili devono avere il formato PDF della versione 1.4 e devono soddisfare ad altri requisiti, oltre a contenere un certo numero di *metadati* specificati dalla norma stessa. Esistono due sottoformati: il PDF/A-1a e il PDF/A-1b. Il formato PDF/A-1a deve rispondere ai requisiti del formato PDF/A-1b oltre ad altri requisiti specifici, in particolare che i font siano tutti inclusi nel file ed abbiano codifica UNICODE, che il file debba mantenere le informazioni relative alla sua struttura logica, e che queste possano essere esaminate con i motori di ricerca. Viceversa il formato PDF/A-1b, con meno pretese, richiede che i font siano tutti inclusi nel file, anche se non rispettano la codifica UNICODE, e che il file sia riproducibile a schermo e sia stampabile esattamente nello stesso modo di quando il file è stato archiviato. Per il Politecnico sarebbe sufficiente il sottoformato PDF/A-1b. Mi risulta che sia allo studio da parte del Politecnico la possibilità di richiedere il rispetto di una norma molto più restrittiva, che garantisca l’accessibilità e la fruibilità dei documenti archiviati anche a coloro che hanno alcune forme di disabilità; tuttavia a tutt’oggi (2016) l’ISO non ha ancora messo a punto tale norma, quindi l’apposita commissione del Politecnico sta giustamente precorrendo i tempi perché l’Ateneo sia pronto ad attuare la futura norma. Nel frattempo l’apposito gruppo di lavoro del TUG sta cercando di estendere i suoi pacchetti per l’archiviabilità anche per la norma PDF/A-1a, che richiede di disporre di file PDF classificati come *tagged PDF*.

Le istruzioni che seguono per la produzione di un file conforme alla normativa PDF/A sono applicabili se si compone la tesi sia con il programma `pdflatex` sia con `xelatex` sia con `lualatex`. In particolare, per sfruttare al meglio le possibilità offerte da `xelatex` è necessario usare il pacchetto *fontspec* che, senza ulteriori opzioni o specificazioni, usa di default i font Latin Modern nella versione OpenType; volendo si possono usare i comandi specifici disponibili con quel pacchetto per caricare font diversi. Notate che se si è predisposto il preambolo del file sorgente per la tesi in modo da poterlo compilare con `pdflatex`, si può ugualmente compilare il documento anche con `xelatex` o `lualatex`, perché TOPtesi è

perfettamente conscio del motore di composizione che si sta usando e sa che il file sorgente parte dal presupposto che si volevano creare gli hyperlink. In altre parole, non c'è bisogno di modificare sostanzialmente il file sorgente per passare da pdf<sub>l</sub>atex a xelatexo lua<sub>l</sub>atex, se non eventualmente usando solo il ramo *vero* oppure *falso* del test indicato qui sotto. Però il file prodotto con xelatex necessita di ulteriore elaborazione per soddisfare le specifiche richieste dalla norma ISO relativa al formato archiviabile; con pdf<sub>l</sub>atex e lua<sub>l</sub>atex non sono necessarie operazioni di postprocessing.

Si può comunque fare riferimento alla parte iniziale del preambolo del file usato per comporre questa stessa documentazione, che qui si riporta nuovamente:

```
1 % !TEX encoding = UTF-8 Unicode
2 % !TEX TS-program = luaLaTeX
3 \begin{filecontents*}{\jobname.xmpdata}
4 \Title{La classe TOPtesi}
5 \Author{Claudio Beccari}
6 \Publisher{Claudio Beccari}
7 \Keywords{Monografia di laurea\sep
8 Tesi di laurea\sep
9 Tesi di dottorato\sep
10 classe LaTeX\sep
11 pdfLaTeX\sep
12 XeLaTeX\sep
13 LuaLaTeX}
14 \end{filecontents*}
15 %
16 \documentclass[12pt,twoside]{toptesi}
17 \ProvidesFile{toptesi-it.tex}[2016/06/30 v.0.99e]
18 \ifPDFTeX
19 %
20 \usepackage[a-1b]{pdfx}
21 %
22 \usepackage[utf8]{inputenc}
23 \usepackage[T1]{fontenc}
24 \usepackage{newtxtext,newtxmath,textcomp,textalpha}
25 \usepackage{amsmath,amssymb}
26 \usepackage{mflogo}
27 \usepackage{guit}
28 \setactivedoublequote
29 \else
30 \ifLuaTeX
31 \usepackage[a-1b]{pdfx}
32 \fi
33 \usepackage{fontspec}
34 \usepackage{xcolor}
```

```
35 \newcommand*{\MP}{\setbox0\hbox{M}\relax
36 \includegraphics[height=\ht0]{MPlogo}}
37 \newcommand*{\GuIT}{\setbox0\hbox{Hg}\relax
38 \raisebox{-\dp0}{\includegraphics[height=\dimexpr\ht0+\dp0]{GuITlogo}}}}
39 \setmainfont[Ligatures=TeX]{TeX Gyre Termes}
40 \setsansfont[Ligatures=TeX, Scale=MatchLowercase]{TeX Gyre Heros}
41 \setmonofont{UM Typewriter}
42 \setmainlanguage[babelshorthands]{italian}
43 \usepackage{amsmath}
44 \usepackage{unicode-math}
45 \setmathfont{XITS Math}
46 \fi
47 \usepackage{metalogo,longtable,booktabs,array,tabularx,fancyvrb}
48 %
49 \usepackage{imakeidx}
50 \usepackage{hyperref}
51 \hypersetup{%
52     pdfpagemode={UseOutlines},
53     bookmarksopen,
54     pdfstartview={FitH},
55     colorlinks,
56     linkcolor={blue},
57     citecolor={blue},
58     urlcolor={blue}
59 }
```

Un commento va fatto subito: il lettore non si spaventi di una prima parte di preambolo così complessa, e non ha nessun bisogno di copiarla integralmente. Per questo documento ho predisposto un test per verificare con quale programma di composizione compilo il documento e come si vede imposto i font e le lingue come specificato sopra, ma nel ramo *falso* (cioè non si sta compilando con pdf<sub>l</sub>atex ma con lua<sub>l</sub>atex oppure xel<sub>l</sub>atex; tuttavia se il programma con cui si compila si evita di caricare il pacchetto speciale *pdfx*, necessario per produrre file PDF/A compatibili. Il lettore sa perfettamente con quale programma vuole compilare la sua tesi e può tranquillamente mettere nel suo preambolo solo il ramo *vero* del test se compila con pdf<sub>l</sub>atex o solo il ramo *falso* se compila con lua<sub>l</sub>atex o con xel<sub>l</sub>atex. Se vuol produrre un file PDF/A compatibile con xel<sub>l</sub>atex ha due vie:

- toglie il test `\ifLuaTeX` e usa solo il ramo *falso* del test `\ifPDFTeX`; tuttavia ciò non basta e deve poi procedere come specificato nel paragrafo 3.9.3; oppure
- lascia perdere xel<sub>l</sub>atex e compila con lua<sub>l</sub>atex; il frutto della compilazione è persino migliore, visto che lua<sub>l</sub>atex usa le funzionalità del pacchetto *microtype* in modo più avanzato rispetto a xel<sub>l</sub>atex.

Nel codice precedente le prime due righe sono commenti per il programma di compilazione, ma per il programma di editing sono comandi di autoconfigurazione; essi dicono al programam di editing di salvare il documento dopo la prima creazione e dopo ogni modifica con la codifica `utf8`; e la seconda dice che quando si preme il bottone per la compilazione deve dare al sistema operativo di lanciare il programma `lualatex`. Ricordo che questo tipo di commenti di autoconfigurazione sono decifrati corretta mente dagli editor TeXShop, TeXworks e TeXstudio (forse anche da altri editor, ma questi tre sono quelli che io consiglieri per ogni utente).

Le righe dalla 3 alla 14 contengono l'ambinete *filecontents\** che provvede a generare la lista dei metadati che ho usato per questo documento.

La riga 16 contiene la dichiarazione della classe; come si vede specifico solo il corpo normale dei font al valore di 12 pt, e seleziono la composizione in bianca e volta, successivamente non specifico nessuna correzione per la legatura. Importante per questo documento è la riga 17, dove specifico il nome del file, ma ne specifico anche la data e la versione; così sono sempre sicuro di quale versione mi sto servendo e i lettori che trovano questo file fra la documentazione di *toptesi* sanno subito se si tratta di documentazione aggiornata o se non sia il caso di aggiornare la propria installazione.

Con la riga 18 comincia il test importante per configurare il preambolo correttamente secondo le necessità di compilazione; dalla riga 19 alla riga 28 si ha il ramo *vero* da usarsi quando si compila con `pdflatex`; invece dalla riga 30 alla riga 45 si ha il ramo *falso* da usarsi quando si compila con `lualatex`; per compilare con `xelatex` ho già detto che, oltre a togliere le righe 30 e 32 (ma non la riga 31), bisogna poi procedere come specificato nel paragrafo 3.9.3.

Le righe successive dalla 47 alla 59 servono per mostrare non solo i pacchetti che ho caricato, ma anche come ho configurato le opzioni per *hyperref*; nessuna opzione nella chiamata del pacchetto, ma tutte le impostazioni eseguite mediate l'argomento del comando `\hypersetup`.

Vorrei sottolineare l'opportunità di specificare come indicato nel codice precedente i metadati richiesti per la compilazione del documento in modo che sia conforme con lo standard PDF/A. Non è obbligatorio ma in questo modo si è sicuri che il file dei metadati abbia l'estensione giusta e il nome assolutamente coincidente con il nome del main file del documento, cioè della tesi.

L'altro punto importante è non solo caricare il pacchetto *pdfx*, ma è necessario caricarlo nell'ordine indicato, cioè come primo pacchetto.

Ho collaudato questo preambolo compilando sia con `pdflatex` sia con `lualatex`; volendo, avrei potuto seguire le raccomandazioni che scrivo nel paragrafo 3.9.3 ma non l'ho fatto; visto che ottengo lo stesso risultato, anzi un risultato migliore, con `lualatex` mi è parso un collaudo superfluo. Con questo non voglio dire che `xelatex` sia diventato inutile; esso come `pdflatex` ha un file di formato che contiene tutte le impostazioni per tutte le lingue che il sistema T<sub>E</sub>X può gestire, mentre `lualatex` in questo momento di ogni lingua può gestire una sola impostazione per ogni lingua; per esempio il greco ha tre varianti e il file di



formato di xelatex le potrebbe gestire tutte e tre come ci riesce pdflatex; lo stesso vale per il latino e per diverse altre lingue; lualatex al momento può gestire una sola variante di ogni lingua. Perciò in questo momento xelatex potrebbe essere preferibile a lualatex per le tesi di carattere linguistico, filologico, umanistico in generale. Personalmente ora (giugno 2016) dispongo di impostazioni mie personali per gestire le tre varianti del greco e le tre varianti del latino, ma esse non sono ancora state ancora approvate dal gruppo di lavoro tex-hyphen del T<sub>E</sub>X Users Group (TUG) e non le posso distribuire senza la loro approvazione; quando saranno approvate, saranno direttamente disponibili con gli aggiornamenti di T<sub>E</sub>X Live.

Va inoltre notato che i due rami del test contengono le istruzioni per caricare pacchetti diversi e alcune definizioni, quelle dei loghi di  $\mathcal{G}\mathcal{I}\mathcal{T}$  e di METAPOST, che ricorrono a font vettoriali, ma non OpenType, per cui potrebbero dare luogo a problemi per la conformità alle norme PDF/A quando si usa X<sub>Ǝ</sub>L<sup>A</sup>T<sub>E</sub>X. Si spiegherà meglio nell'apposito paragrafo.

### 3.9.1 pdflatex e il formato PDF/A

Usando pdflatex è necessario usare il pacchetto specifico *pdfx* richiamandolo con l'opzione giusta; precisamente il preambolo deve contenere la richiesta:

```
\usepackage[a-1b]{pdfx}
```

come è stato mostrato precedentemente nella pagina 58; quando si usa pdflatex non è molto importante l'ordine con cui si caricano i vari pacchetti, ma la documentazione di *pdfx* raccomanda di caricarlo il più presto possibile; tuttavia sarebbe meglio caricarlo dopo aver specificato tutti i pacchetti aggiuntivi che potrebbero influire sui collegamenti ipertestuali, cioè i pacchetti che modificano i comandi di sezionamento, la composizione delle didascalie e simili. *pdfx* carica già *hyperref* per configurarne i collegamenti ipertestuali in modo conforme alle specifiche della norma PDF/A. Questo implica, però, che il pacchetto *pdfx* venga caricato prima che nel preambolo venga caricato *hyperref*; anzi, avendo caricato *pdfx* non è necessario caricare *hyperref*, ma basta configurarlo con `\hypersetup` per le esigenze specifiche del documento. Non è vietato caricarlo di nuovo, purché nel preambolo, dopo aver caricato *pdfx*, *hyperref* sia caricato senza opzioni proprio per evitare l'errore "option clash". Questa possibilità di tenere separato il caricamento di *hyperref* dalla sua configurazione mediante `\hypersetup`, è comoda perché consente di aggiungere la chiamata di *pdfx* (prima di *hyperref*) in un secondo tempo, quando si vuole aggiungere la compatibilità con le norme PDF/A.

Il pacchetto *pdfx* è già presente con ogni installazione completa del sistema T<sub>E</sub>X; se si disponesse di una installazione di base o, comunque, parziale e il pacchetto non fosse già installato, bisogna provvedere con i metodi specifici di ciascuna installazione. Se ne consiglia la lettura della documentazione, perché consente una certa dose di personalizzazione.

Il pacchetto *pdfx* con l'opzione *a-1b* provvede a quasi tutto il necessario, per esempio a definire un profilo di colore, ma non provvede ai metadati specifici di ogni particolare documento, di cui invece parlerò più avanti.

I metadati di carattere generale sono inseriti nel file di uscita dall'azione diretta del file `pdfx.sty`; ma per i metadati specifici del documento che si sta componendo è necessario predisporre nella cartella dove risiede il file principale della tesi (quello sul quale opera l'eseguibile `pdflatex`) un file contenente i metadati. Il metodo migliore per farlo è mostrato nel codice della pagina 58: si tratta di inserire i metadati all'interno dell'ambiente `file-contents*` prima ancora della dichiarazione della classe; questo ambiente  $\text{\LaTeX}$  è l'unica struttura del linguaggio di mak-up che può essere inserita prima di `\documentclass`.

Ci si ricordi solo che il pacchetto *toptesi* controlla solo se si sta usando `pdflatex` e assume che se non si sta componendo con `pdflatex` allora si sta usando `xelatex` oppure `lualatex`. In particolare si ottiene un errore se si compila con `latex`, il programma ancora esistente ma che io considero obsoleto, con il quale si produce un file in formato DVI, ormai quasi estinto. Si noti che, in effetti, sebbene il comando `latex` esista ancora, il motore di composizione di `latex` e di `pdflatex` è lo stesso, ma ne differiscono le impostazioni; il test `\ifPDFTeX` che usa TOPtesi per conoscere se si sta usando `pdfTeX` per produrre un file in formato PDF piuttosto che in formato DVI, non è detto che riesca a distinguere questi due casi. Non ho mai verificato, perché usando un calcolatore Apple, il mio formato di default è sempre PDF; per ottenere il formato DVI devo dare comandi specifici mediante il terminale, cosa che evidentemente non faccio mai perché, oltretutto, non dispongo di un visualizzatore per il formato DVI.

Se esistono validi motivi per non usare `pdflatex`, allora bisogna usare il semplice `latex`, e poi convertire il file DVI ottenuto in un file PS e, per la conversione in file PDF/A, bisogna poi usare `ghostscript`; si può procedere come indicato sotto nel paragrafo 3.9.3, indicando per la variabile `file1` il valore `$1.ps` invece del valore `$1.pdf`. È meglio evitare di usare quei pacchetti che richiedono comandi PostScript, come per esempio *PSTricks* o *XYpic*, ma è preferibile usare i pacchetti di grafica, come *pgf* con il suo modulo *tikz*, che consentono di fare quasi tutto quello che si può fare con gli altri pacchetti che richiedono il linguaggio PostScript. Inoltre esistono pacchetti da usare con `pdflatex` che permettono di usare (almeno) il potentissimo *PSTricks* anche con `pdflatex`; questi pacchetti, lavorando in background senza che l'utente debba metterci mano, estraggono il codice di *PSTricks* dal file sorgente, lo salvano in un nuovo file `.tex` di servizio che compilano con `latex` e ne trasformano il file di uscita in formato PDF per poi importarlo già scontornato nel file che interessa all'utente; Se fra una compilazione e l'altra non sono state eseguite modifiche al codice *PSTricks*, questa operazione in background viene tranquillamente saltata, quindi non si nota nemmeno nessun rallentamento nell'elaborazione della tesi. Vale comunque il consiglio di non usare questi moduli che richiedono l'uso del linguaggio PostScript, perché i moduli che lavorano direttamente con il linguaggio PDF lavorano altrettanto bene nella stragrande maggioranza dei casi.

Bisogna ricordare che produrre un file PDF/A non è semplice, ma se si usa `pdflatex` con i font di default, se si incorporano solo figure PNG (senza trasparenze) e JPEG con profilo di colore RGB (red, green, blue), se ogni file PDF da incorporare contiene al suo interno anche i font che eventualmente sono necessari per comporre il suo testo, non si

dovrebbero incontrare problemi di certificazione della natura PDF/A del file della tesi.

Più avanti c'è un paragrafo dedicato al caso che un file così prodotto non risulti conforme alla norma PDF/A.

### 3.9.2 lualatex e il formato PDF/A

Anche il programma lualatex può produrre direttamente file PDF/A conformi a partire dalla distribuzione del sistema T<sub>E</sub>X Live del 2016.

Con lualatex è importante specificare il preambolo come mostrato nella pagina 58 sia per quel che riguarda i metadati sia per quel che riguarda la posizione della chiamata del pacchetto *pdfx*.

Il programma lualatex è meno elastico rispetto al programma pdflatex per quel che riguarda la posizione della chiamata del pacchetto *pdfx* che deve essere assolutamente il primo dopo `\documentclass`.

Per il resto valgono le stesse indicazioni scritte per l'uso del programma pdflatex. Sottolineo ancora che il file sorgente della tesi deve essere assolutamente salvato con la codifica *utf8* e che i metadati devono essere scritti e salvati nel loro file *.xmpdata* con la stessa codifica.

Se i metadati devono essere usati da programmi di biblioteconomia per la ricerca dei contenuti, delle parole chiave, e simili, suggerirei di scrivere i metadati in inglese sia perché probabilmente la ricerca delle informazioni bibliotecomiche risulta facilitata anche all'estero, sia perché si evitano i caratteri accentati, almeno in tutti i campi in cui l'italiano non è essenziale, come il titolo (se è in italiano) e i nomi propri.

Come si legge nella parte di preambolo che appare nella pagina 58, questo stesso file è stato composto con lualatex ed la verifica della conformità ha dato esito positivo.

### 3.9.3 xelatex e il formato PDF/A

Con xelatex le cose sono un po' più complesse che con lualatex, ed è per questo motivo che consiglio di usare direttamente lualatex. Il motivo è dovuto al fatto che xelatex non produce direttamente la sua uscita nel formato PDF, ma in un formato DVI esteso che viene trasformato in PDF dal programma xdvipdfmx, che di per sé non sa nulla a proposito del fatto che il file da trasformare contenga o non contenga dei metadati.

Per questo motivo si deve compilare “a mano”, dando il comando di compilazione dal terminale nel modo seguente:

```
xelatex -shell-escape -output-driver=xdvipdfmx -z 0 <filename>.tex
```

Come si vede xelatex deve lavorare con l'opzione `-shell-escape` che consente di lanciare un programma esterno; in questo caso l'uscita di xelatex viene inviata al driver di uscita xdvipdfmx (e fin qui non ci sono sorprese rispetto a quello che xelatex fa di solito), ma la differenza consiste nell'opzione passata a xdvipdfmx: `-z 0`. Questa opzione dice al

programma di non comprimere il suo file di uscita; questo fatto produce due effetti: quello buono è che non comprime i metadati; quello “cattivo” è che il file di uscita è enormemente più grande rispetto a quando fosse compresso, anche 10 volte più grande.

Per il resto, avendo predisposto il main file della tesi come se si dovesse lavorare con `lualatex` e avendo osservato le stesse cautele per i font, i colori, eccetera, il file ottenuto, nonostante le sue dimensioni, può venire sottoposto a verifica attraverso il modulo Preflight di Adobe Acrobat Pro, come bisogna fare anche quando il file PDF fosse ottenuto con `pdflatex` o `lualatex`; ma il buono, questa volta, è che se il file ottenuto passa la verifica di conformità, Adobe Acrobat è anche in grado di comprimere il file, lasciando inalterati (non compressi) i metadati.

Vista questa trafila, si capisce anche perché sconsiglio di comporre la tesi con `xelatex` ma consiglio di usare direttamente `lualatex`.

### 3.9.4 Uso di `ghostscript` o di `pdftpages`

Esistono altri metodi per ottenere la produzione di file conformi alla norma PDF/A. Uno è quello di ottenere la trasformazione mediante il programma `ghostscript` e l'altro consiste nel produrre un secondo file importando le pagine del primo mediante il pacchetto `pdftpages`. Nel seguito si illustreranno i due metodi; anche se nel seguito continuerò ad esemplificare parlando della tesi, ritengo che questi due metodi siano più utili, per i lettori che leggendo questa documentazione dimostrano di voler usare il sistema  $\text{\TeX}$ , per controllare singoli file da importare nella tesi, sia in formato PDF sia, solo con `ghostscript`, in formato PS.

Infatti è meglio creare un piccolo file presumibilmente PDF/A compatibile e verificarne la compatibilità: se è compatibile, non ci sono problemi ad importarlo nella tesi; se è incompatibile prima si esaminano le cause dell'incompatibilità, prima lo si può correggere per renderlo compatibile o si possono prendere decisioni valide su cosa farne per importarlo nella tesi.

#### Uso di `ghostscript`

Dopo aver ottenuto un file PDF della tesi e ci si ritrova con un file che non è ancora conforme allo standard PDF/A, si può provvedere con altri programmi. Qui indicherò due o tre metodi che possono avere vantaggi o svantaggi; nessuno è perfetto, ma funzionano.

Si può usare `ghostscript` per trasformare un file PDF in un altro PDF/A. Bisogna predisporre nella stessa cartella dove si trova il main file del documento un file che contenga i metadati; per comporre questo documento composto con `xelatex`, il file adatto per `ghostscript` è il seguente:

```
%!  
% This is a sample prefix file for creating a PDF/A document.  
% Feel free to modify entries marked with "Customize".
```

```
% This assumes an ICC profile to reside in the file (srgb.icc),
% unless the user modifies the corresponding line below.

% Define entries in the document Info dictionary :
/ICCPROFILE (~/icc/srgb.icc) def          % Customize
def

[ /Title (La classe TOPtesi)              % Customize.
  /Author (Claudio Beccari)               % Customize.
  /Subject (File PDF ottenuto con XeLaTeX) % Customize.
  /DOCINFO pdfmark

% Define an ICC profile :

[/_objdef {icc_PDFA} /type /stream /OBJ pdfmark
[{icc_PDFA}
<<
  /N currentpagedevice /ProcessColorModel known {
    currentpagedevice /ProcessColorModel get dup /DeviceGray eq
    {pop 1} {
      /DeviceRGB eq
      {3}{4} ifelse
    } ifelse
  } {
    (ERROR, unable to determine ProcessColorModel) == flush
  } ifelse
>> /PUT pdfmark
[{icc_PDFA} ICCProfile (r) file /PUT pdfmark

% Define the output intent dictionary :

[/_objdef {OutputIntent_PDFA} /type /dict /OBJ pdfmark
[{OutputIntent_PDFA} <<
  /Type /OutputIntent          % Must be so (the standard requires).
  /S /GTS_PDFA1                % Must be so (the standard requires).
  /DestOutputProfile {icc_PDFA} % Must be so (see above).
  /OutputConditionIdentifier (sRGB) % Customize
>> /PUT pdfmark
[Catalog] <</OutputIntents [ {OutputIntent_PDFA} ]>> /PUT pdfmark
```

Questo file è formato dal nome del file .tex., (per esempio toptesi-it per questo file), agglutinato alla “desinenza” -def.ps, per un nome complessivo, per esempio, toptesi-it-def.ps. Ripeto: deve essere memorizzato nella stessa cartella dove si trova il main file del documento. Voglio far notare che ho usato questo file di metadati

per alcuni anni; oggi la libreria di ghostscript, versione 9.16, indica di modificare un file `PDFA_def.ps` diverso da quel modello, ma, devo ammetterlo, la documentazione di che cosa indicare nelle nuove parti configurabili è tutt'altro che chiara e, pur essendo possibile farlo, non ci sono riuscito.

Si noti ancora che i file `.icc` necessari per specificare i profili di colore devono risiedere in una cartella accessibile attraverso uno dei percorsi indicati nella variabile di sistema `PATH`; io per la mia macchina ho creato una cartella chiamata `icc`, inserita nella mia `$HOME`; nei sistemi UNIX il simbolo `~` indica questa 'casa' delle cartelle; nei sistemi Windows recenti, la `$HOME` non è rappresentata da un simbolo, ma corrisponde al nome dell'utente nella cartella `C:\Users\`; quindi la `$HOME` dell'utente Mario, sarà indicata con `C:\Users\Mario\`. Nella cartella `icc` (o comunque la si voglia chiamare) bisogna copiare i vari file `.icc` distribuiti insieme all'installazione di ghostscript; nelle macchine UNIX, questi file si trovano in una sottocartella di sistema che si chiama `/usr/local/share/ghostscript/⟨versione⟩/iccprofiles/`. Con la versione che sto usando `⟨versione⟩` vale 9.16 e contiene una decina di file che descrivono i profili di colore, compreso il file `srgb.icc` indicato nel precedente listato di codice PostScript (che quindi, se va bene per lo scopo, non necessita di nessuna personalizzazione).

Comunque per eseguire la conversione conviene predisporre una procedura `.bat` (per Windows) o `bash` per UNIX del tipo seguente (mostro la procedura `bash` che con minime modifiche si trasforma in una procedura `bat`) dal nome `pdf2pdfa`:

```
#!/bin/bash
file1=$1.pdf
file2=$1-a.pdf
file3=$1-def.ps
echo 'Running gs to pdfa'
# QUANTO SEGUE DEVE ESSERE SCRITTO SU UNA SOLA RIGA
gs -dPDFACompatibilityPolicy=1 -dPDFA=1 -dBATCH -dNOPAUSE
-sColorConversionStrategy=/RGB -sDEVICE=pdfwrite
-sOutputFile=$file2 $file3 $file1
```

Salvato il file in un ramo del `PATH` dove il sistema operativo può trovare i file (per esempio in `/bin/`) e assicuratici che sulle macchine UNIX il file abbia impostato il bit di eseguibilità, basta dare il comando nella finestra comandi:

`pdf2pdfa <Nome del main file senza estensione>`

e ghostscript si mette a lavorare finché produce in uscita un file PDF con il nome del file originario allungato con `-a`. A questo punto si può procedere con il software di verifica della natura PDF/A del file ottenuto. Se bisogna trasformare un file PS, basta cambiare la seconda riga `file1=$1.pdf` in `file1=$1.ps` senza la necessità di modificare altro.

Le variazioni per Windows consistono nel fatto che l'eseguibile di ghostscript si chiama `gswin32c` invece che `gs` e che i parametri di una macro si indicano con `%1`, `%2`, eccetera,

mentre quelli per UNIX si indicano con \$1, \$2, eccetera; in bash i commenti iniziano con il segno # mentre in bat essi cominciano con REM.

Nessuno vieta di arricchire la definizione delle due versioni della procedura per verificare l'esistenza del file PDF o PS da convertire, o l'esistenza del file dei metadati, con la possibilità di emettere dei messaggi cosicché l'utente possa regolarsi nel caso qualcosa risulti mancante. Tuttavia l'essenza dello script bash è quella indicata.

Va notato che i collegamenti ipertestuali all'interno del file convertito con ghostscript in un file conforme alla norma PDF/A vengono tutti disattivati; secondo me non è una grave perdita per un file da archiviare, ma certamente lo è per un documento da consultare a schermo.

### Uso di *pdfpages*

Un secondo metodo con il quale ho avuto risultati positivi ricorre al pacchetto *pdfpages* ma, come col metodo precedente, ha il difetto di perdere gli eventuali collegamenti ipertestuali interni.

Supponiamo quindi di disporre di un file PDF (ma non PS) composto secondo le raccomandazioni indicate nei paragrafi precedenti. Tanto per mettere i puntini sulle 'i', il lettore attento ha notato che le due parti del test indicato nella pagina 58 sono leggermente diverse, infatti per i loghi  $\mathbb{G}\mathbb{I}\mathbb{T}$  e METAPOST ho creato le definizioni per produrre questi loghi importando e scalando due file PDF appositamente predisposti (con pdf<sub>l</sub>atex), conformi alla norma PDF/A e opportunamente scontornati. Li ho predisposti con i loghi molto grandi, ma pur sempre vettoriali, in modo che possano venire ingranditi o rimpiccioliti a piacere. Le due macro \GuIT e \MP, infatti, scalano quelle due immagini PDF in modo che siano grandi come il testo circostante e abbiano la linea di base esattamente allineata con quella del testo. In quanto file PDF/A, essi contengono tutte le informazioni che devono contenere, metadati e font esattamente come la norma richiede; presi a sé, essi passano le verifiche eseguite con Adobe Acrobat Pro XI.

Si prepara ora un altro semplicissimo file che viene chiamato, per esempio, *toptesi-it-pages.tex* con il seguente contenuto:

```
% !TEX encoding = UTF-8 Unicode
% !TEX TS-program = pdflatex
\documentclass[a4paper]{report}
\usepackage{pdfpages}
\usepackage[a-1b]{pdfx}
\nofiles
\begin{document}\pagestyle{empty}
\includepdf[pages=-]{toptesi-it.pdf}
\end{document}
```

e lo si affianca con il corrispondente file `toptesi-it-pages.xmpdata` il cui contenuto è identico al file `toptesi-it.xmpdata` che si sarebbe preparato per comporre l'analogo file se lo si fosse potuto comporre con `pdflatex`.

Il piccolo file precedente è in grado di produrre un file PDF/A perché il suo preambolo è correttamente predisposto per questo scopo; carica il pacchetto `pdfpages` che serve per importare tutte o una selezione di pagine di un file PDF comunque composto, dentro il file che si ottiene elaborandolo con `pdflatex`. Se il file `toptesi-it.pdf` non contiene file importati con caratteristiche incompatibili, il file PDF ottenuto risulta conforme alla norma PDF/A, ma se il file di partenza conteneva collegamenti ipertestuali, il file finalmente ottenuto è privo di collegamenti ipertestuali.

Ho usato come esempio il nome di questo stesso file di documentazione; il lettore campeggia che si tratta solo di un esempio, perché questo documento si compila perfettamente in formato PDF/A senza ricorrere a questo procedimento; Ma, se per esempio, si fosse composta la tesi con un word processor e la si fosse esportata in formato PDF, questo procedimento potrebbe crearne la versione PDF/A compatibile senza sforzo e con la sola spesa di perdere i collegamenti ipertestuali interni.

Più comodamente i procedimenti descritti in questi ultimi due paragrafi possono servire per aggiungere quanto manca, per esempio, ad una figura PDF per verificarne la conformità oppure per scoprirne le cause di non conformità per eventualmente modificarla in modo che sia compatibile tanto da poterla includere nella tesi vera e propria. Per esperienza diretta io preferisco seguire il procedimento che ricorre al pacchetto `pdfpages`.

### **Non usare i due pacchetti `pax` e `pdfpages` assieme allo scopo di conservare i collegamenti ipertestuali**

Il pacchetto `pax` è stato costruito per ridare funzionalità ai collegamenti ipertestuali contenuti in file PDF immessi completamente o in parte dentro altri file PDF. Funziona perfettamente, ma il pacchetto `pdfx` non riesce a gestire le operazioni del pacchetto `pax`, per cui tutte le informazioni relative ai collegamenti vengono ripristinate, ma il file complessivo non rispetta le specifiche della norma PDF/A e gli errori segnalati dalla funzione Preflight di Adobe Acrobat Pro XI riguardano solo gli hyperlink, quando invece lo stesso file PDF ricomposto solo con l'uso di `pdfx` e `pdfpages` è perfettamente conforme con le norme PDF/A, anche se i suoi link non sono attivi.

## **3.10 Verifica della conformità**

Bisogna innanzi tutto disporre degli strumenti per verificare se un file è conforme alla norma PDF/A.

In rete ci sono siti dove viene eseguita la verifica della conformità di un file PDF con la norma del sottoformato PDF/A-1b, ma di solito l'esito della verifica è negativo, per lo



meno a me non è mai capitato che un file conforme venisse riconosciuto tale mediante le verifiche in rete.

Per verificare in modo serio la conformità ci sono due strade che non si escludono a vicenda.

1. Si cerca in dipartimento una stazione di lavoro dove un computer sia dotato del programma commerciale Adobe Acrobat Pro XI; spesso i dipartimenti dispongono di licenze multiple, oppure è possibile che un ricercatore dia una mano lasciando usare un suo calcolatore sul quale è montato il software indicato. Si procede poi come indicato qui di seguito.
2. La Adobe, come molte altre imprese produttrici di software, dispone di un programma *Education* che consente agli studenti e ai docenti di università e scuole secondarie superiori di acquistare i loro software a prezzi molto, molto vantaggiosi. Io come privato cittadino non mi sarei mai comprato l'Adobe Acrobat Pro XI se non avessi avuto la possibilità di avvantaggiarmi di questa offerta vantaggiosissima, ma come professore ho potuto farlo senza bisogno di ricorrere ad una versione fornitami dal Politecnico sotto una sua licenza multipla.

Disponendo dunque di Adobe Acrobat Pro XI, si apre il file PDF precedentemente caricato su una chiavetta, si va nella voce di menù Edit e si sceglie l'opzione Preflight; nella finestrina che si apre si sceglie la voce PDF/A compliance e sotto questa la voce Verify compliance with PDF/A-1b; poi si clicca sul bottone Analyse; dopo pochi secondi si apre una seconda finestrina dove appare il risultato dell'analisi. Se il risultato è positivo il file è definitivamente pronto come conforme alla norma PDF/A-1b.

Se invece l'analisi indicasse errori, bisogna provvedere a correggerli; è vero che nella stessa finestrina c'è anche il bottone Analyse and fix, ma l'operazione di "aggiustamento" di solito riesce a correggere errori molto veniali, quindi con i programmi di composizione del sistema T<sub>E</sub>X (col quale non si compiono errori veniali) questo aggiustamento è raro che abbia successo.

Bisogna avere pazienza e cercare di leggere con attenzione i messaggi di diagnostica che Preflight produce. Se mancano i metadati e/o il profilo di colore, significa che il file non è stato prodotto con le procedure descritte nei paragrafi precedenti.

Gli errori che restano sono di solito dei tipi seguenti.

1. Se è stato specificato un profilo di colore CMYK (celeste, lilla, giallo, nero) si è fatto un errore nello specificare il profilo di colore oppure si è inserita un'immagine a colori che è colorata con lo schema CMYK. Nel primo caso è bene verificare di avere specificato il profilo giusto. Nel secondo caso bisogna convertire il profilo di colore mediante qualsiasi programma di fotoritocco; segnalo il programma aperto e libero GIMP (GNU Image Management Program), disponibile per le tre piattaforme più comuni e i sistemi operativi Mac, Windows e Linux. Esso è in grado di

aprire immagini in “qualsiasi” formato bitmapped o vettoriale, di elaborarlo secondo i desideri dell’utente, per poi salvarlo in qualsiasi formato bitmapped, ma non nei formati vettoriali.

2. Alcuni font hanno glifi di larghezza nulla, fra questi ci sono almeno due glifi della famiglia matematica *cmsym*; *toptesi* provvede già da solo a sostituire questi glifi con altri simili o con disegni, in modo da correggere l’errore; tuttavia questi glifi potrebbero essere presenti in file importati.
3. Alcune figure sia raster (a cui siano state sovrapposte legende testuali con caratteri vettoriali) sia vettoriali (che contengono legende testuali) possono mancare dell’inclusione dei font; per cui quando queste immagini vengono immesse nel file PDF che si vorrebbe conforme con la norma, questa conformità viene a mancare. Se si sono prodotti personalmente tali file da includere, bisogna avere cura di far sì che tutti i glifi dei font usati nel file da includere siano essi stessi inclusi direttamente; esattamente com ho fatto con i loghi *qJr* e *METAPOST* di cui ho detto prima. Purtroppo questo è un caso molto frequente; quindi capita sovente di dover procedere alla correzione di questi file.

Per le figure con legende testuali, basta convertire l’intera figura nel formato *.png* (se contiene disegni al tratto e purché non contenga trasparenze) oppure *.jpg* se contiene colori sfumati) e il gioco è fatto.

Ma se si vuole mantenere la natura vettoriale si potrebbe usare il programma gratuito *Inkscape* che è in grado di aprire file vettoriali (PDF, EPS, PS) e di trasformarli in file con le estensioni vettoriali, avendo trasformato nel contempo i caratteri nelle loro outline, cioè nei disegni dei contorni dei glifi, riempiti dello stesso colore che avevano quei glifi nella figura originale. Spesso questa trasformazione va a buon fine.

Come esperienza personale ritengo che sia opportuno convertire ogni figura vettoriale con i glifi sostituiti dai loro contorni, solo nel formato EPS, che non gestisce le trasparenze, gestite, invece, dal formato *.pdf*. In questo modo i tre programmi che ci interessano, *pdflatex*, *xelatex* e *lualatex*, sono tutti e tre in grado di ricevere figure in formato *.eps*, e di incorporarlo nel file PDF in uscita. In questo modo si è sicuri di poter evitare le trasparenze, che sono vietate con il formato PDF/A-1b.

Come si vede, dunque, la produzione di un file conforme alla norma PDF/A non è una operazione semplice e senza disporre di Adobe Acrobat Pro XI si consuma molto tempo a ricorrere a postazioni di lavoro esterne, dovendo chiedere favori a laboratori o ad altre persone. Però se si osservano le indicazioni di questo paragrafo, è possibile che il file della tesi sia conforme alla norma fin dal primo momento. Questa stessa guida è stata prodotta con questo tipo di correzioni, ma io sono avvantaggiato perché dispongo personalmente del software necessario alla verifica.

# Capitolo 4

## I comandi specifici introdotti da TOPtesi

### 4.1 Introduzione

I comandi specifici introdotti da TOPtesi si aggiungono a tutti quelli definiti da  $\text{\LaTeX}$  e dalla sua classe standard *report*; mentre questi sono tutti in inglese o sono abbreviazioni inglesi, i comandi introdotti da TOPtesi sono prevalentemente in italiano o sono abbreviazioni italiane.

Questi comandi sono di diverse categorie; alcuni si possono usare solo in modo matematico altri solo in modo testo; alcuni solo nel preambolo, alcuni hanno senso solo durante la composizione del testo; alcuni servono solo per il frontespizio. Essi saranno descritti nei paragrafi seguenti.

### 4.2 Le opzioni

La classe *toptesi* accetta diverse opzioni nel comando di dichiarazione della classe; negli esempi acclusi al pacchetto ne sono state usate diverse, ma qui forse vale la pena di elencarle tutte in ordine, senza ripetere le opzioni già definite per la classe *report*.

`chapterbib` serve per specificare che si desidera la composizione della bibliografia alla fine di ogni capitolo; la bibliografia va composta a mano; se si desidera comporla con  $\text{\BibTeX}$  si invochi invece il pacchetto `chapterbib.sty` con il solito comando `\usepackage`. Si abbia cura di leggere attentamente la documentazione di quel pacchetto. Alternativamente si può usare il pacchetto *biblatex* (con la bibliografia da elaborare con il programma esterno *biber*); bisogna specificargli opportune opzioni e si possono comporre direttamente bibliografie distinte per capitoli; si legga con attenzione la documentazione di *biblatex*.

`classica` Serve per usare delle denominazioni un po' diverse dei comandi e per dare una forma diversa al loro contenuto; il frontespizio ne viene un poco modificato con un

look più adatto alle tesi in discipline classiche. Si veda la tabella 4.3 per i comandi disponibili quando questa opzione è attivata.

`cucitura` Serve per spostare il blocco del testo verso l'esterno quando si teme che la piegatura delle pagine verso il centro del fascicolo rilegato possa impedire la lettura agevole delle parole vicino al margine interno. Con legature eseguite bene, questa correzione non è necessaria; si ritiene che possa essere utile quando la tesi cartacea viene spillata.

`14pt` Aumenta la possibilità di scelta del corpo normale per la composizione della tesi; può contribuire a migliorare la lettura quando il testo è fitto di notazioni molto articolate, ma in generale il corpo così grande è un trucco per rimpolpare una tesi di modeste dimensioni; ne sconsiglio l'uso per questo secondo scopo. Avverto che con un corpo così grande è facile che certe righe risultino troppo lunghe e vadano a capo in punti non adeguati e/o producano righe non giustificate con gli avvisi "Overfull hbox"; questo è un altro motivo per il quale sconsiglio di usare questa opzione, se non quando è strettamente necessario. Se la si usa, si verifichi che il titolo, i titoli dei capitoli, quelli dei paragrafi, eccetera siano composti correttamente. In particolare il titolo della tesi e i titoli dei capitoli possono andare a capo solo prima di articoli, preposizioni semplici o articolate, congiunzioni e brevi avverbi come "non"; per evitare di andare a capo dopo queste parti del titolo si abbia l'accortezza di inserire il segno di legatura ~ fra queste brevi parole e le parole seguenti; per esempio, il titolo della tesi fornita come modello va scritto nella forma `La~pressione barometrica di~Giove`.

`autoretitolo` Questa opzione funziona solo se viene specificata anche l'opzione `classica`; se la si inserisce senza specificare `classica`, non succede nulla di male, semplicemente la classe informa di aver trovato delle opzioni che né lei né altri pacchetti hanno usato. Serve per comporre la testatina di sinistra sulle pagine pari con l'indicazione del candidato e del titolo della tesi. È ovvio che il titolo della tesi con questa opzione deve essere molto breve, ed è per questo che è stato messo a disposizione dello studente l'argomento facoltativo del comando `\titolo` che consente di specificare un titolo di tre o quattro parole (brevi) ma di senso compiuto, che possa sostituire il titolo normale, specialmente se questo è un po' lungo.

`oldstyle` Anche questa opzione funziona solamente insieme all'opzione `classica`; serve per scrivere i numeri delle pagine con le cifre minuscole o all'antica, cioè con segni di altezze e profondità diverse; si confronti `1234567890` con `1234567890`.

`pdfa` Prima della distribuzione di T<sub>E</sub>X Live 2016 il pacchetto `pdfx` doveva essere caricato dalla classe, non nel preambolo del documento come si può fare ora con la distribuzione aggiornata di T<sub>E</sub>X Live. Per compatibilità con il passato, l'opzione esiste ancora, ma produce solo un avviso e non carica `pdfx`.

`noTOPfront` e `usefrontespizio` sono opzioni equivalenti; servono per evitare il caricamento del pacchetto nativo di TOPtesi, *topfront*, in modo da poter comporre il frontespizio con pacchetti esterni, evitando conflitti con quei pacchetti.

## 4.3 Comandi di tipo generale

I comandi di tipo generale si possono usare in ogni contesto, in particolare alcuni sono fatti per essere usati sia in modo testo sia in modo matematico. Essi sono raccolti nella tabella 4.1.

Vale la pena di commentare sull'uso dell'ambiente *interlinea* e del comando `\interlinea`. Il primo confina il suo effetto all'interno dell'ambiente da lui stesso formato; il secondo agisce come una dichiarazione che resta in vigore finché una dichiarazione contraria non ne modifichi il valore.

**Tuttavia sia l'ambiente sia il comando non dovrebbero essere mai usati!** La composizione tipografica non ha nulla a che vedere con la composizione dattilografica. Quest'ultima si faceva con mezzi avanzati per l'epoca, ma oggi quei mezzi sono del tutto obsoleti; le poche macchine da scrivere meccaniche o elettromeccaniche che esistono ancora fuori da qualche museo, vengono usate per riempire formulari o compilare le informazioni sui documenti cartacei che sopravvivono alla invasione delle carte plastificate; ma tolti gli usi burocratici non mi viene in mente nessun altro uso degno di nota.

La composizione tipografica esige un perfetto equilibrio fra il corpo del font usato e la distanza fra due righe successive, distanza che prende il nome tecnico di *scartamento* o *avanzamento di riga*, ma spesso viene chiamato impropriamente *interlinea*; questo scartamento a seconda del font in uso può essere dal 10% al 20% maggiore del corpo del font usato.

Questa documentazione è scritta in corpo 12 pt e lo scartamento è di 14,5 pt; si dice che questo testo è composto in corpo 12/14,5.

L'*interlinea*, come suggerisce il nome, era originariamente lo spazio aggiuntivo da inserire fra una linea e l'altra; quando la composizione tipografica era eseguita con font ricavati da punzoni metallici, l'*interlinea* era la striscia di metallo che veniva interposta fra una riga di caratteri metallici e la successiva. Il comando `\interlinea` e l'ambiente corrispondente hanno pertanto dei nomi che si rifanno alla tipografia tradizionale, ma vengono usati come in dattilografia. In effetti l'argomento del comando e dell'ambiente serve solo come fattore moltiplicativo dello *scartamento*; porre questo fattore al valore 1.05 vuol dire moltiplicare lo scartamento per 1,05 portandolo quindi al valore di 15,5225 pt. Questo capoverso è composto con questo fattore impostato con l'ambiente *interlinea* e, nonostante si tratti solo di un aumento del 5% dello scartamento, l'occhio lo percepisce in modo più grande di quanto non faccia pensare il suo piccolo valore.

L'avanzamento di default scelto per i caratteri in uso è l'avanzamento otticamente ottimale; se si desidera usare un font diverso da quello di default, si potrebbe, per esempio,

Tabella 4.1. Comandi di tipo generale

Comando	Default	Scopo	Esempio d'uso
<code>\interlinea{...}</code>	1.0	Modifica l'argomento di <code>\linespread</code> <b>NON usare se non costretti con la forza!</b> Modo testo	<code>\interlinea{1.05}</code> oppure <code>\begin{interlinea}{1.05}</code> ... <code>\end{interlinea}</code> Vedi annotazioni sull'interlinea nel testo
<code>\ohm</code>		Omega “diritto” Modi testo e matematico	<code>45\ohm</code>
<code>\ped{...}</code>	nessuno	Pedice in tondo Modi testo e matematico	<code>\$V\ped{eff}\$</code>
<code>\unit{...}</code>	nessuno	Unità di misura in tondo unite al numero Modi testo e matematico	<code>15\unit{k\ohm}</code>
<code>\gei</code>		Unità immaginaria in tondo Solo modo matematico	<code>\$\eu^{\gei\omega t}\$</code>
<code>\eu</code>		Numero “e” in tondo Solo modo matematico	<code>\$\eu^{\gei\omega t}\$</code>
<code>\gradi</code>		circoletto alzato Modo testo e matematico	<code>27\unit{\gradi C}</code>
<code>\listing{...}</code>	nessuno	Listato di un programma in caratteri typewriter	<code>\listing{toptesi.tex}</code>
<code>\blankpagestyle{...}</code>	plain	Impostazione dello stile della pagina eventualmente emessa da <code>\cleardoublepage</code> quando deve passare ad una pagina dispari in composizione fronte retro	<code>\blankpagestyle{empty}</code>
<code>\ifTOPfront</code>	true	Viene impostato a false quando viene specificata l'opzione <code>noTOPfront</code> ; può essere usato per specificare azioni diverse a seconda del tipo di frontespizio da comporre	
<code>\goodpagebreak[...]</code>	4	Inserisce un fine pagina condizionale; l'argomento facoltativo serve per specificare il numero di righe necessarie prima della fine pagina condizionale	<code>\goodpagebreak[5]</code>

invocare il pacchetto *newpxtext* per usare il Palatino esteso come font di testo. Siccome questo font a pari corpo ha le minuscole più grandi di quelle dei font di default, potrebbe

essere una idea sensata quello di sperimentare con diversi valori del fattore di *interlinea*, ma poi si scoprirebbe che questo fattore differirebbe di pochi centesimi dall'unità e quindi ci sarebbe da domandarsi se ne valga la pena.

Se proprio si vuole stampare su carta delle bozze scritte abbastanza larghe per potervi inserire le correzioni e le annotazioni a mano, allora si imposti il fattore di *interlinea* al massimo a 1.5, ma quando stampate la bella copia, la versione finale, ricordatevi di reimpostare per *interlinea* il valore unitario di default.

Tra l'altro non si vuole mica usare l'espedito di un grande fattore di *interlinea* solo per rimpolpare una tesi dal volume modesto? Esso sarebbe un espediente talmente puerile che sarebbe scoperto al primo sguardo. Ricordate che alcune tesi svolte all'inizio degli anni 20 e non superavano le 30 pagine dattiloscritte o scritte a mano (!) e sulle quali si sta studiando ancora oggi dopo più di 80 anni!

Vale la pena di commentare il misterioso comando `\blankpagestyle`. Quando si compone fronte e retro senza usare l'opzione di classe `openright`, i capitoli vengono sempre aperti nelle pagine di destra, cioè nelle pagine dispari. Per fare questo il comando `\chapter` agisce eseguendo subito il comando `\cleardoublepage` che, fra le altre cose, controlla se la nuova pagina su cui scrivere il titolo del capitolo sia dispari. Se non lo fosse provocherebbe la stampa di una pagina "bianca"; bianca nel senso che non contiene testo ma contiene la testatina e il piedino.

Fino alla versione 0.62 di TOPtesi, questa eventuale pagina bianca prima dell'apertura di un nuovo capitolo conteneva la testatina e il piedino; d'accordo nel piedino c'è solo il numero della pagina, e ci può stare, ma nella testatina rimaneva il titolo del capitolo precedente: tipograficamente molto antiestetico ed errato. Ora di default questa pagina bianca viene composta solo con il piedino che, ricordiamo, contiene solo il numero della pagina. Alcuni preferiscono che questa pagina bianca sia completamente bianca e non contenga nemmeno il piedino; ecco, in questo caso basta specificare:

```
\blankpagestyle{empty}
```

e il problema è risolto. Volendo si potrebbe specificare il nome di qualunque altro stile di pagina già definito ma, ad essere franchi, non vedo alternative fra i due stili `plain` e `empty` rispetto agli altri stili che contengono sempre le testatine, eventualmente ridotte al solo loro filetto.

Se lo si vuole usare, consiglio di farlo subito dopo `\begin{document}`.

Se invece si desidera lasciare il valore di default a `plain` e usare saltuariamente un altro stile, si usi esplicitamente il comando `\clearadoublepage` specificando lo stile desiderato come argomento facoltativo, per esempio:

```
\clearadoublepage[empty]
```

Riassumendo: per una pagina veramente vuota sempre e in ogni caso; si specifichi

```
\blankpagestyle{empty}
```

subito dopo `\begin{document}`.

Se si desidera una pagina vuota saltuariamente si espliciti `\cleardoublepage` con l'argomento facoltativo `empty`. Se si desidera modificare lo stile della pagina bianca quando il comando `\cleardoublepage` è emesso da uno specifico comando `\part` o uno specifico comando `\chapter` si usi `\cleardoublepage[empty]` *subito prima* di quel comando di sezionamento.

Qualche annotazione va fatta per il comando `\goodpagebreak`. Il nucleo di  $\text{\LaTeX}$  contiene il comando `\goodbreak` che dovrebbe impostare un consenso di eseguire un fine pagina impostando una penalità molto piccola negativa, più o meno simile a quello che fa il comando `\pagebreak[1]`. In realtà questo comando nativo di Plain  $\text{\TeX}$ , conservato in  $\text{\LaTeX}$ , di fatto non funziona come ci si aspetterebbe. Il comando di questa classe, `\goodpagebreak` procede in un altro modo, anche perché si comporta in modo diverso se viene specificato in modo verticale o in modo orizzontale.

Infatti in modo orizzontale inserisce incondizionatamente un comando `\vadjust` il cui argomento vale esattamente `\newpage` incondizionato quando la *riga* che lo contiene viene scritta nel file di uscita; si dovrà porre un minimo di attenzione a fare sì che lo spazio alla fine della macro non elimini spazi interparola, né introduca spazi spuri; quando lo si usa, quindi, sarà opportuno lasciare uno spazio prima e dopo la macro, per esempio

`...prima_\goodpagebreak_di...`

Vale la pena di usare questo comando in certi casi in cui la bozza mostra un salto pagina prima dell'ultima riga di un capoverso, cioè quando viene prodotta una riga vedova; quando si corregge la bozza si cerca il testo della riga prima di quella vedova e si inserisce la macro fra due parole qualsiasi di quel testo. È evidente che se si modifica il capoverso e non c'è più bisogno del salto pagina incondizionato, bisogna cancellare la macro da quella posizione.

In modo verticale, invece, viene calcolata la differenza fra le lunghezze `\pagegoal` e `\pagetotal`; la prima corrisponde più o meno a `\textheight` diminuita dello spazio riservato alle note in calce e del loro spazio di separazione dal testo; è insomma l'obiettivo per raggiungere la giusta altezza del testo; la seconda lunghezza è la lunghezza netta del testo raggiunta fino a quel momento; se questa differenza è *non maggiore* (che per la sintassi nativa di  $\text{\TeX}$  vuol dire *minore o uguale* al numero di righe specificate di default (4 righe) o specificate mediante l'argomento facoltativo, allora viene inserito un salto pagina incondizionato. Vale la pena servirsi di questo comando in modo verticale quando un oggetto relativamente grande e indivisibile viene rinviato alla pagina successiva, così da obbligare  $\text{\LaTeX}$  a stiracchiare lo spazio elastico verticale al fine di giustificare la pagina che precede il salto pagina.

Questa spiegazione contorta descrive una situazione piuttosto frequente che si manifesta prima dei titoli di sezioni, sottosezioni, e simili. Infatti ogni titolo è separato da un certo spazio dal testo che inizia la sezione e siccome le righe orfane sono tipograficamente molto sgradevoli (meno delle righe vedove, ma comunque sgradevoli), il titolo della sezione



deve essere seguito da almeno due righe di testo. Quindi il titolo della sezione, lo spazio verticale di separazione e le due righe di testo formano un blocco di almeno 4 righe; se non ci stanno in fondo alla pagina,  $\text{\LaTeX}$  manda tutto questo blocco alla pagina successiva, stiracchiando gli spazi verticali della pagina corrente. Mettendo la macro `\goodpagebreak` prima del titolo di quella sezione che viene rinviata alla pagina successiva, si produce una pagina mozza, ma non troppo, e senza spazi stiracchiati, seguita da una pagina che inizia con il titolo della nuova sezione; ecco come fare.

```
... fine dell'ultimo capoverso.
```

```
\goodpagebreak
```

```
\section{...}
```

L'algoritmo non è infallibile, ma funziona quasi sempre; può talvolta succedere che il fine pagina produca una pagina quasi vuota e la nuova sezione cominci dopo questa pagina; questo inconveniente è causato dal modo asincrono con cui vengono composte le pagine del file di uscita rispetto alla sequenza di capoversi che creano la bozza non impaginata; in sostanza il funzionamento asincrono fra la composizione del testo, il *paragraph builder* e il *page builder*. La cosa è difficile da spiegare in questa documentazione e la si rimanda alla guida tematica *Il  $\text{\LaTeX}$  Reference Manual commentato* liberamente scaricabile dal sito [www.guitex.org](http://www.guitex.org). Ma in questi rari casi la soluzione consiste nello spostare `\goodpagebreak` in modo orizzontale fra le parole della penultima riga prima di questo salto pagina mal gestito. Il lettore non si scoraggi: questi sono ritocchi destinati a rifinire la tesi quando la si è sostanzialmente completata. In questa documentazione non ho mai usato `\goodpagebreak` in modo orizzontale e l'ho usato in modo verticale solo due tre volte.

## 4.4 Comandi per il frontespizio

Questo paragrafo descrive i comandi per comporre il frontespizio con il pacchetto *topfront*, nativo della classe *toptesi*. Se si vuole comporre il frontespizio con un altro pacchetto, non si usino questi comandi ma quelli descritti nella documentazione del pacchetto alternativo. Naturalmente si sarà specificata l'opzione `noTOPfront` fra le opzioni della classe, al fine di evitare conflitti con i pacchetti esterni.

I comandi per la composizione del frontespizio *non possono* essere inseriti nel preambolo, cioè prima di `\begin{document}`, qualunque sia il programma di compilazione usato, perché nessuno di quei comandi è definito prima di che il modulo *topfront* sia stato eventualmente caricato, il che avviene solo al momento di eseguire il comando `\begin{document}`; se si specifica l'opzione `noTOPfront` il modulo *topfront* non viene nemmeno caricato.

È per questo che si consiglia di usare facoltativamente il file di configurazione e comunque tutti i comandi necessari per il frontespizio, non presenti nel file di configurazione, all'interno dell'ambiente *frontespizio* o *frontespizio\**. In questo modo non si corre il rischio di inserirli nel preambolo. Se poi si usa un pacchetto esterno per comporre il frontespizio, a maggior ragione bisogna specificare i comandi necessari come specificato nella documentazione di quel pacchetto esterno.

Con le variazioni di TOPtesi a partire dalla versione 5.85, lo si ripete, la cosa migliore da fare è quella di usare questi comandi dentro uno degli ambienti *frontespizio* o *frontespizio\**; si pigliano due piccioni con una fava; siccome questi ambienti compongono direttamente il frontespizio, essi devono trovarsi dopo `\begin{document}` e questo va bene sia per pdfL<sup>A</sup>T<sub>E</sub>X sia per LuaL<sup>A</sup>T<sub>E</sub>X.

I comandi di questa sezione possono essere introdotti in un ordine qualunque, ma è più chiaro se sono introdotti nell'ordine in cui sono elencati nelle tabelle 4.4-4.5 riportate dalla pagina 85 alla pagina 88.

Essi possono anche essere scritti nel file di configurazione `\mainfile.cfg`, anzi, direi che è meglio che siano inseriti in quel file, lasciando nel preambolo solo quelli che cambiano i valori di default o che inseriscono informazioni non presenti di default.

Alcuni comandi sono generali; altri si riferiscono specificatamente alla monografia di laurea, o alla tesi laurea, o alla dissertazione di dottorato.

La differenza è che generalmente la monografia di laurea non ha un relatore; se nella vostra facoltà anche la monografia di laurea ha un relatore comportatevi come per la tesi di laurea. Se nel vostro ateneo o nella vostra facoltà succede così, allora inserite nel file di configurazione il seguente comando

```
\TesiDiLaurea{Tesi di Laurea}
```

Per la tesi del secondo ciclo aggiungete dopo la parola “laurea” l’indicazione “magistrale” (oppure “specialistica”, se così è l’uso nel vostro ateneo).

Similmente cambiate le stringhe inserite dai vari comandi facendo ampio uso di quelli esposti nelle tabelle 4.4–4.5; in particolare consiglieri di inserirli nel file di configurazione, specialmente se devono rappresentare delle modifiche “permanenti”. **Usateli anche per predisporre quelle stringhe in una lingua diversa dall’italiano.**

Per comporre questo testo si è usata la classe *toptesi*, e si è supposto di scrivere una monografia; tuttavia alcune delle stringhe di default non sono adeguate a questo uso, quindi il preambolo di questo documento contiene le seguenti specificazioni che illustrano l’impiego di diversi fra i comandi descritti nelle tabelle citate<sup>1</sup>:

```
\begin{frontespizio}
```

---

<sup>1</sup> In verità ho imbrogliato un poco. I comandi qui elencati producono un frontespizio molto simile a quello effettivamente presente in questo documento, ma non esattamente identico; in fondo questo documento non è una vera monografia di laurea, quindi mi sono permesso qualche lieve “abbellimento”.

```
\NomeMonografia{Manuale d'uso}
\monografia{La classe \textsf{TOPtesi}}
\sottotitolo{Per comporre le tesi al Poli\\
             e in molte altre università}
\candidato{Claudio Beccari}
\sedutadilaurea{Versione \fileversion}
\ateneo{}% senza il nome che è già contenuto nel logo
\logosedede{logo_blu}
\end{frontespizio}
```

Vale la pena di fare le seguenti osservazioni.

1. Se si usa l'ambiente *frontespizio* o il comando `\frontespizio` il logo viene messo in testa; se si usa la forma asteriscatadel comando o l'ambiente *frontespizio\** il logo viene messo nella metà inferiore della pagina del titolo. Se il logo dell'ateneo contiene per disteso il logotipo, cioè il nome per disteso, allora l'uso del comando `\ateneo` contenente il nome generico dell'ateneo implica una ridondanza di informazioni che potrebbe anche essere antiestetica; quindi se si usa il comando `\ateneo` sarebbe meglio usare l'ambiente *frontespizio\**, e viceversa. Ma questa possibilità di scelta per la posizione del logo e la presenza del nome dell'ateneo consente di rispettare le diverse specifiche indicate dalle segreterie didattiche degli atenei; si veda la tabella riassuntiva [3.1](#) nella pagina [30](#).
2. `\titolo` non deve essere usato per la monografia (ma `\sottotitolo` si può usare, come si è fatto per il frontespizio di questo manuale); per la monografia il comando `\monografia` imposta sia il titolo sia le altre informazioni che distinguono la monografia da una tesi di laurea magistrale.
3. Il comando `\titolo` accetta un argomento facoltativo, la versione breve del titolo del frontespizio; serve con l'opzione classica e l'opzione autoretitolo, dove il titolo breve va nella testatina delle pagine; se il titolo del frontespizio fosse troppo lungo la testatina verrebbe molto male!
4. Non dare i comandi `\relatore`, `\secondorelatore` e `\terzorelatore` né per la dissertazione di dottorato né per la monografia.
5. L'indicazione della materia su cui si svolge la tesi di laurea o di dottorato non viene normalmente indicata se non, talvolta, nelle facoltà umanistiche.
6. I secondi e terzi candidati non hanno senso né per le tesi di dottorato né per le monografie. Per tutti i candidati il comando per inserirne il nominativo accetta due forme:

```
\candidato{\Nome Cognome}
```

 <p>POLITECNICO DI TORINO</p> <p>Corso di Laurea in Ingegneria delle Telecomunicazioni</p> <p>Tesi di Laurea Magistrale</p> <p><b>Titolo della tesi di laurea magistrale</b></p> <p>Sottotitolo della tesi di laurea magistrale</p> <p><b>Relatore</b> prof. Enrico Rosa</p> <p><b>Candidate</b> Susanna Rossi Laura Bruni</p> <p>Dicembre 2011</p>	 <p>UNIVERSITÀ DI FERMIA SCUOLA DI DOTTORATO Dottorato in Meccanica – XIII ciclo</p> <p>Tesi di Dottorato</p> <p><b>Titolo della tesi di dottorato</b></p> <p>Sottotitolo della tesi di dottorato</p> <p><b>Maria Neri</b></p> <p><b>Tutore</b> prof. Alberto Aranci</p> <p><b>Coordinatore del corso di dottorato</b> prof. Giorgio Bigi</p> <p>Luglio 2011</p>
 <p>UNIVERSITÀ DI MARCONIA E DI EINSTEINIA Corso di Laurea in Ingegneria Biomedica</p> <p>Tesi di laurea</p> <p><b>Titolo della tesi di laurea</b></p> <p>Sottotitolo della tesi di laurea</p> <p><b>Relatori</b> prof. Mario Rossi Ing. Pietro Verdi</p> <p><b>Candidati</b> Giovanni Bianchi matricola: 654321 Laura Azzurri matricola: 987654</p> <p>Ottobre 2011</p>	 <p>SCUOLA NAZIONALE SUPERIORE Corso di Laurea in Energetica</p> <p>Monografia di Laurea</p> <p><b>Titolo della monografia</b></p> <p>Sottotitolo della monografia</p> <p>ANNA BIANCHI</p> <p>Settembre 2011</p>

Figura 4.1. I quattro frontespizi fondamentali

oppure

`\candidato{\langle Nome Cognome \IDN numero di matricola \rangle}`

Il `\langle numero di matricola \rangle` non è quasi mai necessario, quindi la sequenza `\IDN` viene inserita solo se si vuole indicare anche il numero di matricola. La macro `\IDN` è predefinita per contenere la stringa `\\quad matricola:\space`; può venire ridefinita per contenere il nome dell'*Identification number* oppure semplicemente *ID*, e i due punti prima del comando `\space` potrebbero essere sostituiti con qualche altro segno o semplicemente eliminati. Ovviamente questo è possibile anche per il comando `\candidata`.

7. Per la monografia l'informazione della data può essere omessa se non c'è una data per la presentazione.
8. Il pacchetto *TOPtesi* inizialmente veniva distribuito con un certo numero di loghi di università italiane e straniere. La politica di *T<sub>E</sub>X Live* è quella di distribuire solo materiale con licenza libera e incondizionata; i loghi delle varie università certamente non lo sono, quindi i loghi non vengono più distribuiti con *toptesi*. Ogni laureando è quindi tenuto a chiedere il logo alla sua università accettandone tutte le limitazioni d'uso che l'università potrebbe imporgli.

Per quanto riguarda i loghi, la loro inserzione implica l'uso del pacchetto *graphicx*, che *toptesi* carica di default; specificare nuovamente quel pacchetto non produce nessun danno, perché *L<sup>A</sup>T<sub>E</sub>X* controlla da solo se lo deve caricare o non lo deve ricaricare. Però, evidentemente, lo si è già detto all'inizio, non è bene caricare una seconda volta i pacchetti che sono già stati caricati

Per i formati dei file grafici da includere, le versioni recenti di *pdfL<sup>A</sup>T<sub>E</sub>X*, di *X<sub>Y</sub>L<sup>A</sup>T<sub>E</sub>X* e di *LuaL<sup>A</sup>T<sub>E</sub>X* accettano i formati PDF, PNG, JPG, MPS, EPS. La cosa è scritta a chiare lettere nella guida *grfguide.pdf* (già presente nella propria installazione del sistema *T<sub>E</sub>X* aggiornata e completa), ma è un fatto che viene dimenticato troppo spesso e che obbliga a cercare aiuto da chi ne sa di più, compagni o professori; ma che anche loro talvolta dimenticano.

Esistono programmi per passare da un formato all'altro, ma anche disponendo di quei programmi e conoscendo i vantaggi e gli svantaggi di un formato rispetto ad un altro, questa operazione viene dimenticata troppo spesso, perché troppo spesso ci si dimentica di queste limitazioni sui formati.

Per questo io preferisco usare sempre e solamente i compilatori *pdfL<sup>A</sup>T<sub>E</sub>X*, *X<sub>Y</sub>L<sup>A</sup>T<sub>E</sub>X* o *LuaL<sup>A</sup>T<sub>E</sub>X*. L'unico inconveniente è che non posso usare direttamente il pacchetto *PSTricks*; ma finora me la sono cavata molto bene anche senza le prestazioni di questo bellissimo pacchetto. D'altra parte esistono pacchetti che consentono di usare *PSTricks* anche con *pdfL<sup>A</sup>T<sub>E</sub>X* e gli altri programmi; l'operazione diventa un po' più complicata ma se ne occupa direttamente il programma di tipocomposizione usato; l'utente deve solo caricare uno di

quei pacchetti e seguirne le indicazioni specificate nella sua documentazione (vedi per esempio il pacchetto *pdftricks* già presente in ogni installazione aggiornata e completa di T<sub>E</sub>X Live; documentazione: `texdoc pdftricks`).

## 4.5 Altri comandi

La classe *toptesi* richiede l'uso di altri comandi il più notevole dei quali è senza dubbio `\frontespizio` e la sua variante `\frontespizio*` von i rispettivi ambienti; si veda la tabella riassuntiva 3.1 nella pagina 30). È un comando che non vuole argomenti e che va dato all'inizio, subito dopo l'apertura del documento o all'inizio del primo file incluso dal master file. Si veda comunque la tabella 4.2. Ricordo che i comandi `\frontespizio` e `\frontespizio*` sono definiti dal pacchetto *topfront*; se si vuole comporre il frontespizio con un pacchetto esterno, si è specificata l'opzione `noTOPfront`, che inibisce il caricamento del pacchetto, perciò questi comandi non sono più disponibili.

La differenza fra i due comandi consiste nel fatto che `\frontespizio` compone il frontespizio della tesi come richiesto dal Politecnico di Torino, con il logo dell'ateneo in testa alla pagina. Invece la variante asteriscata compone il frontespizio scrivendo in testa il nome dell'ateneo e mettendo il logo nella metà inferiore della gabbia di stampa. Il risultato si vede chiaramente nella figura 4.1.

Merita sottolineare che il formato richiesto dal Politecnico di Torino va bene nel senso che il logo di quell'ateneo contiene già il suo logotipo<sup>2</sup>. Sarebbe quanto mai inestetico se in testa al frontespizio comparisse due volte il nome del Politecnico; quindi non è strano che l'intestazione della tesi contenga formalmente solo il logo. Lo stesso potrebbe essere fatto, per esempio, con i loghi di diverse università, per esempio quello dell'Università di Torino, di Ca' Foscari a Venezia, di Bologna, dell'Università Cattolica, di Padova, oltre al logo del Politecnico di Milano, giusto per citarne alcuni, perché contengono il marchio e il logotipo.

Per le università che hanno un nome proprio, come per esempio *Tor Vergata* per la seconda università di Roma ma, che io sappia, non hanno il logotipo nel loro logo, sarebbe quanto mai inestetico usare il logo sopra, in testa, poi il nome proprio, poi tutto il resto; io non consiglierei mai di usare `\frontespizio` in questi casi, ma userei solo `\frontespizio*` o l'ambiente corrispondente.

Nella figura 4.1 nella pagina 80 sono riportati quattro esempi di frontespizi relativi ai tipi di tesi seguenti:

- tesi di laurea specialistica o magistrale,

---

<sup>2</sup>Il logo del Politecnico di Torino è formato da due parti accostate con un piccolo spazio e un filetto verticale fra le due; nella parte di sinistra compare il marchio dell'ateneo, il solito simbolo tondo, mentre la parte di destra contiene il logotipo, cioè il nome POLITECNICO DI TORINO su due righe scritto con certi font e con un colore particolare; font e colore non possono venire modificati, pena la violazione d'uso del logo.

- dissertazione di dottorato
- tesi di laurea del Vecchio Ordinamento,
- monografia di laurea,

Con questi modelli lo studente che compila la sua tesi usando TOPtesi può scegliere la versione che fa al caso suo e sa anche che cosa deve configurare per eseguire alcuni cambiamenti.

Tabella 4.2. Comandi ulteriori di *toptesi*

Comando o ambiente	Default	Scopo
<code>\figurespagetrue</code>	<code>\figurespagefalse</code>	Fa o non fa stampare l'indice delle figure
<code>\tablespagetrue</code>	<code>\tablespagefalse</code>	Fa o non fa stampare l'indice delle tabelle
<code>frontespizio</code>		Contiene i dati per il frontespizio e l'eventuale retrofrontespizio e lo compone con i loghi nella testatina
<code>frontespizio*</code>		Contiene i dati per il frontespizio e l'eventuale retrofrontespizio e lo compone con il loghi nella parte bassa della pagina
<code>\frontespizio</code>	nessuno	Fa stampare il frontespizio ed eventualmente il retrofrontespizio; il logo dell'ateneo appare in testa alla pagina
<code>\frontespizio*</code>	nessuno	Fa stampare il frontespizio ed eventualmente il retrofrontespizio; il logo dell'ateneo appare in basso nella pagina; vedi la tabella riassuntiva <a href="#">3.1</a>
<code>\sommario</code>	nessuno	Inizia un capitolo non numerato che ha per intestazione la parola SOMMARIO (anche in lingua)
<code>\ringraziamenti</code>	nessuno	Inizia un capitolo non numerato che ha per intestazione la parola RINGRAZIAMENTI (anche in lingua)
<code>\indici</code>	nessuno	Fa stampare l'indice generale, e, se sono stati dati i comandi <code>\figurespagetrue</code> e <code>\tablespagetrue</code> , anche gli indici delle figure e delle tabelle.
<code>\paginavuota</code>	nessuno	Emette nel file di uscita una pagina totalmente bianca, senza nemmeno il numero della pagina.
<code>\NoteWhiteLine</code>	nessuno	Anche questo comando richiama a gran voce e serve per mettere in nota una riga bianca.

Tabella 4.3. Comandi ulteriori per il frontespizio e per il corpo della tesi definiti con l'opzione `classica`

Comando	Scopo
<code>\candidato</code>	Si usa come di solito ma con l'opzione <code>classica</code> produce la scrittura “Laureando” invece che “Candidato”.
<code>\candidata</code>	Come sopra al femminile. Sia questo comando, sia il precedente accettano che l'argomento sia scritto nella forma $\langle \text{Nome Cognome} \backslash \text{IDN matricola} \rangle$ in alternativa a $\langle \text{Nome Cognome} \rangle$ .
<code>\tomo</code>	Esegue i frontespizi successivi di una tesi divisa in tomi scrivendovi Tomo primo, Tomo secondo, eccetera, a seconda del numero progressivo dei volumi in cui è suddivisa la tesi (massimo quattro). Questo comando viene usato <i>al posto</i> di <code>\frontespizio</code>
<code>\annoaccademico</code>	Il suo argomento può essere un anno solare o due anni separati da una lineetta. Viene scritto nel frontespizio della tesi o del singolo tomo con la specificazione che si tratta dell'anno accademico e non della data della presentazione della tesi.
<code>\EnDash</code>	Produce una lineetta lunga come —, ma ribassata in modo che stia bene fra numeri di stile antico.
<code>\nota[...]</code>	Serve per comporre una nota senza ricorrere al contatore numerico di default. Il simbolo con cui viene richiamata di default è l'asterisco, ma si può mettere qualunque segno matematico senza esplicitare i segni di dollaro, per esempio si può scrivere <code>\nota[\dagger]{Questa nota ...}</code>
<i>dedica</i>	È un ambiente con cui si può stampare una pagina di dedica; generalmente questa pagina viene dopo il frontespizio.
<i>citazioni</i>	È un ambiente che consente di scrivere una pagina con frasi argute. L'arguzia dipende dall'autore; spesso nei libri, raramente nelle tesi, l'autore cita frasi celebri o che in qualche modo hanno a che fare con il contenuto del testo.



Tabella 4.4. Comandi per il frontespizio della monografia di laurea, della tesi di laurea e della dissertazione di dottorato (*continua*)

Comando	Default	Scopo	Esempio d'uso
<code>\ateneo{...}</code>	POLITECNICO DI TORINO	Definisce il nome generico dell'Ate- neo	<code>\ateneo{II Università di Roma}</code>
<code>\nomeateneo{...}</code>	nessuno	Definisce il nome proprio dell'Ate- neo	<code>\nomeateneo{Tor Vergata}</code>
<code>\facolta[...]{...}</code>	nessuno	Definisce il no- me della Facol- tà opzionalmente con l'indicazione dell'ordinale	<code>\facolta[III]% {Scienze}</code>
<code>\struttura</code>		sinonimo del co- mando <code>\facolta</code>	
<code>\corsodilaurea{...}</code>	nessuno	Definire il nome del corso di laurea	<code>\corsodilaurea% {Fisica}</code>
<code>\corsodidottorato% {...}</code>	nessuno	Definisce il nome del corso di dotto- rato	<code>\corsodidottorato% {Fisica}</code>
<code>\monografia{...}</code>	nessuno	Definisce il titolo della monografia e imposta lo stile del frontespizio	<code>\monografia% {Il teorema di Eulero}</code>
<code>\titolo[...]{...}</code>	nessuno	Definisce il titolo della tesi o della dissertazione	<code>\titolo% [La pressione]% {La pressione di Giove}</code>
<code>\sottotitolo{...}</code>	nessuno	Definisce il sotto- titolo della tesi o della dissertazio- ne	<code>\sottotitolo{Metodo Barometrico}</code>
<code>\Materia{...}</code>	nessuno	Definisce la mate- ria su cui verte la tesi	<code>\Materia{Remote Sensing}</code>
<code>\materia{...}</code>	nessuno	sinonimo del co- mando <code>\Materia</code>	<code>\materia% {Letteratura ostrogota}</code>

Tabella 4.4. Comandi per il frontespizio della monografia di laurea, della tesi di laurea e della dissertazione di dottorato (*continua*)

Comando	Default	Scopo	Esempio d'uso
<code>\relatore{...}</code>	nessuno	Definisce il nome del relatore	<code>\relatore%</code> <code>{prof.~Albert</code> <code>Einstein}</code>
<code>\secondorelatore{...}</code>	nessuno	Se c'è, definisce il nome del secondo relatore	<code>\secondorelatore%</code> <code>{dott.~Grazia</code> <code>Deledda}</code>
<code>\terzorelatore{...}</code>	nessuno	Se c'è, definisce il nome del terzo relatore	<code>\terzorelatore%</code> <code>{ing.~Thomas</code> <code>A.~Edison}</code>
<code>\direttore{...}</code>	nessuno	Definisce il nome del direttore del ciclo di dottorato	<code>\direttore%</code> <code>{prof.~Albert</code> <code>Enstein}</code>
<code>\coordinatore{...}</code>	nessuno	Definisce il nome del coordinatore del ciclo di dottorato	<code>\coordinatore%</code> <code>{prof.~Albert</code> <code>Einstein}</code>
<code>\QualificaDirettore%</code> <code>{...}</code>	Direttore o Coordinatore	Definisce il titolo da inserire prima del nome del direttore della scuola di dottorato	<code>\QualificaDirettore%</code> <code>{PhD Director}</code>
<code>\tutore{...}</code>	nessuno	Definisce il nome del tutore	<code>\tutore{prof.~Karl</code> <code>Von Braun}</code>
<code>\tutoreaziendale{...}</code>	nessuno	Definisce la qualifica del tutore aziendale	<code>\tutoreaziendale%</code> <code>{Supervisore}</code>
<code>\candidato{...}</code>	nessuno	Definisce il nome del candidato	<code>\candidato{Galileo</code> <code>Galilei}</code>
<code>\candidata{...}</code>	nessuno	Definisce il nome della candidata	<code>\candidata{Maria</code> <code>Curie}</code>
<code>\secondocandidato%</code> <code>{...}</code>	nessuno	Se c'è, definisce il nome del secondo candidato	<code>\secondocandidato%</code> <code>{Evangelista</code> <code>Torricelli}</code>
<code>\secondacandidata%</code> <code>{...}</code>	nessuno	Se c'è, definisce il nome della seconda candidata	<code>\secondacandidata%</code> <code>{Rita Levi</code> <code>Montalcini}</code>

Tabella 4.4. Comandi per il frontespizio della monografia di laurea, della tesi di laurea e della dissertazione di dottorato. Comando per comporre il verso del frontespizio.

Comando	Default	Scopo	Esempio d'uso
<code>\terzocandidato{...}</code>	nessuno	Se c'è, definisce il nome del terzo candidato	<code>\terzocandidato%{Alessandro Volta}</code>
<code>\terzacandidata{...}</code>	nessuno	Se c'è, definisce il nome della terza candidata	<code>\terzacandidata%{Eleonora Duse}</code>
<code>\sedutadilaurea{...}</code>	data corrente	Definisce il mese e l'anno (volendo il giorno) della seduta di laurea	<code>\sedutadilaurea%{Dicembre 2025}</code>
<code>\esamedidottorato%{...}</code>	data corrente	Definisce il mese e l'anno (volendo il giorno) della seduta di discussione	<code>\esamedidottorato%{Febbraio 2013}</code>
<code>\scuoladidottorato%{...}</code>	SCUOLA DI DOTTORATO	Definisce il nome ufficiale della scuola di dottorato	<code>\scuoladidottorato%{SCUDO}</code>
<code>\ciclodidottorato%{...}</code>	nessuno	Definisce il numero ordinale del ciclo di dottorato	<code>\ciclodidottorato%{XV ciclo}</code>
<code>\logoseded[...]{...}</code>	nessuno	Inserisce nel frontespizio il logo dell'ateneo, specificandone opzionalmente l'altezza (default: 3 cm)	<code>\logoseded[2cm]{logouno,%logodue}</code>
<code>\setlogodistance%{...}</code>	3 em	imposta la distanza fra i loghi	<code>\setlogodistance%{25mm}</code>
<code>\retrofrontespizio%{...}</code>	nessuno	Se specificato, serve solitamente per scrivere nel verso della pagina del frontespizio alcune dichiarazioni di carattere legale; accetta un argomento composto anche da diversi capoversi	

Tabella 4.5. Comandi per modificare le parole e/o le brevi frasi scritte nel frontespizio o per cambiare le parole italiane di default in altre parole diverse, o per esprimere gli stessi concetti in un'altra lingua

<code>\FacoltaDi{...}</code>	definisce una stringa con il nome generico della struttura didattica, per esempio “Scuola di ”, prima del nome della struttura; di default è impostata una stringa nulla, cosicché non viene scritto nulla nel frontespizio in merito alla struttura didattica competente
<code>\StrutturaDidattica{...}</code>	sinonimo del comando <code>\FacoltaDi</code>
<code>\DottoratoIn{...}</code>	definisce una stringa equivalente a “Dottorato in” prima del nome del dottorato
<code>\CorsoDiLaureaIn{...}</code>	definisce una stringa equivalente a “Corso di Laurea in” prima del nome del corso di laurea
<code>\TesiDiLaurea{...}</code>	definisce una stringa equivalente a “Tesi di Laurea”
<code>\NomeMonografia{...}</code>	definisce una stringa equivalente a “Monografia di Laurea”
<code>\NomeDissertazione{...}</code>	definisce una stringa equivalente a “Dissertazione di Dottorato”
<code>\InName{...}</code>	definisce una stringa equivalente a “in”; in tedesco potrebbe essere “auf”, in francese “en”, ecc.
<code>\CandidateName{...}</code>	definisce una stringa equivalente a “Candidato”; per questo e i due comandi successivi il programma riesce a scegliere la stringa giusta adattata in numero genere; nel cambiare queste stringhe il compositore ha una sola possibilità e deve scegliere direttamente il genere e il numero.
<code>\AdvisorName{...}</code>	definisce una stringa equivalente a “Relatore”
<code>\CoAdvisorName{...}</code>	definisce una stringa equivalente a “Correlatore”; questo comando si può usare sempre, ma il suo contenuto viene effettivamente usato solo se si specifica l'opzione <code>classica</code> ; se in italiano non piace “Correlatore” ma si preferisce “Corelatore” o “Co-relatore”, sempre con l'opzione <code>classica</code> , si può correggere la versione di default.
<code>\TutorName{...}</code>	definisce una stringa equivalente a “Tutore”
<code>\NomeTutoreAziendale{...}</code>	definisce una stringa equivalente a “Supervisore aziendale”; usando un argomento che contenga anche indicazioni di “a capo”, nella seconda riga si può scrivere il nome dell'azienda.
<code>\CycleName{...}</code>	definisce una stringa equivalente a “ciclo”; serve essenzialmente per indicare il ciclo di dottorato
<code>\NomePrimoTomo{...}</code>	definisce una stringa equivalente a “Tomo primo”
<code>\NomeSecondoTomo{...}</code>	definisce una stringa equivalente a “Tomo secondo”
<code>\NomeTerzoTomo{...}</code>	definisce una stringa equivalente a “Tomo terzo”
<code>\NomeQuartoTomo{...}</code>	definisce una stringa equivalente a “Tomo quarto”; tutte e quattro queste stringhe dipendono dalla lingua usata e dall'ordine che si vuole o si deve dare alle due parole.

# Conclusioni

Il pacchetto TOPtesi fa quasi tutto quello che è necessario per comporre uno qualunque di quegli scritti che vengono chiamati *monografia di laurea*, *elaborato finale di laurea*, *tesi di laurea*, *tesi di laurea specialistica*, *tesi di laurea magistrale*, *dissertazione di dottorato*.

Non fa tutto, ci mancherebbe altro, visto che nessuna persona è in grado di prevedere tutte le necessità delle altre persone.

Da un lato alcuni devono adattarsi a quello che è disponibile, magari dandosi da fare per creare qualche cosa di nuovo e utile per se stessi e forse anche per gli altri. Qualcun altro deve sapersi immedesimare nelle necessità altrui e magari deve darsi da fare per creare qualcosa di nuovo e utile agli altri e forse anche a se stesso.

Una cosa certamente che rimane da fare è un procedimento più semplice per la composizione del frontespizio in una lingua diversa dall'italiano. Attualmente il procedimento si rifà all'uso delle macro che servono per definire le stringhe in lingua; il sistema dovrebbe essere completo ma richiede un po' di attenzione da parte dell'utente.

Il pacchetto TOPtesi carica di default sia `babel` o `polyglossia` sia `graphicx`; certo potrebbe caricare di default anche `amsmath` e i suoi compagni, ma non tutti hanno bisogno di scrivere e comporre matematica avanzata e se non vengono usati essi comportano uno spreco di memoria e un sia pur piccolo rallentamento nell'esecuzione. Ritengo quindi che sia meglio che ognuno si carichi i pacchetti che intende davvero usare.

Si ricordi che specificando l'opzione `pdfa`, bisogna usare esclusivamente `pdflatex`; viene caricato un file di estensione specifico che a sua volta attiva la possibilità di disporre dei link interni ed esterni, che sono così comodi per “navigare” dentro il documento. Eventuali personalizzazioni devono perciò venire eseguite immediatamente dopo la dichiarazione della classe, altrimenti non vengono recepite correttamente. Bisogna altresì ricordarsi di predisporre il file con i *metadata* richiesti dal formato PDF/A.

È possibile usare il semplice `latex`; ma questo serve solo per produrre un file in formato DVI, che può servire solo per una stampa della tesi su carta; può venire trasformato nel formato PS tramite l'applicativo `dvips`, ma questo formato non stamperebbe meglio di quel che si potrebbe ottenere dai “previewer” recenti per il formato DVI che siano anche in grado di inviare i file DVI in stampa. Viceversa è più interessante disporre della versione finale in formato PDF, cosa che si può ottenere mediante il programma `ps2pdf` il quale, con opportuni artifici, permette anche di produrre un file in formato PDF/A. Ma allora

perché non ottenere questo risultato in un passaggio solo? Certo, potrebbe esserci l'esigenza di usare estensioni che fanno uso del linguaggio PostScript, ma, solitamente, agendo direttamente con `pdflatex` si possono usare altri pacchetti di estensione che consentono di evitare del tutto il linguaggio PostScript.

Non consiglierei, invece, l'uso generalizzato di `XYLATEX`, ma lo consiglierei solo per quelle tesi, specialmente di carattere umanistico, che hanno bisogno di maneggiare agevolmente font di diversi tipi per scrivere con alfabeti o sillabari o sistemi di ideogrammi per i quali `pdflatex` è, sì, attrezzato, ma rende molto più faticosa l'intera operazione. Molto interessante sarebbe anche il programma di composizione `LuaATEX`, che incorpora una buona parte del linguaggio di scripting Lua; questo a sua volta permette di compiere diverse azioni esternamente al programma di tipocomposizione, e in certi casi esso si può considerare una estensione molto utile del programma `pdflatex`. Benché questo programma possa usare la microtipografia come `pdflatex` e i font OpenType come `XYLATEX`, non vale la pena farne uso se non si ha davvero bisogno di sfruttarne le funzionalità. Inoltre oggi il nuovo linguaggio di programmazione interno del sistema `TEX`, L<sup>3</sup>, viene usato prevalentemente proprio per creare il formato di questo programma di tipocomposizione.

So per certo che le tesi di filologia classica possono essere composte molto bene con `pdflatex` e TOPtesi; se bisogna usare una buona dose di lingua greca classica è conveniente caricare il pacchetto *teubner*<sup>3</sup>; gli intenditori sanno perfettamente che i font della casa editrice Teubner di Lipsia sono fra i più gradevoli che esistano; il pacchetto *teubner* fa uso di un'ottima imitazione di quei font e mette a disposizione una miriade di comandi per comporre quei segni “strani” che i filologi usano per scrivere le loro opere.

Per le edizioni critiche esiste il pacchetto `eledmac`; io non l'ho mai usato, ma sembra che sia di grande aiuto per gli umanisti.

Per gli scienziati e i tecnologi esistono troppi pacchetti specializzati e sarebbe impossibile, e forse inutile, elencarli tutti o anche elencare solo i più importanti.

Non resta che augurare:

**Buona composizione con `pdflatex` o `XYLATEX` o `LuaATEX`!**

---

<sup>3</sup>A tutt'oggi (2016) il pacchetto *teubner* funziona correttamente solo con `pdflatex`.

# Appendice A

## The L<sup>A</sup>T<sub>E</sub>X Project Public License

LPPL Version 1.3, 2003-12-01

Copyright 1999 2002-03 L<sup>A</sup>T<sub>E</sub>X3 Project

Everyone is allowed to distribute verbatim copies of this license document, but modification of it is not allowed.

### A.1 Preamble

The L<sup>A</sup>T<sub>E</sub>X Project Public License (LPPL) is the primary license under which the L<sup>A</sup>T<sub>E</sub>X kernel and the base L<sup>A</sup>T<sub>E</sub>X packages are distributed.

You may use this license for any work of which you hold the copyright and which you wish to distribute. This license may be particularly suitable if your work is T<sub>E</sub>X-related (such as a L<sup>A</sup>T<sub>E</sub>X package), but you may use it with small modifications even if your work is unrelated to T<sub>E</sub>X.

The section ‘Whether and How to Distribute Works under This License’, below, gives instructions, examples, and recommendations for authors who are considering distributing their works under this license.

This license gives conditions under which a work may be distributed and modified, as well as conditions under which modified versions of that work may be distributed.

We, the L<sup>A</sup>T<sub>E</sub>X3 Project, believe that the conditions below give you the freedom to make and distribute modified versions of your work that conform with whatever technical specifications you wish while maintaining the availability, integrity, and reliability of that work. If you do not see how to achieve your goal while meeting these conditions, then read the document ‘cfgguide.tex’ and ‘modguide.tex’ in the base L<sup>A</sup>T<sub>E</sub>X distribution for suggestions.

### A.2 Definitions

In this license document the following terms are used:

**Work** Any work being distributed under this License.

**Derived Work** Any work that under any applicable law is derived from the Work.

**Modification** Any procedure that produces a Derived Work under any applicable law – for example, the production of a file containing an original file associated with the Work or a significant portion of such a file, either verbatim or with modifications and/or translated into another language.

**Modify** To apply any procedure that produces a Derived Work under any applicable law.

**Distribution** Making copies of the Work available from one person to another, in whole or in part. Distribution includes (but is not limited to) making any electronic components of the Work accessible by file transfer protocols such as FTP or HTTP or by shared file systems such as Sun’s Network File System (NFS).

**Compiled Work** A version of the Work that has been processed into a form where it is directly usable on a computer system. This processing may include using installation facilities provided by the Work, transformations of the Work, copying of components of the Work, or other activities. Note that modification of any installation facilities provided by the Work constitutes modification of the Work.

**Current Maintainer** A person or persons nominated as such within the Work. If there is no such explicit nomination then it is the ‘Copyright Holder’ under any applicable law.

**Base Interpreter** A program or process that is normally needed for running or interpreting a part or the whole of the Work. A Base Interpreter may depend on external components but these are not considered part of the Base Interpreter provided that each external component clearly identifies itself whenever it is used interactively. Unless explicitly specified when applying the license to the Work, the only applicable Base Interpreter is a “L<sup>A</sup>T<sub>E</sub>X-Format”.

### A.3 Conditions on Distribution and Modification

1. Activities other than distribution and/or modification of the Work are not covered by this license; they are outside its scope. In particular, the act of running the Work is not restricted and no requirements are made concerning any offers of support for the Work.
2. You may distribute a complete, unmodified copy of the Work as you received it. Distribution of only part of the Work is considered modification of the Work, and no right to distribute such a Derived Work may be assumed under the terms of this clause.



3. You may distribute a Compiled Work that has been generated from a complete, unmodified copy of the Work as distributed under Clause 2 above, as long as that Compiled Work is distributed in such a way that the recipients may install the Compiled Work on their system exactly as it would have been installed if they generated a Compiled Work directly from the Work.
4. If you are the Current Maintainer of the Work, you may, without restriction, modify the Work, thus creating a Derived Work. You may also distribute the Derived Work without restriction, including Compiled Works generated from the Derived Work. Derived Works distributed in this manner by the Current Maintainer are considered to be updated versions of the Work.
5. If you are not the Current Maintainer of the Work, you may modify your copy of the Work, thus creating a Derived Work based on the Work, and compile this Derived Work, thus creating a Compiled Work based on the Derived Work.
6. If you are not the Current Maintainer of the Work, you may distribute a Derived Work provided the following conditions are met for every component of the Work unless that component clearly states in the copyright notice that it is exempt from that condition. Only the Current Maintainer is allowed to add such statements of exemption to a component of the Work.
  - (a) If a component of this Derived Work can be a direct replacement for a component of the Work when that component is used with the Base Interpreter, then, wherever this component of the Work identifies itself to the user when used interactively with that Base Interpreter, the replacement component of this Derived Work clearly and unambiguously identifies itself as a modified version of this component to the user when used interactively with that Base Interpreter.
  - (b) Every component of the Derived Work contains prominent notices detailing the nature of the changes to that component, or a prominent reference to another file that is distributed as part of the Derived Work and that contains a complete and accurate log of the changes.
  - (c) No information in the Derived Work implies that any persons, including (but not limited to) the authors of the original version of the Work, provide any support, including (but not limited to) the reporting and handling of errors, to recipients of the Derived Work unless those persons have stated explicitly that they do provide such support for the Derived Work.
  - (d) You distribute at least one of the following with the Derived Work:
    - i. A complete, unmodified copy of the Work; if your distribution of a modified component is made by offering access to copy the modified component from a designated place, then offering equivalent access to copy

- the Work from the same or some similar place meets this condition, even though third parties are not compelled to copy the Work along with the modified component;
- ii. Information that is sufficient to obtain a complete, unmodified copy of the Work.
7. If you are not the Current Maintainer of the Work, you may distribute a Compiled Work generated from a Derived Work, as long as the Derived Work is distributed to all recipients of the Compiled Work, and as long as the conditions of Clause 6, above, are met with regard to the Derived Work.
  8. The conditions above are not intended to prohibit, and hence do not apply to, the modification, by any method, of any component so that it becomes identical to an updated version of that component of the Work as it is distributed by the Current Maintainer under Clause 4, above.
  9. Distribution of the Work or any Derived Work in an alternative format, where the Work or that Derived Work (in whole or in part) is then produced by applying some process to that format, does not relax or nullify any sections of this license as they pertain to the results of applying that process.
  10.
    - (a) A Derived Work may be distributed under a different license provided that license itself honors the conditions listed in Clause 6 above, in regard to the Work, though it does not have to honor the rest of the conditions in this license.
    - (b) If a Derived Work is distributed under this license, that Derived Work must provide sufficient documentation as part of itself to allow each recipient of that Derived Work to honor the restrictions in Clause 6 above, concerning changes from the Work.
  11. This license places no restrictions on works that are unrelated to the Work, nor does this license place any restrictions on aggregating such works with the Work by any means.
  12. Nothing in this license is intended to, or may be used to, prevent complete compliance by all parties with all applicable laws.

## A.4 No Warranty

There is no warranty for the Work. Except when otherwise stated in writing, the Copyright Holder provides the Work ‘as is’, without warranty of any kind, either expressed or implied, including, but not limited to, the implied warranties of merchantability and fitness for a particular purpose. The entire risk as to the quality and performance of the Work is

with you. Should the Work prove defective, you assume the cost of all necessary servicing, repair, or correction.

In no event unless agreed to in writing will the Copyright Holder, or any author named in the components of the Work, or any other party who may distribute and/or modify the Work as permitted above, be liable to you for damages, including any general, special, incidental or consequential damages arising out of any use of the Work or out of inability to use the Work (including, but not limited to, loss of data, data being rendered inaccurate, or losses sustained by anyone as a result of any failure of the Work to operate with any other programs), even if the Copyright Holder or said author or said other party has been advised of the possibility of such damages.

## A.5 Maintenance Of the Work

The Work has the status ‘author-maintained’ if the Copyright Holder explicitly and prominently states near the primary copyright notice in the Work that the Work can only be maintained by the Copyright Holder or simply that is ‘author-maintained’.

The Work has the status ‘maintained’ if there is a Current Maintainer who has indicated in the Work that they are willing to receive error reports for the Work (for example, by supplying a valid e-mail address). It is not required for the Current Maintainer to acknowledge or act upon these error reports.

The Work changes from status ‘maintained’ to ‘unmaintained’ if there is no Current Maintainer, or the person stated to be Current Maintainer of the work cannot be reached through the indicated means of communication for a period of six months, and there are no other significant signs of active maintenance.

You can become the Current Maintainer of the Work by agreement with any existing Current Maintainer to take over this role.

If the Work is unmaintained, you can become the Current Maintainer of the Work through the following steps:

1. Make a reasonable attempt to trace the Current Maintainer (and the Copyright Holder, if the two differ) through the means of an Internet or similar search.
2. If this search is successful, then enquire whether the Work is still maintained.
  - (a) If it is being maintained, then ask the Current Maintainer to update their communication data within one month.
  - (b) If the search is unsuccessful or no action to resume active maintenance is taken by the Current Maintainer, then announce within the pertinent community your intention to take over maintenance. (If the Work is a  $\text{\LaTeX}$  work, this could be done, for example, by posting to comp.text.tex.)

3. (a) If the Current Maintainer is reachable and agrees to pass maintenance of the Work to you, then this takes effect immediately upon announcement.  
(b) If the Current Maintainer is not reachable and the Copyright Holder agrees that maintenance of the Work be passed to you, then this takes effect immediately upon announcement.
4. If you make an ‘intention announcement’ as described in 2b. above and after three months your intention is challenged neither by the Current Maintainer nor by the Copyright Holder nor by other people, then you may arrange for the Work to be changed so as to name you as the (new) Current Maintainer.
5. If the previously unreachable Current Maintainer becomes reachable once more within three months of a change completed under the terms of 3b) or 4), then that Current Maintainer must become or remain the Current Maintainer upon request provided they then update their communication data within one month.

A change in the Current Maintainer does not, of itself, alter the fact that the Work is distributed under the LPPL license.

If you become the Current Maintainer of the Work, you should immediately provide, within the Work, a prominent and unambiguous statement of your status as Current Maintainer. You should also announce your new status to the same pertinent community as in 2b) above.

## A.6 Whether and How to Distribute Works under This License

This section contains important instructions, examples, and recommendations for authors who are considering distributing their works under this license. These authors are addressed as *you* in this section.

### A.6.1 Choosing This License or Another License

If for any part of your work you want or need to use \*distribution\* conditions that differ significantly from those in this license, then do not refer to this license anywhere in your work but, instead, distribute your work under a different license. You may use the text of this license as a model for your own license, but your license should not refer to the LPPL or otherwise give the impression that your work is distributed under the LPPL.

The document ‘modguide.tex’ in the base  $\LaTeX$  distribution explains the motivation behind the conditions of this license. It explains, for example, why distributing  $\LaTeX$  under the GNU General Public License (GPL) was considered inappropriate. Even if your work is unrelated to  $\LaTeX$ , the discussion in ‘modguide.tex’ may still be relevant, and authors intending to distribute their works under any license are encouraged to read it.

## A.6.2 A Recommendation on Modification Without Distribution

It is wise never to modify a component of the Work, even for your own personal use, without also meeting the above conditions for distributing the modified component. While you might intend that such modifications will never be distributed, often this will happen by accident – you may forget that you have modified that component; or it may not occur to you when allowing others to access the modified version that you are thus distributing it and violating the conditions of this license in ways that could have legal implications and, worse, cause problems for the community. It is therefore usually in your best interest to keep your copy of the Work identical with the public one. Many works provide ways to control the behavior of that work without altering any of its licensed components.

## A.6.3 How to Use This License

To use this license, place in each of the components of your work both an explicit copyright notice including your name and the year the work was authored and/or last substantially modified. Include also a statement that the distribution and/or modification of that component is constrained by the conditions in this license.

Here is an example of such a notice and statement:

```
%% pig.dtx
%% Copyright 2003 M. Y. Name
%
% This work may be distributed and/or modified under the
% conditions of the LaTeX Project Public License, either
% version 1.3 of this license or (at your option) any later
% version. The latest version of this license is in
% http://www.latex-project.org/lppl.txt
% and version 1.3 or later is part of all distributions
% of LaTeX version 2003/12/01 or later.
%
% This work has the LPPL maintenance status "maintained".
%
% This Current Maintainer of this work is M. Y. Name.
%
% This work consists of the files pig.dtx and pig.ins
% and the derived file pig.sty.
```

Given such a notice and statement in a file, the conditions given in this license document would apply, with the ‘Work’ referring to the three files ‘pig.dtx’, ‘pig.ins’, and ‘pig.sty’ (the last being generated from ‘pig.dtx’ using ‘pig.ins’), the ‘Base Interpreter’ referring to any “ $\text{\LaTeX}$ -Format”, and both ‘Copyright Holder’ and ‘Current Maintainer’ referring to the person ‘M. Y. Name’.

To prevent the Maintenance section of LPPL from allowing someone else to become the Current Maintainer without your agreement, you could change “maintained” above into “author-maintained”.

#### **A.6.4 Important Recommendations**

##### Defining What Constitutes the Work

The LPPL requires that distributions of the Work contain all the files of the Work. It is therefore important that you provide a way for the licensee to determine which files constitute the Work. This could, for example, be achieved by explicitly listing all the files of the Work near the copyright notice of each file or by using a line such as:

```
% This work consists of all files listed in manifest.txt.
```

in that place. In the absence of an unequivocal list it might be impossible for the licensee to determine what is considered by you to comprise the Work and, in such a case, the licensee would be entitled to make reasonable conjectures as to which files comprise the Work.

# Indice analitico

- \\, 81
- \AdvisorName, 88
- ambiente
  - citazioni*, 84
  - dedica*, 84
  - filecontents\**, 60, 62
  - frontespizio*, 23, 26, 28, 30, 35, 40, 41, 78, 79, 83
  - frontespizio\**, 28, 30, 35, 78, 79, 83
  - interlinea*, 73, 75
  - otherlanguage*, 55
  - Preambolo\**, 41, 42
  - titlepage*, 6, 9, 35, 36, 39
- \annoaccademico, 84
- \ateneo, 28, 30, 79, 85
- \begin, 40, 41, 43, 74–78
- \blankpagestyle, 74, 75
- \candidata, 81, 84, 86
- \CandidateName, 88
- \candidato, 79, 81, 84, 86
- \chapter, 43, 75, 76
- \ciclodidottorato, 87
- classe
  - book*, 8
  - report*, 8, 50, 71
  - sapthesis*, 3, 8, 21
  - sufftesi*, 3, 8, 21, 36, 41
  - TesiClassica*, 3
  - TesiModerna*, 3
  - toptesi*, 5, 6, 8, 24, 27, 28, 40, 49–51, 56, 60, 71, 77, 78, 81–83
- \cleardoublepage, 74–76
- \cleradoublepage, 75
- \CoAdvisorName, 88
- \coordinatore, 86
- \Correlatore, 41
- \corsodidottorato, 85
- \corsodilaurea, 85
- \CorsoDiLaureaIn, 88
- \CycleName, 88
- \direttore, 86
- \documentclass, 26, 40, 62, 63
- \DottoratoIn, 88
- \E, 46
- \end, 41, 43, 74
- \EnDash, 84
- \endinput, 43
- \english, 26, 55, 56
- \esamedidottorato, 87
- \eu, 74
- \facolta, 22, 41, 85
- \FacoltaDi, 22, 88
- \figurespagefalse, 83
- \figurespagetrue, 83
- file
  - frn.tex*, 40
  - .bib*, 45
  - .bmp*, 32

- 
- .cfg, 6
  - .eps, 32, 70
  - .fmt, 1
  - .icc, 66
  - .jpg, 32, 33, 70
  - .mps, 32
  - .pdf, 32, 70
  - .png, 32, 33, 70
  - .tex, 3, 17, 62
  - .tiff, 32
  - .wmf, 32
  - .xmpdata, 63
  - capitolo1.tex, 44
  - capitolo2.tex, 44
  - capitolo3.tex, 44
  - euler, 7
  - grfguide.pdf, 81
  - lualatex,fmt, 2
  - myfile.tex, 25
  - pdflatex,fmt, 2
  - pdflatex.log, 52
  - pdfx.sty, 62
  - preliminari.tex, 44
  - sample.tex, 51, 53
  - srgb.icc, 66
  - tesi.tex, 43
  - topfront.example.pdf, 3
  - toptesi-example-con-  
frontespizio.pdf,  
3
  - toptesi-example-con-  
frontespizio.tex,  
42
  - toptesi-example-luatex.pdf, 3
  - toptesi-example-xetex.tex, 42
  - toptesi-example.pdf, 3
  - toptesi-example.tex, 42
  - toptesi-it-pages.tex, 67
  - toptesi-it-pages.xmpdata, 68
  - toptesi-it.pdf, 68
  - toptesi-it.xmpdata, 68
  - toptesi.cfg, 23
  - xelatex,fmt, 2
  - xelatex.log, 52
  - font
    - iwona, 37
    - newpxtext, 74
  - \fontsize, 37
  - \foreignlanguage, 55
  - \frontespizio, 23, 28, 30, 79, 82–84
  - \frontespizio\*, 26, 28, 30, 82, 83
  - \gei, 74
  - \goodbreak, 76
  - \goodpagebreak, 74, 76, 77
  - \gradi, 74
  - \GuIT, 67
  - \hypersetup, 61
  - \IDN, 81, 84
  - \ifPDFTeX, 62
  - \ifTOPfront, 74
  - \includeonly, 44
  - \indici, 83
  - \InName, 88
  - \interlinea, 73, 74
  - \italiano, 55, 56
  - \linespread, 74
  - \listing, 74
  - \Logo, 41
  - \logosedede, 33, 87
  - \Materia, 85
  - \materia, 85
  - \monografia, 79, 85
  - \MP, 67
  - \NCorrelatore, 41
  - \newcommand, 41
  - \newpage, 76
  - \nomeateneo, 85
  - \NomeDissertazione, 88



- \NomeMonografia, 88
- \NomePrimoTomo, 88
- \NomeQuartoTomo, 88
- \NomeSecondoTomo, 88
- \NomeTerzoTomo, 88
- \NomeTutoreAziendale, 88
- \nota, 84
- \NoteWhiteLine, 83
- \ohm, 74
- opzione
  - shell-escape, 63
  - z 0, 63
  - 14pt, 72
  - a-1b, 61
  - ancient, 46, 47
  - autoretitolo, 72, 79
  - chapterbib, 71
  - classica, 28, 71, 79, 84
  - cucitura, 51, 72
  - english, 6
  - greek, 46
  - italian, 6
  - landscape, 37, 39
  - latin1, 45
  - noTOPfront, 6, 40, 73, 74, 77, 82
  - oldstyle, 72
  - openright, 75
  - OT1, 7, 48
  - pdfa, 6, 72
  - polutoniko, 47
  - T1, 37, 48–50
  - usefrontespizio, 73
  - utf8, 4, 5, 7, 45, 49, 60, 63
- pacchetto
  - amsfonts*, 7
  - amsmath*, 7
  - amssymb*, 7
  - amsthm*, 7
  - babel*, 6, 26, 27, 30, 46, 51, 55
  - biblatex*, 45, 71
  - bm*, 7
  - caption*, 8
  - cfr-lm*, 50
  - etoolbox*, 5
  - fontenc*, 7, 49
  - fontspec*, 7, 49, 57
  - frontespizio*, 3, 8, 35, 36, 39–41
  - geometry*, 36, 42
  - graphicx*, 5, 81
  - hperref*, 61
  - hyperref*, 6, 60, 61
  - ifxetex*, 6
  - inputenc*, 7
  - iwona*, 37
  - listings*, 32
  - microtype*, 50, 59
  - newpx...*, 49
  - newpxmath*, 49
  - newpxtext*, 49
  - newtx...*, 49
  - newtxmath*, 49
  - newtxtext*, 49
  - pax*, 68
  - pdfpages*, 39, 42, 64, 67, 68
  - pdftricks*, 82
  - pdfx*, 6, 59–61, 63, 68, 72
  - pgf*, 62
  - poliglossia*, 30
  - polyglossia*, 6, 20, 27, 46, 51, 56
  - PSTricks*, 62, 81
  - teubner*, 47, 90
  - tikz*, 7, 62
  - topcoman*, 6, 24, 30
  - topfront*, 6, 8, 24–26, 28, 30, 33, 35, 40, 73, 77, 82
  - toptesi*, v, 5, 26, 34, 55, 62, 70
  - xspace*, 46
  - XYpic*, 62
  - \pagebreak, 76
  - \pagegoal, 76
  - \pagetotal, 76

- \paginavuota, 83
- \part, 76
- \ped, 74
- programma
  - Adobe Acrobat, 16
  - Adobe Reader, 16
  - apt-get, 14
  - biber, 45, 71
  - bibtex, 45
  - emacs, 48
  - ghostscript, 16, 62, 64, 66, 67
  - gs, 66
  - GSView, 16
  - gswin32c, 66
  - Inkscape, 70
  - latex, 20, 62
  - lualatex, 27, 30, 32, 40, 45, 49, 50, 55–64, 70
  - MacTeX, 15, 17, 54
  - pdflatex, v, 20, 26, 27, 32, 40, 46, 49, 50, 52, 54, 56–64, 67, 68, 70
  - pdflatex., 68
  - pdftex, 56, 62
  - rpm, 14
  - SumatraPDF, 17
  - TeXmaker, 17
  - TeXnicCenter, 17
  - TeXShop, 17
  - TeXStudio, 17
  - TeXworks, 17
  - Vim, 48
  - Writer2LaTeX, 18
  - xdvipdfmx, 63
  - xelatex, 26, 27, 30, 32, 40, 45, 49, 50, 55–64, 70
  - xetex, 19
  - yast, 14
- \quad, 81
- \QualificaDirettore, 86
- \ref, 27
- \relatore, 79, 86
- \renewcommand, 41
- \retrofrontespizio, 87
- \ringraziamenti, 56, 83
- \scuoladidottorato, 87
- \secondacandidata, 86
- \secondocandidato, 86
- \secondorelatore, 79, 86
- \sedutadilaurea, 87
- \selectlanguage, 26
- \setlogodistance, 34, 87
- \setotherlanguage, 26
- \sommario, 56, 83
- \sottotitolo, 79, 85
- \space, 81
- \spagnolo, 56
- \struttura, 22, 85
- \StrutturaDidattica, 22, 88
- \tablespagefalse, 83
- \tablespagetrue, 83
- \terzacandidata, 87
- \terzocandidato, 87
- \terzorelatore, 79, 86
- \TesiDiLaurea, 88
- \textheight, 76
- \thispagestyle, 36
- \titolo, 72, 79, 85
- \tomo, 84
- \tutore, 86
- \tutoreaziendale, 86
- \TutorName, 88
- \unit, 74
- \Universita, 41
- \usefont, 37
- \usepackage, 40, 41, 71
- \vadjust, 76
- \xspace, 46