

combelow: abbasso i segni diacritici di serie B

Gianluca Pignalberi

Sommario

Romeno e lettone devono essere considerate lingue di serie B nel mondo di $\text{T}_{\text{E}}\text{X}$? Se finora lo potevano essere, questo piccolo pacchetto tenta di riportarle al livello giusto con il semplice uso del segno diacritico corretto. Niente più cediglia al posto del *comma below*.

Abstract

Should Romanian and Latvian be considered second choice languages in the $\text{T}_{\text{E}}\text{X}$ world? Though up to now they could have been, this small package tries to put them back at the right level, just using the correct diacritic mark. No more cedilla instead of *comma below*.

1 Introduzione

Se avete modo di vedere, o leggere se ne siete in grado — io purtroppo no —, delle pubblicazioni in lingua romena, vi accorgerete senz'altro che spesso le 't' e le 's' hanno un segno strano. Questo segno diacritico, un modificatore di suono, è una virgola centrata sotto il carattere in questione.

Se provate a riscrivere un testo cercando di riprodurre fedelmente quel segno diacritico, vi troverete presto frustrati dal fatto che $\text{L}_{\text{A}}\text{T}_{\text{E}}\text{X}$ non ha un comando per comporlo.

Guardando più attentamente i giornali romeni noterete che, in base alla testata e al font in uso, quella specie di virgola viene sostituita con una cediglia (¸). Anch'io, dovendo scrivere un volantino in romeno (copiando bovinamente quanto mi è stato scritto), ho dovuto sottostare alla sostituzione, con ampio senso di insoddisfazione, nonostante le rassicurazioni di madrelingua romeni.

Per determinare da quale parte stava la correttezza, ho fatto ricorso alle tabelle Unicode (KORPELA, 2006; UNICODE CONSORTIUM, 2009a) e a Wikipedia (WIKIPEDIA (2009b); e il più dettagliato corrispondente italiano WIKIPEDIA (2009a)). Queste mi hanno detto alcune cose non proprio armoniche, di cui proviamo a ricostruire la natura. Le tabelle di Unicode (UNICODE CONSORTIUM, 2009c,d) non prevedono caratteri con *comma below*, ma solo con cediglia. La forma della cediglia però varia a seconda della lettera a cui è applicata, e non riesco a capire la logica di quest'applicazione. Si può comunque notare che la cediglia sotto 's' e 't' è a forma di uncino (la forma caratteristica del simbolo, somigliante alla parte inferiore del numero 5), mentre sotto le altre lettere per cui

è prevista la cediglia questa è a forma di virgola. In UNICODE CONSORTIUM (2009d) viene detto che 'š' e 'ţ' sono varianti tipografiche preferibili a 's' e 't' per il romeno. In ogni caso, il simbolo '¸' compare da solo, come *comma below*, nella tabella delle composizioni degli accenti (UNICODE CONSORTIUM, 2009b).

Wikipedia, invece, è un po' più dettagliata, spiegando in quali lingue si usa la cediglia con la lettera 's' (ad esempio in turco, tartaro e turkmeno) e la proposta di Ambroise Firmin-Didot per usarla con la 't'. Spiega il suo uso in lettone, dicendo che tipograficamente il simbolo è una virgola, e ha un caso particolare di posizionamento che vedremo nel paragrafo 3; precisa infine la questione della cediglia in romeno. Fornisce inoltre un link molto interessante al Diacritics Project (TYPO.CZ e DESIGNIQ.CZ, 2009). Qui il simbolo in questione viene chiamato *comma accent*.

Tutte le informazioni raccolte mi hanno permesso la scrittura di un pacchetto (veramente più di uno, come leggerete nel paragrafo 4), rilasciato da pochi giorni su CTAN, chiamato *combelow*.

2 $\text{L}_{\text{A}}\text{T}_{\text{E}}\text{X}$ e cediglia

Abbiamo accennato al fatto che gli utenti $\text{L}_{\text{A}}\text{T}_{\text{E}}\text{X}$ hanno sempre approssimato il *comma below* con la cediglia in caso di testi in lingua romena. La stessa cosa avviene con *babel* caricando l'opzione per la lingua romena (BRAAMS, 2006). Siamo abituati a vedere la cediglia sotto la lettera 'c' in francese. Abbiamo visto anche che metterla sotto 's' e 't' non crea problemi dal punto di vista tipografico: abbiamo sempre considerato lettere con profondità praticamente pari a zero.

La lingua lettone prevede l'uso del *comma below* con 'g', 'k', 'l', 'n' e 'r'. Possiamo sempre approssimare il *comma below* con la cediglia? Vediamo che succede: g, k, l, n, r. La cediglia non è adatta a essere usata con lettere discendenti in quanto non tiene conto della profondità della lettera a cui si applica (il fatto viene anche citato a pagina 47 di ELJKHOUT (1993)). Oltre a non essere il simbolo corretto nei casi esaminati. Volendo poi essere pignoli, non viene neanche posto bene sotto la 'r': andrebbe allineato all'asta.

3 Comma below: casi normali e speciali

Il segno *comma below* ha un caso particolare di applicazione, proprio nella lingua lettone, con la 'g',

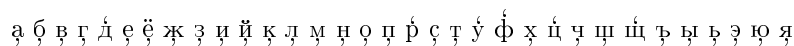


FIGURA 1: Il segno comma below applicato all’alfabeto russo: anche in questo caso il rilevamento automatico dei discendenti funziona sempre, così come quello della diversa altezza degli ascendenti. Non mi risulta che il comma below sia applicato all’alfabeto cirillico

l’unica lettera discendente. Mentre nei casi normali il segno diacritico si trova centrato sotto la lettera a cui si applica, nel caso della ‘g’ questo deve essere posto centrato al di *sopra* del carattere, e ruotato di 180°. In questo caso cambia anche il nome del segno diacritico, diventando *turned comma above* (‘ ’).

Il comando introdotto dal pacchetto `combelow` è `\cb`, e prende come unico argomento il carattere a cui bisogna applicare il segno diacritico. In caso il carattere sia Unicode multibyte, questo *deve* essere posto in parentesi graffe. Quello che si ottiene applicando il nuovo segno diacritico ai caratteri visti delle lingue romena e lettone è questo: `ĝ`, `ķ`, `ļ`, `ņ`, `ŗ`, `ș`, `ț`.

Per come avevo scritto il codice in due versioni precedenti del pacchetto (mai rilasciate), il comando non era in grado di riconoscere le lettere discendenti, potendo quindi discriminare il solo caso della ‘g’; per tacere del maggiore spreco di memoria dovuto alla minor efficienza dell’algoritmo usato. Ora, come vedremo nel paragrafo 5, il pacchetto riconosce tutti i casi, anche per alfabeti diversi dal latino. Vediamo un esempio di applicazione del comma below alle lettere dell’alfabeto russo nella figura 1, sebbene l’esempio abbia solo valore estetico: non c’è questo segno in cirillico.

4 Peripezie di un codice

La mia “conoscenza” di `TEX` e `LATEX` mi ha permesso di scrivere un pacchetto che ritenevo quasi definitivo (versione 0.99), ma che non ha resistito ai suggerimenti di Claudio Beccari in fase di giudizio. Siccome è una persona *veramente* competente, ha trovato immediatamente il suo pregio e i suoi difetti. La mia affezione al codice e al numero mi ha convinto a lavorare a versioni parallele del pacchetto (una sorta di `fork`), e a contrassegnare le versioni con una lettera aggiuntiva. Nelle sezioni seguenti descriviamo tutte le versioni prodotte, a partire dalla prima versione, quella radiografata dal comitato scientifico, e definita in seguito 0.99a. Si tenga presente che alcune idee per eliminare i difetti sono tratte da un codice di esempio di `combelow` scritto da Claudio Beccari. La qualità del suo codice è semplicemente annichilente.

5 combelow 0.99a

Il codice del pacchetto viene riportato qui di seguito, riformattato per esigenze di impaginazione e senza commenti di qualunque natura. Allo stesso modo saranno riformattati tutti gli altri codici.

```

1 \NeedsTeXFormat{LaTeX2e}
2 \ProvidesPackage{combelow}
3   [2009/08/23 v0.99a comma below
4     command]
5 \RequirePackage{graphicx}
6 \RequirePackage{ifthen}
7 \newdimen\fcxpos
8 \newdimen\fcypos
9 \newdimen\ftpos
10 \newcommand{\cb}[1]{%
11 \setbox1=\hbox{#1}%
12 \setbox2=\hbox{,}%
13 \fcxpos=.5\wd1%
14 \advance\fcxpos by .5\wd2%
15 \ftpos=.5\wd1%
16 \advance\ftpos by -.5\wd2%
17 \ifthenelse{\lengthtest{\dp1>
18 .1\ht1}}%
19 {\fcypos=\ht1%
20 \advance\fcypos by .6ex%
21 \raisebox{0pt}{#1\hspace
22 {-\fcxpos}\raisebox{\fcypos}%
23 {\rotatebox[origin=c]{180}{,}
24 \hspace{\ftpos}}}}%
25 {\fcypos=\dp1%
26 \advance\fcypos by .4ex%
27 \raisebox{0pt}{#1\hspace
28 {-\fcxpos}\raisebox{-\fcypos},
29 \hspace{\ftpos}}}%
30 }
\endinput

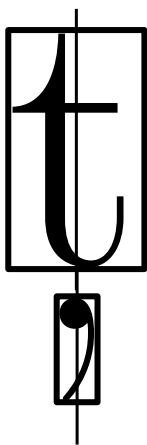
```

5.1 Commento al codice

Le righe da 1 a 8 specificano la versione minima di `LATEX` prevista, caricano i pacchetti necessari al funzionamento e definiscono alcune variabili usate per posizionare il segno diacritico e il testo dopo la lettera accentata; le righe 9–29 definiscono il comando `\cb`.

In due box (righe 10 e 11) poniamo il carattere da accentare e la virgola (il segno diacritico). Siccome il segno diacritico deve essere centrato sopra o sotto la lettera (le mediane sono allineate verticalmente), la posizione del suo estremo sinistro sull’asse *x* corrisponderà alla metà del box della lettera, traslato a sinistra della metà del proprio box (righe 12 e 13); vedi la figura 2.

Le righe 14 e 15 calcolano la posizione del testo dopo la lettera accentata: questo si troverà a destra del box del carattere, separato da esso da uno spazio pari a metà del box della virgola. Se non è il valore esatto, è senz’altro un’ottima approssimazione (vedi la figura 3).

FIGURA 2: Allineamento dei box nel comando `\cb`

București

FIGURA 3: Nome originale della capitale romena composto con `combelow`

Le righe 16–28 si occupano del posizionamento del segno diacritico: se la profondità del carattere è più del 10% della sua altezza (valore arbitrario, righe 16 e 17) allora bisogna porre un turned comma above, superando di poco (0.6 ex) l'altezza della lettera (righe 18–23), altrimenti si pone il segno diacritico normale sotto la lettera, lasciando il solito “spazio di rispetto” (posto arbitrariamente a 0.4 ex, righe 24–28). In entrambi i casi si lascia lo spazio orizzontale discusso per riprendere la normale composizione del testo seguente.

5.2 Pro e contro

Il pregio del pacchetto appena presentato è che è indipendente dalla codifica, cioè funziona sempre senza doversi preoccupare di quello che carichiamo in fontenc. A fronte di questo pregio presenta una plethora di difetti non da poco: l'algoritmo di posizionamento adottato fa perdere le informazioni sulla crenatura, e anche la sillabazione non sarà più possibile per la parola accentata. Vediamo un esempio di questo problema nelle figure 4 e 5. Rimandiamo al paragrafo 9 una breve discussione su quest'ultimo punto.

Ruotare la virgola di 180° è sostanzialmente inutile: si carica il pacchetto `graphicx` solo per ottenere un simbolo che è identico alla virgoletta singola aperta (‘). Un altro problema riscontrabile con questa versione del pacchetto è che se una lettera discendente con comma below deve essere inclinata (corsivo o inclinato — slanted) il segno diacritico non viene più centrato sulla lettera. Usando la virgoletta singola aperta al posto della virgola ruotata il problema scompare (vedi il paragrafo 6). Come già accennato nel paragrafo 2, anche queste due versioni della pacchetto non posizionano bene

l'accento sotto la ‘r’: la centratura mal si accorda con una lettera con asta decentrata.

Ulteriore problema è che il comando, così come è definito, non può essere usato in didascalie di figure e tabelle. La soluzione di questo problema consiste nel sostituire, nella definizione di `\cb`, il comando `\newcommand` con `\DeclareRobustCommand` (stessa sintassi). Infine, l'ultimo problema è quello relativo alla grandezza della virgola usata come segno diacritico. Molti font usati dai giornali mostrano il comma below più piccolo della virgola, come se fosse composto con un font inferiore. Ecco, `combelow 0.99a` non è in grado di farlo senza dare problemi in composizione.

6 combelow 0.99b

Tenuto conto delle osservazioni fatte nel paragrafo 5 la versione 0.99b diventa così:

```

1 \NeedsTeXFormat{LaTeX2e}
2 \ProvidesPackage{combelow}
3   [2009/08/23 v0.99b comma below
   command]
4 \RequirePackage{graphicx}
5 \RequirePackage{ifthen}
6 \newdimen\fcxpos
7 \newdimen\fcypos
8 \newdimen\ftpos
9 \DeclareRobustCommand{\cb}[1]{%
10 \setbox1=\hbox{#1}%
11 \setbox2=\hbox{ , }%
12 \fcxpos=.5\wd1%
13 \advance\fcxpos by .5\wd2%
14 \ftpos=.5\wd1%
15 \advance\ftpos by -.5\wd2%
16 \ifthenelse{\lengthtest{\dp1>
17   .1\ht1}}%
18 {\fcypos=\ht1%
19 \advance\fcypos by -.7ex%
20 \raisebox{0pt}{#1\hspace
21   {-\fcxpos}\raisebox{\fcypos}
22   {\hspace{\ftpos}}}%
23 {\fcypos=\dp1%
24 \advance\fcypos by .4ex%
25 \raisebox{0pt}{#1\hspace
26   {-\fcxpos}\raisebox{-\fcypos}
27   ,\hspace{\ftpos}}}%
28 }
29 \endinput

```

La differenza di resa sulla *g* corsiva è notevole, e visibile nella figura 6. Nel codice la differenza sostanziale sta nelle righe 19–22.

7 combelow 0.99c: un fantastico passo indietro

Su consiglio di Claudio Beccari, promosso sul campo mio mentore in quest'avventura (partita come autarchica, tanto orgogliosamente quanto avventa-

Mutu: „Amenda îmi afectează viața privată“

„În afara terenului sunt deloc în apele mele, nici nu pot să fiu. Acea amendă este de un lucru particular care îmi afectează viața privată... Totuși, când intru pe teren este ca o ușurare	fletească“, a precizat Mutu la postul de televiziune italian Sky. Tribunalul de Arbitraj Sportiv (TAS) a anunțat, la 31 iulie, că a respins memoriul formulat de Adrian Mutu la decizia Camerei	pentru Rezolvarea Disputelor (DRC) din cadrul FIFA, care a hotărât, la 7 mai 2008, că jucătorul trebuie să plătească clubului Chelsea despăgubiri în valoare de 17 milioane de euro. Cămera pentru	Rezolvarea Disputelor (DRC) din cadrul FIFA a decis, la 7 mai 2008, Mutu, acum la AC Fiorentina, trebuie să plătească fostului său club, Chelsea, despăgubiri în valoare de 17.173.990 de euro, pen-	tru încălcarea contractului, iar internaționalul român a cerut TAS să anuleze hotărârea FIFA și să stabilească faptul că nu trebuie plătită nicio compensație.
---	---	--	--	--

FIGURA 4: Stralcio di testo in romeno. Il segno diacritico usato è una cediglia, che consente la sillabazione (vedi la parola *internaționalul* in quinta colonna). Testo tratto da Gazeta Românească Nr. 32, An VII. Sono state corrette le virgolette del testo originale (") con quelle per la lingua romena („ “)

Mutu: „Amenda îmi afectează viața privată“

„În afara terenului sunt deloc în apele mele, nici nu pot să fiu. Acea amendă este de un lucru particular care îmi afectează viața privată... Totuși, când intru pe teren este ca o ușurare	fletească“, a precizat Mutu la postul de televiziune italian Sky. Tribunalul de Arbitraj Sportiv (TAS) a anunțat, la 31 iulie, că a respins memoriul formulat de Adrian Mutu la decizia Camerei	pentru Rezolvarea Disputelor (DRC) din cadrul FIFA, care a hotărât, la 7 mai 2008, Mutu, acum la AC Fiorentina, trebuie să plătească clubului Chelsea despăgubiri în valoare de 17 milioane de euro. Cămera pentru	Rezolvarea Disputelor (DRC) din cadrul FIFA a decis, la 7 mai 2008, Mutu, acum la AC Fiorentina, trebuie să plătească fostului său club, Chelsea, despăgubiri în valoare de 17.173.990 de euro, pen-	tru încălcarea contractului, iar internaționalul român a cerut TAS să anuleze hotărârea FIFA și să stabilească faptul că nu trebuie plătită nicio compensație.
---	---	--	--	--

FIGURA 5: Stesso testo in romeno. Il segno diacritico usato è un comma below, che purtroppo non consente la sillabazione nel caso analogo: “internaționalul” esce fuori margine



FIGURA 6: La \grave{g} corsiva resa con la versione 0.99a (a sinistra) e 0.99b (a destra)

tamente; povero illuso), una versione intermedia non poteva non tenere conto del codice plain \TeX usato da Knuth per posizionare gli accenti.

Dal file `plain.tex` ho preso il codice di `\c #1` e, prima di procedere similmente a Beccari, ho provato `\accent`. Mai l'avessi fatto, pur sapendo-

lo essere solo un esperimento. Il codice seguente, molto compatto, produce un output improponibile (figura 7):

```

1 \NeedsTeXFormat{LaTeX2e}
2 \ProvidesPackage{combelow}
3 [2009/08/23 v0.99c comma below
  command]
4 \DeclareRobustCommand{\cb}[1]{%
5 \setbox\z@\hbox{#1}%
6 \ifdim\dp\z@=\z@{\accent44 #1}%
7 \else{\accent96 #1}%
8 \fi}%

```

g k l n r s t

FIGURA 7: Il comma below prodotto con la primitiva `\accent` è insoddisfacente per colpa del posizionamento verticale

9 `\endinput`

Non c'è possibilità di controllare il posizionamento verticale, in questo modo. Se l'accento fosse disegnato appropriatamente non ci sarebbe questo problema, e il codice sarebbe perfetto. Un metodo del genere è dipendente dalla codifica. In caso nella codifica scelta ci fossero caratteri diversi da ' , ' in posizione 44 e ' ' ' in posizione 96, sarebbe diverso anche il carattere accentato risultante.

8 `combelow 0.99d`

L'ultima versione prodotta è molto più simile al codice di Beccari, ed è l'altra strada di `plain` relativamente al codice di `\c`. Il codice è molto compatto ma non elimina ogni eventuale problema di crenatura, così come la versione immediatamente precedente. Il sorgente è:

```

1 \NeedsTeXFormat{LaTeX2e}
2 \ProvidesPackage{combelow}
3   [2009/08/23 v0.99d comma below
   command]
4 \DeclareRobustCommand{\cb}[1]{%
5 \setbox\z@\hbox{#1}%
6 \ifdim\dp\z@=\z@
7   \oalign{\unhbox\z@\crcr
8   \hidewidth\lower.4ex
9   \hbox{,}\hidewidth}%
10 \else\oalign{\unhbox\z@
11   \crcr\hidewidth\raise.3ex
12   \hbox{'}\hidewidth}%
13 \fi}}%
14 \endinput

```

ed è quello usato per produrre il testo della figura 5.

9 Sillabazione e crenatura

Abbiamo accennato al problema di sillabazione scaturente dall'uso di `combelow` nel paragrafo 5.2. TUG UK (2009), e soprattutto BECCARI (2009), dedicano alcune pagine all'argomento; ne riassumiamo qui i punti salienti.

L'algoritmo di sillabazione di TEX non sillaba se trova un comando `\accent`. La cosa si può evitare per i pacchetti di macro di TEX usando un font codificato adeguatamente, ad esempio T1 di `fontenc` nel caso di LATEX (se codifichiamo il testo nella figura 4 in OT1 anziché T1 otteniamo un risultato uguale, a meno del segno diacritico, a quello della figura 5, che dimostra la veridicità di quanto appena detto). Questo succede perché in questa codifica

(e in altre, ma non OT1 a 128 caratteri) una lettera accentata è mappata nel glifo corrispondente, e questo permette all'algoritmo di procedere alla sillabazione. In base a KNUTH (2001) (pag. 46), le legature non soffrono di questo problema, e non confondono l'algoritmo di sillabazione di TEX.

La speranza di eliminare ogni problema legato alla codifica è riposta nei candidati successori di TEX (ad esempio LuaTEX), basati su Unicode. Ciò dovrebbe dare la ragionevole certezza che ogni carattere LATEX (almeno quelli esistenti in uno o più alfabeti) sarà mappato in un glifo valido e con tutte le informazioni relative alla sillabazione e alla crenatura.

Una soluzione alternativa, sempre propostami da Claudio Beccari, prevede l'uso dei font virtuali. Questi forniscono un mezzo per generare un singolo glifo mettendo insieme altri glifi, regole e comandi di composizione e informazioni per posizionare il tutto, in modo da specificare esattamente come tutto viene assemblato. Per iniziare a saperne di più sui font virtuali si può fare riferimento all'articolo di KNUTH (1993) ad essi relativo.

Un discorso simile può essere fatto per la crenatura, di cui il carattere composto non ha informazione. Se una codifica avesse le lettere con comma below, TEX potrebbe fare crenature esattamente come quelle delle lettere con la cediglia in codifica T1.

10 Carattere vietato in Unicode

Dopo aver scritto la prima bozza del volantino di cui al paragrafo 1, e prima di decidere di scrivere un pacchetto, ho pensato: «Perché non inserire il carattere adatto da un editor Unicode?». Detto, fatto. Ho compilato il volantino con l'opzione `utf8x` di `inputenc`, ma non ho ottenuto il risultato sperato.

Per ammissione dei due autori di MITTELBACH e ROWLEY (2006), il pacchetto in questione fornisce un supporto Unicode che non è tale. Di fatto fornisce un meccanismo per leggere i caratteri multibyte e una tabella di corrispondenze tra caratteri multibyte e comandi LATEX per ottenere i caratteri corrispondenti. Ecco perché non si ottiene alcun carattere correttamente accentato nel caso allo studio: semplicemente perché ancora non esiste, né esiste un simbolo almeno corrispondente all'accento comma below.

11 Conclusioni

Questo pacchetto, nella sua versione più robusta, potrebbe espandersi nel tempo e diventare un contenitore di tutti i segni diacritici "ignorati" da LATEX (un nome possibile è `diacrit`). Già ora si può (e si dovrebbe) modificare `babel` per usare `combelow`. In realtà proprio `babel` dovrebbe integrare il comando proposto, e gli eventuali altri di futura

realizzazione, per caricarli in base alla lingua selezionata, eventualmente sostituendo i comandi di default: se il comma below è una cediglia, si faccia in modo che `\c` venga ridefinito con il contenuto di `\cb`.

12 Sviluppi futuri

Nell'immediato futuro il pacchetto deve subire le seguenti modifiche, intenzionalmente migliorative: usare i comandi `LATEX \DeclareTextAccent` e `\DeclareTextCompositeCommand` per generare le lettere accentate, e trovare un modo (l'ideale sarebbe un algoritmo generale) per decentrare il segno diacritico in base alle lettere da accentare (come nel caso della 'r').

13 Ringraziamenti

Per questo lavoro devo ringraziare, per chi ancora non l'avesse capito, Claudio Beccari per le sue osservazioni, i suoi suggerimenti e curiosità. L'articolo non sarebbe stato molto significativo senza tutte le informazioni che mi ha proposto di studiare e aggiungere.

Un altro ringraziamento va agli anonimi revisori che hanno ulteriormente migliorato la qualità del lavoro.

Riferimenti bibliografici

- BECCARI, C. (2009). *Introduzione all'arte della composizione tipografica con LATEX*. GUIT, Pisa. URL <http://www.guit.sssup.it/download/#manual>. Versione 0.82.
- BRAAMS, J. (2006). *Babel, a multilingual package for use with LATEX's standard document classes*. URL <http://www.ctan.org>.
- ELJKHOUT, V. (1991–1993). *TEX by Topic. A TEXnician reference*. Addison-Wesley, UK. URL <http://www.ctan.org/tex-archive/info/texbytopic/>. L'autore ha riottenuto i diritti nel 2001, e da allora ha reso il libro disponibile sul web.
- KNUTH, D. E. (1993). «Virtual fonts: More fun for Grand Wizards». *TUGboat*, 11 (1). URL <http://www.tex.ac.uk/tex-archive/info/knuth/virtual-fonts>. Text version.
- (2001). «The TEXbook». In *Computers & typesetting / A*, Addison-Wesley, Boston, MA.
- KORPELA, J. K. (2006). *Unicode explained*. O'Reilly, Sebastopol, CA.
- MITTELBACH, F. e ROWLEY, C. (2006). *Providing some UTF-8 support via inputenc*. URL <http://www.ctan.org>.
- TUG UK (2009). «Faq». URL <http://www.tex.ac.uk/faq>.
- TYPO.CZ e DESIGNIQ.CZ (2009). «Diacritics project». URL <http://diacritics.typo.cz/>.
- UNICODE CONSORTIUM (2009a). «Code charts». URL <http://www.unicode.org/charts>.
- (2009b). «Combining Diacritical Marks». URL <http://www.unicode.org/charts/PDF/U0300.pdf>.
- (2009c). «Latin Extended A». URL <http://www.unicode.org/charts/PDF/U0100.pdf>.
- (2009d). «Latin Extended B». URL <http://www.unicode.org/charts/PDF/U0180.pdf>.
- WIKIPEDIA (2009a). «Cediglia». URL <http://it.wikipedia.org/wiki/Cediglia>.
- (2009b). «Cedilla». URL <http://en.wikipedia.org/wiki/Cedilla>.

▷ Gianluca Pignalberi
Free Software Magazine
g dot pignalberi at
freesoftwaremagazine dot com