

Sveučilište u Zagrebu
Prirodoslovno-matematički fakultet
Matematički odsjek

Maja Ribarić

Lokalizacija $\text{T}_{\text{E}}\text{X}$ -a za hrvatski književni jezik

Diplomski rad

Voditelj rada:
Prof. dr. sc. Nenad Antić

Zagreb, 29. kolovoza 2011.

Predgovor

U ovom radu opisan je postupak lokalizacije alata $\text{T}_{\text{E}}\text{X}$ za hrvatski književni jezik. Točnije, opisan je postupak kreiranja, instalacije i upotrebe novog formata `hrtex`, namijenjenog korištenju s 8-bitnim $\text{T}_{\text{E}}\text{X}$ sustavom `pdftex`. Korištenje formata `hrtex` daje dvije nove funkcionalnosti pri radu s alatom `pdftex`: omogućena je obrada datoteka s hrvatskim znakovima kodiranih u UTF-8 i uvedeni su novi, kvalitetnije sastavljeni obrasci za rastavljanje riječi na kraju retka, autora prof. Nenada Antonića.

S obzirom da su i korišteni obrasci za rastavljanje kodirani u UTF-8, može se reći da je postignuto usklađivanje tri ključna pojma: hrvatskih znakova, plain $\text{T}_{\text{E}}\text{X}$ -a i UTF-8. Kako je kodiranje UTF-8 već neko vrijeme službeni međunarodni standard, a svakim danom postaje sve prisutniji i u praksi, ovaj je rad napravljen u nadi da će pospješiti korištenje (modificiranog) plain $\text{T}_{\text{E}}\text{X}$ -a s novijim hrvatskim datotekama. Time bi bila dana prihvatljiva alternativa alatima poput $\text{L}^{\text{A}}\text{T}_{\text{E}}\text{X}$ -a, koji u nekim slučajevima ne omogućuju postizanje svih željenih opcija pri slaganju dokumenata.

K tome, opisane su podloge na kojima je utemeljen rad. Ukratko je opisan program $\text{T}_{\text{E}}\text{X}$, s naglaskom na one aspekte koji su bili promatrani i modificirani pri izradi ovog rada; standard za kodiranje UTF-8 i njegova usporedba s nekim drugim kodiranjima; $\text{T}_{\text{E}}\text{X}$ -ov paket `encT_{\text{E}}\text{X}` kojim je, između ostalog, omogućeno korištenje čeških i slovačkih znakova u UTF-8; $\text{T}_{\text{E}}\text{X}$ -ov paket za rastavljanje riječi na kraju retka `hyph-utf8` i, naposljetku, skupina fontova Latin Modern, koja sadrži sve znakove korištene u europskim jezicima s latiničnim pismom.

Sadržaj

I. O \TeX-u	4
II. Kodiranje znakova: UTF-8	6
III. Latin Modern fontovi	10
IV. Rastavljanje riječi na kraju retka	14
4. 1. Opis algoritma	14
4. 2. Rastavljanje riječi u hrvatskom jeziku	14
4. 3. Pravila o prenošenju dijela riječi u sljedeći red	15
4. 4. Paket <code>hyph-utf8</code>	16
V. Paket <code>enc\TeX</code>	24
5. 1. Osnovne informacije	24
5. 3. Konverzija bajt-za-bajt	25
5. 3. Višebajtna konverzija	25
5. 4. Sažetak naredbi i opcija	32
VI. Opis kreiranja formata <code>hrtex</code>	36
6. 1. Postupak instalacije i korištenja formata <code>hrtex</code>	37
Literatura	62
Sažetak	63
Summary	65
Životopis	67

I. O T_EX-u

T_EX je računalni sustav za slovoslagarstvo namijenjen prvenstveno slaganju knjiga s mnogo matematičkog sadržaja. Pri slaganju rukopisa pomoću T_EX-a, računalo se instruiira kako uneseni tekst transformirati u dokumente vrhunske tipografske kvalitete.

Samo ime T_EX dolazi od grčkog korijena $\tau\epsilon\chi$, od kojeg potječu hrvatske riječi kao što je *tehnologija*. T_EX je razvio američki matematičar Donald Knuth početkom osamdesetih godina 20. stoljeća. Razvio je i prateći jezik METAFONT koji služi za definiranje fontova, a i T_EX i METAFONT napisani su pomoću sustava za programiranje naziva WEB stvorenog upravo u tu svrhu. Moderne distribucije T_EX-a služe se programom Web2C za konvertiranje izvornog WEB koda u programski jezik C.

Nakon objavljivanja prve verzije T_EX-a, koji je slobodni softver, neovisno o Donaldu Knuthu nastalo je mnogo varijanti i ekstenzija za T_EX koje daju nove funkcionalnosti temeljno istom programu. S obzirom da je originalni program namijenjen obradi teksta u ASCII kodiranju, u kojem se od slovnih znakova nalaze samo znakovi engleske abecede, mnoge su od tih ekstenzija uvodile rad s nekim drugim kodiranjem (ili više njih), koje bi podržavalo znakove nekog drugog jezika ili većeg jezičnog područja. Jedna od takvih ekstenzija je i encT_EX, po uzoru na kojeg je kreiran format *hrtex*. Više o encT_EX-u može se naći u poglavlju 5.

Noviji alati vezani uz T_EX ne smiju se zvati tim imenom. Prema Donaldu Knuthu, kao i načelima slobodnog softvera, nužno je da je ime programa usko povezano s njegovom funkcionalnošću, pa se T_EX-om smije zvati samo originalni program s manjim ekstenzijama koje je naknadno dodao sam Knuth. Nakon prve varijante programa napisane 1978. godine i druge, koja je objavljena 1982. i implementirana neovisno o prvoj varijanti, godine 1989. objavljena je i treća varijanta T_EX-a u kojoj je glavna promjena bila omogućavanje rada s 8-bitnim ulazom. To je ujedno bila posljednja verzija u kojoj su donesene značajne promjene. Verzije koje su uslijedile su simbolično imenovane aproksimacijama broja π ; sa svakom novom verzijom dodana je još jedna decimala. To odražava činjenicu da je T_EX stabilan sustav u kojem se očekuju samo manje promjene. Posljednja stabilna verzija T_EX-a potječe iz 2008. godine i nosi oznaku 3.1415926.

Osnovno svojstvo koje T_EX razlikuje od drugih alata za računalno slovoslagarstvo jest odvojenost *rukopisa* (T_EX datoteke) i konačne verzije dokumenta. U programima kao što su MS Word, OpenOffice Writer i LibreOffice korisnik unosi i uređuje tekst pomoću grafičkog sučelja koje istodobno prikazuje kako će pri ispisivanju izgledati dokument koji se uređuje. Kod uređivanja dokumenata T_EX-om, korisnik u tekstualno sučelje unosi tekst dokumenta zajedno s potrebnim T_EX naredbama za definiranje izgleda izlaznog dokumenta. Stoga se T_EX može promatrati i kao programski jezik u kojem se posebnim naredbama (*kontrolnim nizovima*) zadaje krajnji oblik unesenog teksta.

Tekst i naredbe spomenutog programskog jezika unosimo u datoteke s ekstenzijom *.tex*, koje se potom obrađuju npr. alatom *pdftex*. Naredba u T_EX-u sastoji se od *escape znaka*, iza kojeg neposredno slijedi niz znakova koji označavaju određenu predefiniranu funkcionalnost koja se treba izvršiti. U svrhu konzistentnosti između raznih korisnika i verzija sustava, kao *escape* znak se skoro uvijek uzima tzv. *backslash*, tj. znak \backslash . Uz originalne naredbe i naredbe definirane u raznim ekstenzijama T_EX-a, korisniku je omogućeno i definiranje vlastitih naredbi.

Obradom datoteke alatom *tex* nastaje datoteka u posebnom formatu ekstenzije *.dvi* (Device Independent Format). Radi se o formatu koji sadrži vrlo jednostavan i apstraktan opis stranica koje se trebaju ispisati. S obzirom da je u vrijeme nastajanja T_EX-a svaki računalni zaslon i svaki pisač koristio vlastite računalne kodove, *.dvi* format,

jedinstven za sve računalne sustave, bio je potreban kao prijelazni korak između izlaza T_EX-a i pojedinačnih *drivera* koji bi implementirali ispis tog izlaza na različitim zaslonima i pisačima.

Nekoliko godina nakon nastanka T_EX-a, napisan je programski jezik PostScript, kojim je eliminirana potreba za različitim *driverima* za svaku vrstu uređaja. Uskoro je nastao i *dvips*, alat za konverziju DVI datoteka u PostScript dokument. Nakon stvaranja datoteke u DVI formatu, pomoću programa *dvips* iz te se datoteke dobiva gotov dokument u formatu PostScript (*.ps*), koji je spreman za ispis.

T_EX-ov zadatak pri obradi dokumenata za ispis ne obuhvaća baratanje samim oblicima slova. Za fontove koje T_EX koristi može se reći da su *prazni*, u smislu da ne sadržavaju konkretne oblike slova (glifove). S obzirom da za potrebe slaganja dokumenta nisu potrebni oblici slova, već samo njihove dimenzije, T_EX se u tu svrhu služi posebnim datotekama u T_EX Font Metrics formatu (*.tfm*), koje za svaki pojedini font sadrže njegovu visinu, širinu, dubinu itd. Potom gore spomenuti DVI *driveri* obavljaju zadatak popunjavanja tako dobivenog *kostura* dokumenta za ispis konkretnim glifovima odgovarajućih dimenzija.

Ukratko, T_EX pomoću TFM podataka za fontove kreira DVI datoteku, koja se pomoću programa *dvips* konvertira u PostScript. Pri korištenju alata *pdftex*, za koji je namijenjen novi format *hrtex*, dijelovi tog procesa se automatiziraju, te se pozivom jedne korisničke naredbe iz *.tex* datoteke automatski kreira *.pdf* dokument.

Osim samog T_EX-a, treba spomenuti i ϵ -T_EX, proširenje objavljeno 1996. godine koje uvodi nove naredbe, povećava maksimalnu vrijednost nekih starih T_EX registara te, između ostalog, omogućava kombiniranje tekstova pisanih slijeva nadesno s tekstovima pisanimi zdesna nalijevo. ϵ -T_EX je moguće neometano koristiti i za obradu datoteka pisanih za T_EX program, s obzirom da sadrži opciju za kompatibilni način rada. Ovo je proširenje ugrađeno u *hrtex* pri kreiranju formata.

II. Kodiranje znakova: UTF-8

Sustav kodiranja znakova jest kodiranje elemenata određenog skupa znakova npr. nizom prirodnih brojeva ili električnim impulsima, u svrhu omogućavanja prijenosa podataka preko telekomunikacijskih mreža ili pohrane podataka u računalu.

Prvi standard za računalno kodiranje znakova, ASCII (*American Standard Code for Information Interchange*), donesen je preliminarno 1963. godine, a obnovljena verzija s malim tiskanim slovima donesena je 1967. godine. ASCII je donijela Američka udruga za standarde (*American Standards Association*) zbog rastućih potreba telekomunikacijske industrije za takvim standardom. ASCII tablica znakova sadrži velika i mala slova engleske abecede, kontrolne i interpunkcijske znakove, te sveukupno sadrži 128 znakova. Nedugo nakon objave ASCII kao američkog standarda, u nekim su se zemljama počeli koristiti lokalizirani standardi utemeljeni na ASCII standardu, u kojima su određeni znakovi u originalnoj tablici zamijenjeni drugim, lokalno korisnijim znakovima (npr. u nekim europskim zemljama kôd za simbol dolara koristio se kao kôd za simbol funte ili lire). Tako je nastalo više lokaliziranih ASCII kodiranja koji su međusobno bili nekonzistentni. Usto, postojao je problem otežanog unosa određenih znakova: npr., za dobivanje naglašenih slova bilo je potrebno unositi kombinacije slova i posebnih znakova.

Originalni T_EX program razvijen je u Sjedinjenim Američkim Državama i radio je s datotekama u ASCII kodiranju. Kako kod T_EX-a, tako i kod drugih tehnologija, ASCII je dugo bio najčešće korišten standard za kodiranje. Varijanta ASCII kodiranja najbliža originalnoj (nazvana US-ASCII zbog jedinstvenosti naziva) bila je najkorištenije kodiranje na World Wide Web-u sve do 2007. godine, kad ju je prestiglo kodiranje UTF-8.

Važno je spomenuti i skupinu kodiranja ISO 8859, prije svega kodiranje ISO 8859-2. ISO, ili Međunarodna organizacija za standardizaciju (*International Organization for Standardization*), nedugo nakon donošenja standarda ISO 2022 koji je omogućio kombiniranje raznih kodiranja u jednom toku podataka, počela je definirati kodiranja koja bi nadopunjavala ASCII s dodatnih 96 pozicija. Tako je nastala skupina kodiranja ISO 8859. U toj skupini sadržano je više kodiranja za podršku raznim skupinama jezika. Poimence, glavno kodiranje u skupini, ISO 8859-1, pruža podršku za zapadnoeuropske jezike. ISO 8859-2 pruža podršku za istočnoeuropske jezike (između ostalog i hrvatski), ISO 8859-3 za neke južnoeuropske jezike, a ISO 8859-4 za sjevernoeuropske jezike. ISO 8859-5 (ili *ISO Cyrillic*) pruža podršku za neke jezike pisane ćiriličnim pismom, ISO 8859-6 (ili *ISO Arabic*) pokriva arapsko pismo, ISO 8859-7 (ili *ISO Greek*) pokriva grčko pismo, ISO 8859-8 (ili *ISO Hebrew*) pokriva pismo modernog hebrejskog jezika, i naposljetku ISO 8859-11 (ili *ISO Thai*) pokriva tajlandsko pismo. K tome, postoje i kodiranja ISO 8859-9/10/13/14/15/16 koja su obnovljene i dopunjene verzije nekih gore spomenutih kodiranja. Ova je skupina kodiranja, osim iz razloga raširene upotrebe, značajna prvenstveno zbog sadržavanja ISO 8859-1, koje je postalo standardno kodiranje za Unix sustave, te je kompatibilno s ASCII i UTF-8 kodiranjem.

Od kodiranja iz ove skupine, za hrvatski jezik važno je, naravno, kodiranje ISO 8859-2, koje se naziva i *Latin-2*. U njemu se nalaze svi znakovi potrebni za pisanje tekstova na hrvatskom jeziku (uz znakove prisutne u ISO 8859-1, to su još i Č, č, Ć, ć, Đ, đ, Š, š, Ž, ž). Kao i ostatak skupine ISO 8859, ovo je kodiranje 8-bitno, što znači da se svaki znak kodira nekim od brojeva prikazivih s do 8 binarnih znamenki. To ukupno daje 256 pozicija u tablici znakova.

2004. godine organizacija ISO raspustila je radnu skupinu za održavanje 8-bitnih kodiranja. Na području kodiranja znakova, ISO se danas fokusira na *Universal Character Set* – UCS (Univerzalni skup znakova), te na kodiranja koja daju punu podršku za UCS,

ponajprije UTF-8.

UTF-8 (*UCS Transformation Format – 8-bit*) je višebajtno kodiranje za skup znakova UCS. UTF-8 kodira ukupno 1112064 elemenata iz UCS koristeći jedan do četiri bajta (koji se nazivaju *oktetima* u Unicode standardu). Elementi s nižom numeričkom vrijednošću, koji se češće pojavljuju u praksi, kodiraju se manjim brojem bajtova, što ovu shemu kodiranja čini efikasnom (znakovi čiji kodovi zauzimaju više bajtova su rjeđi, što smanjuje ukupnu veličinu datoteke s potrebnim sadržajem). Prvih 128 znakova u UCS skupini identično je ASCII kodiranju, što UTF-8 čini kompatibilnim sa starijim datotekama u ASCII formatu. Tih je 128 znakova kodirano s po jednim bajtom, numeričkih vrijednosti jednakih odgovarajućim ASCII vrijednostima.

Konkretnije, kodovi znakova u UTF-8 kodiranju na pozicijama prvih nekoliko bitova prvoga bajta imaju upisanu vrijednost koja određuje kolika je duljina trenutnog koda. Ako prvi bit ima vrijednost 0, to znači da se radi o kodu koji se sastoji od jednog bajta, i preostalih 7 bitova ujedno čini ASCII vrijednost za odgovarajući znak.

Ako prvi bajt počinje kombinacijom 110, to znači da se radi o dvobajtnom kodu, pri čemu drugi bajt također ima predefinirani početak – kombinaciju 10. Ako prvi bajt počinje kombinacijom 1110, radi se o trobajtnom kodu, pri čemu svaki sljedeći bajt počinje kombinacijom 10, itd. Svi znakovi potrebni za računalnu obradu tekstova na hrvatskom jeziku kodirani su 1-bajtnim ili 2-bajtnim kodovima.

UTF-8 ima mnoge prednosti pred drugim kodiranjima. Tu je već spomenuta kompatibilnost s ASCII kodiranjem, što konverziju starijih aplikacija i datoteka u UTF-8 čini lakšim od konverzije u druga kodiranja. Zatim, UTF-8 ne zahtijeva oznaku poretka bajtova, te je jedino kodiranje koje ne zahtijeva indikaciju kodiranja pri radu s XML datotekama. Također, računalnom analizom je lako određivo predstavlja li određeni niz bajtova validni UTF-8 kôd. To olakšava automatsku identifikaciju kodiranja u slučaju kad nedostaje konkretna specifikacija.

Mnogobrojne prednosti uzrokovale su veliku raširenost ovog kodiranja u primjeni. Danas više od pola internet stranica koristi upravo UTF-8, a nadležne organizacije zahtijevaju obaveznu podršku za UTF-8 za sve internet protokole, klijente za elektroničku poštu i sl.

III. Latin Modern fontovi

Za potrebe T_EX-a Knuth je dizajnirao fontove naziva Computer Modern (CM). U formatu plain T_EX standardno je dostupno samo šesnaest CM fontova koji podržavaju ispis u običnom, *nakošenom*, *italiziranom*, **podebljanom**, **strojopisačem**, matematičkom *formatu* i formatu za neke matematičke simbole. To su fontovi: CM Roman 10/7/5pt, CM Bold Extended 10/7/5pt, CM Slanted Roman 10pt, CM Text Italic 10pt za običan tekst, odnosno CM Typewriter Type 10pt, CM Math Italic 10/7/5pt, CM Math Symbols 10/7/5 pt, CM Math Extension 10pt za specijalni tekst, tj. za matematičke formule, uredni prikaz programskih kodova i sl. Osim fontova inicijalno dostupnih u plain T_EX-u, u dokumentu je moguće koristiti bilo koji font za koji postoje potrebne datoteke na računalu. U tu svrhu koristi se naredba `\font\oznakaFonta=imeFontaURačunalu`. Tom naredbom T_EX dobiva informacije potrebne za ispisivanje teksta u željenom fontu, te je moguće označavati dijelove teksta uobičajenom naredbom `{\oznakaFonta željeniTekst}`.

Znakovi s naglascima predstavljali su izazov od početka raširenije uporabe računala. Svatko tko mora komunicirati s drugim računalnim sustavom na nekom jeziku osim engleskog zna da korištenje tzv. *čudnih znakova* zna biti problematično. Za T_EX korisnike, od velike je važnosti bilo donošenje konzistentnog kodiranja znakova, kao i proširivanje trenutnih CM fontova većim rasponom znakova. Veliki korak naprijed u smjeru sređivanja kaosa s raznim kodiranjima bio je Unicode standard (ISO/IEC 10646), objavljen 1993. Unicode, naravno, ne rješava sve probleme u vezi fontova, te čak dodaje nekoliko specifičnih, novih problema (npr., problemi s veličinom fontova ili s registracijom nestandardnih znakova i jezika). Unatoč tome, za vjerovati je da će situacija na planu računalnog komuniciranja postati bolja kad Unicode postane standard *de facto*, a ne samo *de nomine*.

Nakon što je uočena potreba za skupinom fontova nalik Computer Modern, koja bi podržavala veći raspon znakova za potrebe T_EX korisnika diljem svijeta, razvilo se više paketa koji su u većoj ili manjoj mjeri bili utemeljeni na Computer Modern fontovima, te su omogućavali kvalitetno korištenje raznih dijakritičkih i drugih znakova specifičnih za različita jezična područja. Posebno je važno izdvojiti skupine fontova CM Super i Latin Modern. CM (Computer Modern) Super omogućuje korištenje skoro svih znakova svih alfabeti svijeta. S druge strane, Latin Modern fontovi su primarno usredotočeni na europske jezike, iako pokrivaju i druge, no veće su kvalitete ispisa od CM Super fontova. Zato je u ovom radu odlučeno pokušati prilagoditi Latin Modern fontove u svrhu rada s hrvatskim znakovima.

U ranijim godinama, kad je T_EX bio opremljen samo skupinom fontova Computer Modern, rijetki korisnici T_EX-a koji su se koristili naglašenim slovima bili su primorani upotrebljavati temeljnu naredbu `\accent`, te se pretpostavljalo da će to rješenje biti dovoljno za tu svrhu. Rast popularnosti T_EX-a u zemljama u kojima se koristi mnogo dijakritičkih znakova učinio je tu pretpostavku nevažecom. Može se za to navesti barem tri razloga:

(1) naglašeni znakovi ne ponašaju se poput *normalnih* znakova, npr., ometaju važne T_EX-ove algoritme poput rastavljanja riječi na kraju retka i umetanja razmaka između slova;

(2) CM fontovi ne sadrže sve potrebne dijakritike, npr. nedostaje dijakritički znak ogonek, koji se koristi u poljskom, litvanskom i Navajo jeziku;

(3) kad se dijakritički elementi poput cedille i ogoneka obrađuju kao naglasci, oni se preklapaju s tekstom, što ometa neke primjene, npr. pripremanje tekstova za prijelom. Nameće se zaključak da je potrebno proširiti skupinu CM dodavanjem dijakritičkih slova.

Kao rješenje tog problema nastala je skupina fontova Latin Modern (LM).

2002. godine, tijekom T_EX sastanka u Bachoteku, u Poljskoj, predstavnici europskog društva korisnika T_EX-a, nakon raspravljanja problematike putem elektroničke pošte, osmislili su prijedlog konvertiranja virtualnih fontova AE Larsa Engebretsenena u univerzalniji PostScript Type 1 format, uz proširivanje tih fontova skupom potrebnih dijakritičkih znakova. Uskoro se krenulo u provedbu te zamisli i kreirana je nova skupina fontova.

Originalna namjera bila je sačuvati naziv AE, jer se željelo naglasiti značaj Engebretsenove ideje za taj pothvat. Međutim, uskoro je postalo jasno da bi razlike bile fundamentalne i da je promjena naziva potrebna kako bi se izbjegla zbrka s imenima. Stoga je smišljen naziv *Latin Modern*.

Trenutno, LM skupina fontova sastoji se od 72 fonta za tekst i 20 matematičkih fontova. Popis svih dostupnih fontova može se naći na kraju ovog poglavlja. Iako ne postoji analogon za svaki LM tekstualni font među CM fontovima i obratno, svi LM matematički fontovi pripadaju CM standardu (cmbsy7, cmbsy10, cmmib5 i cmmib7 dodalo je Američko matematičko društvo).

Za omogućavanje pristupa velikoj količini dijakritičkih znakova, pripremljeno je devet različitih 255-znakovnih kodiranja za svaki font:

- cs (CS TUG; cs-*.tfm),
- ec (Cork ili t1; ec-*.tfm),
- qx (GUST; qx-*.tfm)
- rm (*regular math*, korišten u ot1 i ot4; rm-*.tfm)
- l7x (*Lithuanian*; l7x-*.tfm),
- y&y-ov TeX'n'ANSI (ili ly1; texnansi-*.tfm),
- t5 (*Vietnamese*; t5-*.tfm),
- Text Companion for ec (fonts ili ts1; ts1-*.tfm)

Za sva navedena kodiranja postoji i podrška za L^AT_EX. Pri kreiranju formata *hrtex*, posebno je proučavano kodiranje CS TUG, vezano za češki i slovački jezik, koje je zbog jezičnih sličnosti bilo najbliže željenom rješenju za hrvatski jezik. Same datoteke s fontovima (*cs-lm*.tfm*) iskorištene su kao konkretna realizacija iza poveznica kojima su načinjene hrvatske verzije *.tfm* datoteka, što je detaljnije opisano u poglavlju 6.

Popis fontova za uobičajeni tekst u skupini Latin Modern:

- lmb10 – LM Roman Demi, Regular, 10 točaka
- lmbo10 – LM Roman Demi, Oblique, 10 točaka
- lmbx5 lmbx6 lmbx7 lmbx8 lmbx9 lmbx10 lmbx12 – LM Roman, Bold
- lmbxi10 – LM Roman, BoldItalic, 10 točaka
- lmbxo10 – LM Roman Slanted, Bold, 10 točaka
- lmcsc10 – LM Roman Caps, Regular, 10 točaka
- lmcsc010 – LM Roman Caps, Oblique, 10 točaka
- lmdunh10 – LM Roman Dunh, Regular, 10 točaka
- lmduno10 – LM Roman Dunh, Oblique, 10 točaka
- lmr5 lmr6 lmr7 lmr8 lmr9 lmr10 lmr12 lmr17 – LM Roman, Regular
- lmri7 lmri8 lmri9 lmri10 lmri12 – LM Roman, Italic
- lmro8 lmro9 lmro10 lmro12 lmro17 – LM Roman Slant, Regular
- lmss8 lmss9 lmss10 lmss12 lmss17 – LM Sans, Regular
- lmssbo10 – LM Sans, BoldOblique, 10 točaka
- lmssbx10 – LM Sans, Bold, 10 točaka

- lmssdc10 – LM Sans Demi Cond, Regular, 10 točaka
- lmssdo10 – LM Sans Demi Cond, Oblique, 10 točaka
- lmssso8 lmssso9 lmssso10 lmssso12 lmssso17 – LM Sans, Oblique
- lmssq8 – LM Sans Quot, Regular, 8 točaka
- lmssqbo8 – LM Sans Quot, BoldOblique, 8 točaka
- lmssqbx8 – LM Sans Quot, Bold, 8 točaka
- lmssqo8 – LM Sans Quot, Oblique, 8 točaka
- lmtcsc10 – LM Mono Caps, Regular, 10 točaka
- lmteso10 – LM Mono Caps, Oblique, 10 točaka
- lmtk10 – LM Mono Lt, Bold, 10 točaka
- lmtko10 – LM Mono Lt, BoldOblique, 10 točaka
- lmtl10 – LM Mono Lt, Regular, 10 točaka
- lmtlc10 – LM Mono Lt Cond, Regular, 10 točaka
- lmtlco10 – LM Mono Lt Cond, Oblique, 10 točaka
- lmtlo10 – LM Mono Lt, Oblique, 10 točaka
- lmtt8 lmtt9 lmtt10 lmtt12 – LM Mono, Regular
- lmtti10 – LM Mono, Italic, 10 točaka
- lmtto10 – LM Mono Slant, Regular, 10 točaka
- lmu10 – LM Roman Unslanted, Regular, 10 točaka
- lmvtk10 – LM Mono Prop Lt, Bold, 10 točaka
- lmvtko10 – LM Mono Prop Lt, BoldOblique, 10 točaka
- lmvtl10 – LM Mono Prop Lt, Regular, 10 točaka
- lmvtlo10 – LM Mono Prop Lt, Oblique, 10 točaka
- lmvtt10 – LM Mono Prop, Regular, 10 točaka
- lmvtto10 – LM Mono Prop, Oblique, 10 točaka

Popis fontova za matematički tekst u skupini Latin Modern:

- lmbsy5 lmbsy7 lmbsy10 – LM Math Symbols, Bold
- lmex10 – LM Math Extension, Regular, 10 točaka
- lmmi5 lmmi6 lmmi7 lmmi8 lmmi9 lmmi10 lmmi12 – LM Math Italic, Regular
- lmmib5 lmmib7 lmmib10 – LM Math Italic, Bold
- lmsy5 lmsy6 lmsy7 lmsy8 lmsy9 lmsy10 – LM Math Symbols, Regular

IV. Rastavljanje riječi na kraju retka

T_EX-ov mehanizam za rastavljanje riječi na kraju retka je među najstarijima i najkvalitetnijima. Prilagođen je i za upotrebu s drugim programima, pa se može koristiti i kod obrade teksta programom OpenOffice.org. Ključni je korak nalaženje obrazaca za rastavljanje pomoću kojih se pri prelamanju teksta u linije može odrediti kada je dopustivo prelomiti liniju usred riječi. Takvim se postupkom postiže kvalitetan razmak između pojedinih riječi; razmaci su ujednačeni i razumnih proporcija u odnosu na ostale dimenzije teksta.

1. Opis algoritma

Pretpostavimo da su dani obrasci za rastavljanje na određenom jeziku. Obrasci se sastoje od niza kombinacija znakova (slova i brojki) koje predstavljaju informacije o mogućnostima rastavljanja. Naišavši na riječ koju je poželjno rastaviti, T_EX tu riječ najprije potraži u rječniku iznimki. Za engleski, taj se rječnik sastoji od nekoliko stotina važnijih riječi koje sam algoritam ne uspijeva rastaviti, odnosno, ne mogu se rastaviti korištenjem obrazaca. Korisnik po potrebi može i sam zadati pojedine riječi koje se trebaju rastavljati na specifičan način, i to naredbom `\hyphenation` u `.tex` datoteci sa željenim dokumentom, prije unosa samog teksta. Ako se riječ ne nalazi u rječniku iznimki, proširuje se posebnim znakovima za kraj i početak riječi (.) i rastavlja na *podriječi* duljine 1, 2, 3, itd. Ovisno o duljini riječi n , tako se dobiva $n + 2$ podriječi duljine 1, $n + 1$ podriječi duljine 2, n podriječi duljine 3, itd. Zatim se podriječi traže među obrascima za rastavljanje. Za svako mjesto između dva slova određujemo *međuslovnu vrijednost* na sljedeći način: promatramo sve dobivene obrasce koji su relevantni za ta dva slova (tj. sadrže oba slova, ili završavaju lijevom slovom, ili počinju desnim). U svakom od tih obrazaca nalazi se određena međuslovna vrijednost za ta dva slova. Sustav tada računa maksimum svih tako dobivenih vrijednosti. Dobiveni maksimum jest međuslovna vrijednost.

Isti se postupak provodi za sve parove susjednih slova (u što uključujemo i oznake za početak i kraj riječi). Naposljetku, dobivamo međuslovne vrijednosti za sve pozicije među slovima, koje se mogu prikazati na sljedeći način: `.0h0y3p0h0e2n5a4t2i0o0n0`. Konačno, rastavljanje između dva slova se smatra prihvatljivim ako je odgovarajuća međuslovna vrijednost neparna. U gore prikazanom slučaju, riječ se može rastaviti na sljedećim mjestima: `hy-phen-ation`.

Originalni T_EX-ov algoritam za engleski jezik koristi 4447 obrazaca za rastavljanje, u kojem se sve međuslovne vrijednosti nalaze između 0 i 5. Velike neparne vrijednosti, npr. 5, navode na rastavljanje na poželjnim točkama (npr. obrazac `.0h5e0l0o0` vodi na rastavljanje riječi `bach-e-lor`), a velike parne vrijednosti poput 4 sprječavaju nepoželjna rastavljanja (npr. obrazac `.0h0a0c0h4` sprječava rastavljanja tipa `toothach-es`).

Usto, T_EX-ov algoritam daje veću težinu češćim riječima koje se provjereno kvalitetno rastavljaju, pa je u konačnici rastavljanje vrlo uspješno, nalazi većinu prihvatljivih točaka za rastavljanje te gotovo nikad ne krši pravopisna pravila.

2. Rastavljanje riječi u hrvatskom jeziku

Rastavljanje riječi u hrvatskom jeziku dosad se u T_EX-u izvršavalo prema obrascima za rastavljanje autora Igora Marinovića iz 1996. godine. Ti su obrasci pri stvaranju paketa `hyph-utf8` prilagođeni i uključeni u paket kako je opisano u Poglavlju 6. U novom formatu `hrtex` odlučeno je uvesti nove obrasce za rastavljanje, čiji je autor voditelj ovog

rada, prof. dr. sc. Nenad Antić. Ti su obrasci dobiveni novim pristupom, primjenom pravila iz [1], koja su navedena u nastavku. Novi su obrasci u UTF-8 kodiranju, kao i originalna datoteka u paketu hyph-utf8.

3. Pravila o prenošenju dijela riječi u sljedeći red

1. Svaki fonem ne može stajati uz svaki fonem niti svaki fonem može stajati na svakom mjestu u riječi. Niti su moguće sve kombinacije fonema niti se sve kombinacije fonema mogu naći na svakom mjestu u riječi. Neke su kombinacije fonema moguće na početku, neke u sredini, a neke na kraju riječi.

2. Na početku riječi mogu stajati svaki suglasnički fonem sa svakim samoglasničkim fonemom i svaki samoglasnički fonem sam. Od dvočlanih kombinacija suglasničkih fonema sa suglasničkim fonemima na početku riječi mogu stajati, naprimjer, ove: **bl** (blago), **blj** (bljutav), **br** (brati), **bj** (bježati), **cj** (cjevast), **cr** (crijevo), **dl** (dlaka), **dr** (drama), **dj** (djever), **fl** (flanel), **fr** (fratar), **fj** (fjaka), **gm** (gmizati), **gn** (gnati), **gnj** (gnjaviti), **gl** (glava), **glj** (gljiva), **gr** (grana), **gv** (gvožđe), **hm** (hmelj), **hl** (hladan), **hr** (hrana), **hv** (hvala), **km** (kmet), **kn** (knez), **knj** (knjiga), **kl** (klečati), **klj** (kljuvati), **kr** (krov), **kv** (kvasac), **lj** (ljepota), **mn** (mnogo), **ml** (mladost), **mlj** (mljaskati), **mr** (mreža), **mj** (mjesto), **nj** (nježan), **pl** (plesati), **plj** (pljuskati), **pr** (pričati), **pn** (pneumatik), **pj** (pjevati), **sm** (smion), **sn** (snažan), **sl** (sloboda), **slj** (sljubiti), **sr** (sreća), **sv** (svanuti), **sj** (sjever), **šlj** (šljiva), **šč** (ščepati), **šč** (ščućuriti), **šr** (šrapnel), **šv** (švelja), **tm** (tmast), **tl** (tlak), **tr** (trava), **tv** (tvornica), **tj** (tjestenina), **zm** (zmaaj), **zn** (znanje), **zl** (zloba), **zr** (zraka), **zv** (zvono), **zj** (zjenica), **žm** (žmiriti), **žnj** (žnjeti), **žl** (žlica), **žv** (žvale).

3. Od tročlanih kombinacija suglasničkih fonema sa suglasničkim fonemima moguće su, naprimjer, ove: **sml** (smlaviti), **smr** (smračiti), **smj** (smjelost), **shr** (shrvati), **shv** (shvatiti), **skl** (skladan), **sklj** (skljokati se), **snj** (snježan), **skr** (skriviti), **skv** (skvičati), **spl** (splasnuti), **splj** (spljuskati), **spr** (spretan), **spj** (spjevati), **stl** (stlačiti), **str** (strašan), **stv** (stvoriti), **stj** (stjecati), **zbl** (zbližiti), **zbr** (zbrojiti), **zbj** (zbježati), **zdj** (zdjelica), **zdr** (zdravlje), **zgl** (zglavak), **zgr** (zgrabiti), **zgn** (zgnati), **blj** (bljeskati), **dlj** (dljetast), **gnj** (gnjevan), **hlj** (hljebbar), **mlj** (mljeti), **plj** (pljeskati), **sljepoća** (sljepoća), **žlj** (žljebast).

4. U sredini riječi mogu biti i neke druge kombinacije. Isto tako i na kraju riječi.

5. Kombinacije koje mogu stajati na početku riječi mogu stajati i na početku sloga.

6. Slog mogu činiti samoglasnički fonemi i samoglasnički u kombinaciji sa suglasničkim fonemima.

7. Samoglasnički fonem sam za sebe može činiti slog na početku riječi ako je iza njega samoglasnički fonem, odnosno suglasnički i samoglasnički fonem (**i-a-ko**, **u-o-či**, **a-ko**, **e-vo**, **i-pak**, **o-pet**, **u-to**), na kraju riječi ako je ispred njega samoglasnički fonem (**ra-di-o**) te na bilo kojem mjestu riječi ako je ispred njega samoglasnički, a iza njega samoglasnički fonem, odnosno suglasnički (glasnik) i samoglasnički fonem (**pi-sa-o**, **ra-di-o**, **ra-di-oni-ca**).

8. Suglasnički fonemi ni sami za sebe ni u međusobnim kombinacijama (tj. bez samoglasničkih fonema) ne mogu činiti slog.

9. Iz takvih položaja fonema izvode se pravila o prenošenju dijela riječi u sljedeći red.

10. U sljedeći red prenosi se onaj dio riječi koji sam za sebe može činiti slog veći od jednog znaka. Tako se u sljedeći red može prenositi:

- a) samoglasnički fonem (**radi-onica**);

b) suglasnički i samoglasnički fonem ako su na kraju riječi, odnosno ako su iza njih suglasnički i samoglasnički fonem (**profeso-ri** ili **profeso-rima**);

c) dvočlane ili tročlane kombinacije suglasničkih i samoglasničkih fonema ako su na kraju riječi, odnosno ako je iza njih samoglasnički fonem koji sam za sebe može činiti slog, suglasnički i samoglasnički fonem te dvočlane ili tročlane kombinacije suglasničkih i samoglasničkih fonema (**ista-kli** uz **istak-li**, **sta-klenik** uz **stak-lenik**, **se-strin** uz **ses-trin** i **sest-rin**).

11. Ako se unutar riječi nađu jedan pored drugog suglasnički fonemi koji ne mogu biti na početku riječi (pa onda ni na početku sloga), riječ se rastavlja ovako: suglasnički fonem koji onemogućava kombinaciju suglasničkih fonema na početku riječi (pa onda i na početku sloga) pridružuje se prethodnom slogu (prethodnom dijelu riječi), a suglasnički fonem koji to omogućava slijedećem slogu (slijedećem dijelu riječi): **or-mar**, **od-bit**, **od-brojiti**, **ot-platiti** itd.

12. Kako suglasnički fonemi ne mogu sami činiti slogove ni pojedinačno ni u međusobnim kombinacijama, ne prenose se u slijedeći red sami, nego sa slogom, odnosno kombinacijom kojoj pripadaju. Stoga se riječi tipa **kontakt** ne mogu rastavljati ovako: **konta-kt** ni ovako: **kontak-t**, nego ovako: **kon-takt**.

Iz toga pravila isključuju se riječi tipa **bicikl** koje na kraju imaju fonem **l** u kombinaciji sa suglasničkim fonemom. U takvim se riječima naime fonem **l** ponaša kao slogotvorno **r**. Pa ako se **otr** (od **otr**ti rastavlja ovako: **o-tr**, i **bicikl** se rastavlja ovako: **bici-kl** (usp. još **ansam-bl**).

13. Sekvencija **i-je** (kao refleks glasa **jata**) ponaša se kao i svaka druga sekvencija tipa **samoglasnički + suglasnički + samoglasnički** fonem. To znači da će se rastavljati ovako: **i-je** (usp.: **odni-je** – prema **odnijeti** i **ri-jeka**).

14. Dijelovi riječi sa stranim pismom prenose se u slijedeći red prema izgovoru, a ne prema pismu. Tako se riječ **Shakespeare** rastavlja ovako: **Shake-speare**. Usp. još: **Bor-deaux** i **Berna-dette**.

15. Riječ na mjestu udvojenih grafema rastavlja se tako da jedan pripadne prethodnom, a drugi slijedećem dijelu riječi: **pod-dioba**.

16. Dijelovi koordiniranih složenica i subordiniranih složenica jedinice kojih su povezane crticom rastavljaju se na mjestu crtice. Prenose se u slijedeći red tako da se crtica kao znak spojnosti prenosi u slijedeći red, a na njezino mjesto stavlja se crtica kao znak prenošenja dijela riječi u slijedeći red.

17. Dijelovi jednadžbi, formula i sličnih jedinica u kojima se nalaze znakovi matematičkih operacija prenose se u slijedeći red bez znaka prenošenja dijela riječi u slijedeći red. Preporučuje se da se prenosi u slijedeći red onaj dio koji stoji s desne strane operacije.

18. U slijedeći red mogu se prenositi i dijelovi višeznamenastih brojaka. Preporučuje se da se iz cjeline u slijedeći red prenose tri ili niz po tri znamenke.

4. Paket **hyph-utf8**

Kao što je već spomenuto, rastavljanje riječi na kraju retka jedno je od istaknutih funkcionalnosti T_EX-a. S obzirom na to da se može prilagoditi mnogim jezicima i sustavima pisanja, ne iznenađuje činjenica da su još u ranim danima razvoja T_EX-a razvijeni mnogi obrasci za rastavljanje za velik broj jezika. Zbog takvog razvoja, danas dostupne datoteke se često koriste starim i nezgrapnim trikovima kako bi ostale upotrebljive s vrlo starim verzijama T_EX-a. Sve takve datoteke koriste 8-bitno kodiranje ili naredbe za na-

glaske (`'e`, `\v{z}`, itd.). Sustav Unicode nije postojao u trenutku kada se pisala većina tih datoteka.

To je uzrokovalo problem kad je sustav XE T_EX uključen u distribuciju T_EXLive u 2007. godini, jer XE T_EX očekuje UTF-8 kodiranje na ulazu. Autor XE T_EX-a, Jonathan Kew, razvio je način korištenja starih obrazaca za rastavljanje i sa XE T_EX-om i sa starijim proširenjima T_EX-a: za svaku datoteku s obrascima `<hyph>.tex` napisao je datoteku `xu-<hyph>.tex` koja detektira je li pokrenuta od strane XE T_EX-a ili nije; u potonjem slučaju, direktno unosi datoteku `<hyph>.tex`, a inače pokreće konvertiranje svih ne-ASCII znakova u UTF-8 kodiranje, te zatim unosi datoteku s obrascima.

Ukratko, u T_EXLive 2007 distribuciji, XE T_EX koristi originalne (stare) obrasce kao osnovu i u hodu ih konvertira u UTF-8 kodiranje.

Što se tiče ConT_EXt sustava, obrasci za rastavljanje su prethodno bili konvertirani u UTF-8, te se ponovno konvertiraju u 8-bitna kodiranja, ovisno o kodiranju fonta.

Želeći ujediniti ta dva pristupa, autori paketa `hyph-utf8`, Mojca Miklavc i Arthur Reutenauer odlučili su preuzeti konverzije svih datoteka s obrascima tada prisutnih u T_EXLive, te omogućiti njihovo uključivanje u izdanje T_EXLive 2008.

Ključna ideja kod arhitekture koju su osmislili jest da se, nakon konvertiranja obrazaca u UTF-8 kodiranje, obrasci umetnu u strukturu pomoću koje je omogućeno njihovo korištenje s obje vrste T_EX sustava – onih s ugrađenom podrškom za UTF-8 (XE T_EX, LuaT_EX), kao i onih koji podržavaju samo 8-bitni ulaz.

Strategija za postizanje tog cilja bila je sljedeća: za svaki jezik `<lang>`, obrasci za rastavljanje pohranjeni su u datoteku naziva `hyph-<lang>.tex`. Te datoteke sadrže samo čiste obrasce, iznimke pri rastavljanju i komentare. Unose se datotekama naziva `loadhyph-<lang>.tex`. U toj se datoteci obavlja prepoznavanje sustava. Primjer koda za slovenski:

```
% Test whether we received one or two arguments
\def\testengine#1#2!{\def\secondarg{#2}}
% We are passed Tau (as in Taco or TEX,
% Tau-Epsilon-Chi), a 2-byte UTF-8 character
\testengine T!\relax
% Unicode-aware engines (such as XeTeX or LuaTeX)
% only see a single (2-byte) argument
\ifx\secondarg\empty
\message{UTF-8 Slovenian Hyphenation Patterns}
\else
\message{EC Slovenian Hyphenation Patterns}
\input conv-utf8-ec.tex
\fi
\input hyph-sl.tex
```

Ključni je korak navesti T_EX da skenira znak u Unicodeu za veliko grčko slovo tau, u UTF-8 kodiranju. Taj znak koristi dva bajta, koji se zbog toga u 8-bitnim T_EX sustavima očitavaju kao dva različita znaka, pa naredba `\testengine` vidi dva argumenta. S druge strane, UTF-8 sustavi vide samo jedan znak (veliko grčko slovo tau), pa je `\secondarg` prazan, tj. nema drugog argumenta.

Ako koristimo UTF-8 T_EX sustav, dovoljno je uključiti datoteku s UTF-8 obrascima. No, ako koristimo 8-bitni sustav, potrebno je konvertirati UTF-8 nizove bajtova u jedinstveni bajt u odgovarajućem kodiranju. Za slovenski jezik, kao i za većinu europskih jezika s latiničnom abecedom, radi se o kodiranju T1. Konverzija se obavlja pomoću

datoteke `conv-utf8-ec.tex`. Slijedi prikaz odvijanja konverzije na primjeru tri znaka specifična za slovenski jezik:

- 'č' (UTF-8 0xc4, 0x8d , T1 0xa3),
- 'š' (UTF-8 0xc5, 0xa1 , T1 0xb2),
- 'ž' (UTF-8 0xc5, 0xbe , T1 0xba).

Da bi se niz `<0xc4, 0x8d>` konvertirao u `0xa3`, potrebno je bajt `0xc4` učiniti aktivnim, te definirati da vrati vrijednost `0xa3` kad mu je argument jednak `0x8d`. Ostali nizovi rade na isti način. Dio sadržaja datoteke `conv-utf8-ec.tex` koji obavlja ovu funkciju jest:

```
\catcode"C4=\active
\catcode"C5=\active
%
\def^^c4#1{%
\ifx#1^^8d^^a3\else % U+010D
\fi}
%
\def^^c5#1{%
\ifx#1^^a1^^b2\else % U+0161
\ifx#1^^be^^ba\else % U+017E
\fi\fi}
% osigurati da svi gornji znakovi imaju validne lccode-ove:
\lccode"A3="A3 % U+010D
\lccode"B2="B2 % U+0161
\lccode"BA="BA % U+017E
```

Posljednji komentar kaže da k tome za znakove koji se pojavljuju u datotekama s obrascima za rastavljanje treba postaviti `lcode` varijablu na ne-nul vrijednosti. Taj se zadatak prije obavljao u samoj datoteci s obrascima.

Podaci za konvertiranje iz UTF-8 u druga kodiranja fontova dobiveni su iz datoteka u kojima se definiraju kodiranja za L^AT_EX i ConT_EXt, te su skupljeni u datotekama naziva `<enc>.dat`. Datoteke konvertiranja automatski se generiraju iz tih datoteka pomoću skripte u Rubyju.

Potrebno je spomenuti nešto o korištenim oznakama jezika. U skladu sa sasvim novom shemom imenovanja datoteka s obrascima i popratnih datoteka, željelo se usvojiti konzistentnu politiku imenovanja za jezike, uz potpuno napuštanje originalnih naziva, budući da su se oni mjestimično pokazali problematičnima. Radi se o *ad hoc* nazivima koje su tijekom mnogo godina birali razni ljudi, bez imalo sistematičnosti. To je dovelo do mnogobrojnih zbunjujućih situacija, npr. naziv `ukhyphen.tex` za britansko-engleske obrasce mogao je biti pogrešno prepoznat kao skraćenica za *ukrajinski*. Bez poznavanja svih naziva datoteka s obrascima, ne bi bilo moguće samo na temelju naziva datoteke odrediti na koji se jezik odnosi.

Prema tome, bilo je jasno da je kod imenovanja datoteka za različite jezike nužno koristiti oznake jezika, a ne država. No ni to nije bilo dovoljno, kao što je vidljivo iz primjera britanskog engleskog, za koji postoje zasebne datoteke, iako kao jezik nije izdvojen od engleskog.

Istraživanjem se pokazalo da je jedini standard koji raspoznaje sve obrasce tzv. IETF *Best Current Practice* preporuka 47 (BCP 47), koji se objavljuje kao RFC dokument. Trenutna je verzija RFC 4646. On dopušta sve varijante jezika koje je trebalo označiti:

- jezici s različitim varijantama diljem raznih zemalja ili područja, poput engleskog.
- jezici pisani različitim pismima, poput srpskog (latinica i ćirilica).

– jezici s više pravopisnih konvencija, poput modernog grčkog (koji je prošao tzv. *monotonu* reformu u 1982.), te njemačkog (za koji je reforma započeta 1996. i dalje u tijeku).

Uz obradu generičkog slučaja, kod izrade paketa obrađeno je mnogo specijalnih slučajeva, koji su zahtijevali posebnu obradu.

Prvi problem na koji se naišlo bili su obrasci kod kojih se pokušalo smjestiti OT1 i T1 kodiranje u istu datoteku. Prvi jezik za koji je to učinjeno je, povijesno, njemački, a ista se shema zatim usvojila i kod francuskog, danskog i latinskog jezika. Radi se o sljedećoj ideji: za svaki od ovih jezika postoje znakovi koji su kodirani na različitim pozicijama u OT1 i T1 kodiranjima. Kod njemačkog, radi se o oštrom s, kod francuskog, radi se o znaku oe, itd. Da bi se to riješilo, originalno je svaki obrazac koji sadrži jedan od tih znakova umnožen u datoteci, s naredbama za selektivno ignoriranje određenog obrasca ovisno o tome koje se kodiranje fonta koristi.

Takvo bi rješenje bilo vrlo nepraktično ponoviti u arhitekturi paketa `hyph-utf8`, ako bi uopće bilo moguće. Svaka bi riječ poput, npr., *coeur* u francuskom trebala davati dva različita niza bajtova u 8-bitnom načinu, po jedan za OT1 i T1 kodiranje (`c1bur` i `cf7ur`, respektivno). Zato je odlučeno staviti umnožene obrasce u zasebnu datoteku naziva `spechyp- \langle lang \rangle -ot1.tex` koji se uključuje samo u *legacy* načinu rada, nakon glavne datoteke `hyph- \langle lang \rangle .tex`.

Tako zapakirani obrasci trebali bi se ponašati jednako kao i naslijeđene datoteke, omogućujući nekoliko točaka rastavljanja s ne-ASCII znakovima u OT1 kodiranju. Dođue, treba naglasiti da OT1 nikako nije dobra opcija za ove jezike. Ovo je ponašanje podržano u paketu isključivo zbog kompatibilnosti, iako autori sumnjaju da je u praksi korisno. Ukoliko se koristi OT1 za njemački i francuski jezik, postojat će nekoliko obrazaca s oštrim s i 'oe', no mnogi drugi obrasci s naglašenim znakovima će nedostajati. Da bi se u potpunosti iskoristili obrasci za rastavljanje, potrebno je koristiti T1 fontove.

Treba spomenuti da naposljetku u slučaju njemačkog jezika nije korišten spomenuti pristup, jer se željelo uzeti u obzir da se trenutno radi na popravljaju njemačkih obrazaca, te je odlučeno da se novi obrasci koriste s UTF-8 sustavima, ali ne i s 8-bitnim sustavima, kako bi se očuvala kompatibilnost. Kod 8-bitnih sustava, naprosto se direktno uključuje originalna datoteka s obrascima u T1, bez obavljanja ikakve konverzije. Za tri druga jezika, francuski, danski i latinski, korištene su datoteke `spechyp- \langle lang \rangle -ot1.tex`.

Još je jedan zanimljiv problem, s drukčijim komplikacijama, iskrsnuo kod ukrajinskog i ruskog jezika. Tu je za svaki jezik bilo prisutno više skupina obrazaca. Nove datoteke s obrascima, donesene u ovom paketu, također su osmišljene za više kodiranja, ali u drukčijem smislu. U ovom slučaju, kodiranje se uključuje postavljanjem kontrolnog niza `\Encoding` prije uključanja datoteke s obrascima. Ovisno o vrijednosti te naredbe se potom uključuje odgovarajuća datoteka za konverziju, koja radi na jednak način kao i naše datoteke `conv-utf8- \langle enc \rangle .tex`. Postoji i predložena vrijednost za `\Encoding`, a to je T2A za ruski i ukrajinski jezik. T2A je najraširenije kodiranje fonta za te jezike, te se koristi u datoteci s obrascima, pa nije potrebna konverzija ako se ne mijenja predložena vrijednost `\Encoding`.

Zatim, i ruski i ukrajinski jezik imali su nekoliko datoteka s obrascima, s različitim autorima i/ili pravilima za rastavljanje (fonetski, etimološki itd.). Te se datoteke odabiru kontrolnim nizom `\Pattern`, koja je početno postavljena na `as` za ruski (autora Aleksandra Lebedeva), te na `mp` za ukrajinski (autora Maksyma Polyakova). Oba se izbora mogu nadjačati samo kod stvaranja formata, jer se u tom trenutku obrasci *smrzavaju*.

Naposljedku, za omogućavanje rastavljanja unutar riječi koje sadržavaju crticu (znak za rastavljanje), u datoteci `hyph2.tex` bio je iskorišten poseban trik, slično kao kod datoteke `hyph1.tex` Bernda Raichlea za T1 fontove.

Ta tri svojstva morala su se obraditi u našoj arhitekturi na vrlo različite načine. Prvo svojstvo je beznačajno u UTF-8 načinu rada, no zahtijevalo bi fundamentalne promjene u `loadhyph-<lang>.tex` datotekama za 8-bitne sustave, s obzirom da više ne bi bila zadovoljena implicitna pretpostavka da svaki jezik koristi točno jedno 8-bitno kodiranje. Drugo svojstvo se lakše obradilo, no ipak je zahtijevalo dodatke u `loadhyph-<lang>.tex` datotekama. Na kraju, treće svojstvo, iako vrlo zanimljivo, činilo se problematičnijim nego što bi bilo dopustivo u primjeni.

Naposljedku, odlučeno je da se ta svojstva ne uključuju među UTF-8 obrasce prije izlaska T_EXLive 2008, ali su omogućeni u *legacy* načinu rada, kako bi se osigurala kompatibilnost sa starijim dokumentima i sustavima. U suradnji s Vladimirom Volovichem, koji je osmislio način organizacije ruskih obrazaca, te je inspirirao ukrajinski analogon, u paket je uključen popis rastavljenih složenica u datotekama naziva `exhyph-ru.tex` i `exhyph-uk.tex`, respektivno. Pritom korištena strategija jest:

- U UTF-8 načinu rada, uključiti UTF-8 obrasce, a zatim `ex-` datoteku.
- U *legacy* načinu rada, direktno uključiti originalne datoteke s obrascima.

Sveukupno, jedino svojstvo koje nedostaje u T_EXLive 2008 jest mogućnost odabira željenih obrazaca u UTF-8 načinu rada. Za oba jezika, samo je predložena/defaultna skupina obrazaca konvertirana u UTF-8. Postavka `\Pattern`, dakle, u ovom slučaju neće imati nikakvog efekta, ali će se u 8-bitnom načinu rada ponašati kao i prije. U planu je promjena tog ponašanja i omogućavanje cijelog spektra funkcionalnosti koje su imale originalne datoteke s obrascima.

Također je potrebno primijetiti da se u T_EXLive 2007 za bugarski jezik koristio isti mehanizam učitavanja obrazaca, no da je ustvari bilo moguće samo jedno kodiranje i samo jedna datoteka s obrascima, pa ustvari nije postojala mogućnost izbora, te je bilo jednostavno prilagoditi bugarske obrasce novoj arhitekturi.

Paket `hyph-utf8` objavljen je na CTAN, te čini osnovu za podršku rastavljanju riječi u T_EXLive 2008. Pripadne T_EX datoteke nalaze se u `tex/generic/hyph-utf8` i pripadajućim poddirektorijima, a podaci o kodiranju i skripte u Rubyju dostupni su u `source/generic/hyph-utf8`. Prilagodba ovog paketa ostvarena za format `hrtex` detaljnije je opisana u poglavlju 6, a svodi se na nadjačavanje datoteke s UTF-8 obrascima za hrvatski jezik, `hyph-hr.tex`, lokalnom verzijom istog imena, koja sadrži obrasce za rastavljanje kreirane novim pristupom.

Prilog: datoteka `hyph-hr.tex`

```

\patterns{ %
% V1V
% izbjegavamo suviše
a1i a1o a1u %a1a %a1e
e1a e1i e1o e1u %e1e
i1a i1e i1o i1u %i1i
o1e o1i o1o o1u %o1a
u1i u1o u1u %u1a %u1e
%
% K1K
% udvojeni suglasnici
d1d j1j
% V1KV
a1ba a1be a1bi a1bo a1bu
a1ca a1ce a1ci a1co a1cu
a1ča a1če a1či a1čo a1ču
a1ća a1će a1ći a1čo a1ču
a1da a1de a1di a1do a1du
a1dža a1dže a1dži a1džo a1džu
a1đa a1đe a1đi a1đo a1đu
a1fa a1fe a1fi a1fo a1fu
a1ga a1ge a1gi a1go a1gu
a1ha a1he a1hi a1ho a1hu

```

a1ja a1je a1ji a1jo a1ju
 a1ka a1ke a1ki a1ko a1ku
 a1la a1le a1li a1lo a1lu
 a1lja a1lje a1lji a1ljo a1ljju
 a1ma a1me a1mi a1mo a1mu
 a1na a1ne a1ni a1no a1nu
 a1nja a1nje a1nji a1njo a1nju
 a1pa a1pe a1pi a1po a1pu
 a1ra a1re a1ri a1ro a1ru
 a1sa a1se a1si a1so a1su
 a1ša a1še a1ši a1šo a1šu
 a1ta a1te a1ti a1to a1tu
 a1va a1ve a1vi a1vo a1vu
 a1za a1ze a1zi a1zo a1zu
 a1ža a1že a1ži a1žo a1žu
 e1ba e1be e1bi e1bo e1bu
 e1ca e1ce e1ci e1co e1cu
 e1ča e1če e1či e1čo e1ču
 e1ća e1će e1ći e1čo e1ču
 e1da e1de e1di e1do e1đu
 e1dža e1dže e1dži e1džo e1džu
 e1đa e1đe e1đi e1đo e1đu
 e1fa e1fe e1fi e1fo e1fu
 e1ga e1ge e1gi e1go e1gu
 e1ha e1he e1hi e1ho e1hu
 e1ja e1je e1ji e1jo e1ju
 e1ka e1ke e1ki e1ko e1ku
 e1la e1le e1li e1lo e1lu
 e1lja e1lje e1lji e1ljo e1ljju
 e1ma e1me e1mi e1mo e1mu
 e1na e1ne e1ni e1no e1nu
 e1nja e1nje e1nji e1njo e1nju
 e1pa e1pe e1pi e1po e1pu
 e1ra e1re e1ri e1ro e1ru
 e1sa e1se e1si e1so e1su
 e1ša e1še e1ši e1šo e1šu
 e1ta e1te e1ti e1to e1tu
 e1va e1ve e1vi e1vo e1vu
 e1za e1ze e1zi e1zo e1zu
 e1ža e1že e1ži e1žo e1žu
 i1ba i1be i1bi i1bo i1bu
 i1ca i1ce i1ci i1co i1cu
 i1ča i1če i1či i1čo i1ču
 i1ća i1će i1ći i1čo i1ču
 i1da i1de i1di i1do i1đu
 i1dža i1dže i1dži i1džo i1džu
 i1đa i1đe i1đi i1đo i1đu
 i1fa i1fe i1fi i1fo i1fu
 i1ga i1ge i1gi i1go i1gu
 i1ha i1he i1hi i1ho i1hu
 i1ja i1je i1ji i1jo i1ju
 i1ka i1ke i1ki i1ko i1ku
 i1la i1le i1li i1lo i1lu
 i1lja i1lje i1lji i1ljo i1ljju
 i1ma i1me i1mi i1mo i1mu
 i1na i1ne i1ni i1no i1nu
 i1nja i1nje i1nji i1njo i1nju
 i1pa i1pe i1pi i1po i1pu
 i1ra i1re i1ri i1ro i1ru
 i1sa i1se i1si i1so i1su
 i1ša i1še i1ši i1šo i1šu
 i1ta i1te i1ti i1to i1tu
 i1va i1ve i1vi i1vo i1vu
 i1za i1ze i1zi i1zo i1zu
 i1ža i1že i1ži i1žo i1žu
 o1ba o1be o1bi o1bo o1bu
 o1ca o1ce o1ci o1co o1cu
 o1ča o1če o1či o1čo o1ču

o1ća o1će o1ći o1čo o1ču
 o1da o1de o1di o1do o1đu
 o1dža o1dže o1dži o1džo o1džu
 o1đa o1đe o1đi o1đo o1đu
 o1fa o1fe o1fi o1fo o1fu
 o1ga o1ge o1gi o1go o1gu
 o1ha o1he o1hi o1ho o1hu
 o1ja o1je o1ji o1jo o1ju
 o1ka o1ke o1ki o1ko o1ku
 o1la o1le o1li o1lo o1lu
 o1lja o1lje o1lji o1ljo o1ljju
 o1ma o1me o1mi o1mo o1mu
 o1na o1ne o1ni o1no o1nu
 o1nja o1nje o1nji o1njo o1nju
 o1pa o1pe o1pi o1po o1pu
 o1ra o1re o1ri o1ro o1ru
 o1sa o1se o1si o1so o1su
 o1ša o1še o1ši o1šo o1šu
 o1ta o1te o1ti o1to o1tu
 o1va o1ve o1vi o1vo o1vu
 o1za o1ze o1zi o1zo o1zu
 o1ža o1že o1ži o1žo o1žu
 u1ba u1be u1bi u1bo u1bu
 u1ca u1ce u1ci u1co u1cu
 u1ča u1če u1či u1čo u1ču
 u1ća u1će u1ći u1čo u1ču
 u1da u1de u1di u1do u1đu
 u1dža u1dže u1dži u1džo u1džu
 u1đa u1đe u1đi u1đo u1đu
 u1fa u1fe u1fi u1fo u1fu
 u1ga u1ge u1gi u1go u1gu
 u1ha u1he u1hi u1ho u1hu
 u1ja u1je u1ji u1jo u1ju
 u1ka u1ke u1ki u1ko u1ku
 u1la u1le u1li u1lo u1lu
 u1lja u1lje u1lji u1ljo u1ljju
 u1ma u1me u1mi u1mo u1mu
 u1na u1ne u1ni u1no u1nu
 u1nja u1nje u1nji u1njo u1nju
 u1pa u1pe u1pi u1po u1pu
 u1ra u1re u1ri u1ro u1ru
 u1sa u1se u1si u1so u1su
 u1ša u1še u1ši u1šo u1šu
 u1ta u1te u1ti u1to u1tu
 u1va u1ve u1vi u1vo u1vu
 u1za u1ze u1zi u1zo u1zu
 u1ža u1že u1ži u1žo u1žu
 %
 % 1KKV
 % samo kombinacije iz rječnika
 1bja 1bje 1bji 1bjo 1bju
 1bla 1ble 1bli 1blo 1blu
 1blja 1blje 1blji 1bljo 1blju
 1bra 1bre 1bri 1bro 1bru
 1cja 1cje 1cji 1cjo 1cju
 1cma 1cme 1cmi 1cmo 1cmu
 1cra 1cre 1cri 1cro 1cru
 1cva 1cve 1cvi 1cvo 1cvu
 1čla 1čle 1čli 1člo 1člu
 1čva 1čve 1čvi 1čvo 1čvu
 1dja 1dje 1dji 1djo 1dju
 1dlja 1dlje 1dlji 1dljo 1dlju
 1dla 1dle 1dli 1dlo 1dlu
 1dna 1dne 1dni 1dno 1dnu
 1dra 1dre 1dri 1dro 1dru
 1dva 1dve 1dvi 1dvo 1dvu
 1fja 1fje 1fji 1fjo 1fju
 1fla 1fle 1fli 1flo 1flu

```

1fra 1fre 1fri 1fro 1fru
1gla 1gle 1gli 1glo 1glu
1glja 1glje 1glji 1gljo 1glju
1gma 1gme 1gmi 1gmo 1gmu
1gna 1gne 1gni 1gno 1gnu
1gnja 1gnje 1gnji 1gnjo 1gnju
1gra 1gre 1gri 1gro 1gru
1gva 1gve 1gvi 1gvo 1gvu
1hla 1hle 1hli 1hlo 1hlu
1hlja 1hlje 1hlji 1hljo 1hlju
1hma 1hme 1hmi 1hmo 1hmu
1hra 1hre 1hri 1hro 1hru
1hva 1hve 1hvi 1hvo 1hvu
1kća 1kće 1kći 1kčo 1kću
1kla 1kle 1kli 1klo 1klu
1klja 1klje 1klji 1kljo 1klju
1kma 1kme 1kmi 1kmo 1kmu
1kna 1kne 1kni 1kno 1knu
1knja 1knje 1knji 1knjo 1knju
1kra 1kre 1kri 1kro 1kru
1kva 1kve 1kvi 1kvo 1kvu
1mja 1mje 1mji 1mjo 1mju
1mla 1mle 1mli 1mlo 1mlu
1mlja 1mlje 1mlji 1mljo 1mlju
1mna 1mne 1mni 1mno 1mnu
1mra 1mre 1mri 1mro 1mru
1pča 1pče 1pči 1pčo 1pću
1pja 1pje 1pji 1pjo 1pju
1pla 1ple 1pli 1plo 1plu
1plja 1plje 1plji 1pljo 1plju
1pra 1pre 1pri 1pro 1pru
1psa 1pse 1psi 1pso 1psu
1pša 1pše 1pši 1pšo 1pšu
1pta 1pte 1pti 1pto 1ptu
1rja 1rje 1rji 1rjo 1rju
1sha 1she 1shi 1sho 1shu
1sja 1sje 1sji 1sjo 1sju
1ska 1ske 1ski 1sko 1sku
1sla 1sle 1sli 1slo 1slu
1slja 1slje 1slji 1sljo 1slju
1sma 1sme 1smi 1smo 1smu
1sna 1sne 1sni 1sno 1snu
1snja 1snje 1snji 1snjo 1snju
1spa 1spe 1spi 1spo 1spu
1sra 1sre 1sri 1sro 1sru
1sta 1ste 1sti 1sto 1stu
1sva 1sve 1svi 1svo 1svu
1šča 1šče 1šči 1ščo 1šču
1šća 1šće 1šći 1ščo 1šču
1ška 1ške 1ški 1ško 1šku
1šlja 1šlje 1šlji 1šljo 1šlju
1šma 1šme 1šmi 1šmo 1šmu
1špa 1špe 1špi 1špo 1špu
1šta 1šte 1šti 1što 1štu
1šva 1šve 1švi 1švo 1švu
1tja 1tje 1tji 1tjo 1tju
1tka 1tke 1tki 1tko 1tku

1tla 1tle 1tli 1tlo 1tlu
1tma 1tme 1tmi 1tmo 1tmu
1tra 1tre 1tri 1tro 1tru
1tva 1tve 1tvi 1tvo 1tvu
1vja 1vje 1vji 1vjo 1vju
1vla 1vle 1vli 1vlo 1vlu
1vra 1vre 1vri 1vro 1vru
1zba 1zbe 1zbi 1zbo 1zbu
1zga 1zge 1zgi 1zgo 1zgu
1zja 1zje 1zji 1zjo 1zju
1zla 1zle 1zli 1zlo 1zlu
1zma 1zme 1zmi 1zmo 1zmu
1zna 1zne 1zni 1zno 1znu
1zra 1zre 1zri 1zro 1zru
1zva 1zve 1zvi 1zvo 1zvu
1žda 1žde 1ždi 1ždo 1ždu
1žla 1žle 1žli 1žlo 1žlu
1žlja 1žlje 1žlji 1žljo 1žlju
1žma 1žme 1žmi 1žmo 1žmu
1žnja 1žnje 1žnji 1žnjo 1žnju
1žra 1žre 1žri 1žro 1žru
1žva 1žve 1žvi 1žvo 1žvu
%
% 1K2KK
% SAMO NEKE, gdje je važno 2
%
1s2hv 1s2kv 1s2tv
%
% Kr1K
% slogotvorni r
br1b br1đ br1l br1s br1v br2v1n br1z
cr1k cr2k1n cr1n cr2n1k cr1p cr2p1s
cr1t cr1v cr2v1lj
dr1h dr2h1t dr1m dr1v dr2v1n
fr1k fr2k1t
gr1b gr1č gr2č1k gr1k gr1l gr1m
gr1n gr2n1č
hr1k hr2k1n hr1l hr1p hr1v
kr1č kr2č1m kr1l kr1p kr2p1n kr1s
kr1v kr1ž kr2ž1lj
mr1k mr1lj mr1m mr1š mr2š1t mr1t
mr1v mr1z mr2z1n mrž1nj
pr1ć pr1lj prs1n prs1k prš1t pr1ša
pr1ši pr1šu pr1t pr1v pr1ž
sr1c sr1č sr1d srdž1b srk1n sr1n sr1p
tr1č tr1g tr1k tr2k1n tr1lj tr1n tr1p
tr1s tr2s1k tr2s1t tr1t tr1ž
vr1b vr1č vr1g vr1ho vr1hu vr1l vr1p
vr2p1c vr1s vr1š vr1t vrv1j
zr1n
žr1tv
}

\endinput

```

V. Paket encT_EX

1. Osnovne informacije

Paket encT_EX je proširenje T_EX-a autora prof. Petra Olšáka, češkog matematičara i računalnog znanstvenika. Paket je kompatibilan s originalnim T_EX-om. U odnosu na originalni program T_EX u encT_EX-u dodano je deset novih naredbi. Tim se naredbama mogu definirati i čitati tablice konverzija koje koristi T_EX-ov ulazni procesor i koje se koriste tijekom ispisa na terminal, log datoteku i `\write` datoteke. Spomenute tablice konverzija spremaju se u datoteku za format, te se pri čitanju datoteke za format reinicijaliziraju na isto stanje kao u trenutku izvršenja naredbe `\dump`.

Prije nastanka encT_EX-a korištenje znakova iz ISO 8859-2 tablice (ranije spomenuto *Latin-2* kodiranje) s T_EX-om bilo je omogućeno datotekom `latin2.sty`, autora Andreja Brodnika. U datoteci utemeljenoj na verziji za *Latin-1*, autora Olafa Barthela, uvedene su definicije za svaki od standardnih znakova iz tablice *Latin-2*. Pritom su definicije pojedinih znakova preuzete iz prethodno objavljenih datoteka autora Nenada Antonića (za hrvatski znak Đ) te Michala Jaegermanna (za znak *ogonek*). Funkcionalnost znaka *ogonek* preuzeta je iz radova autora Leszeka Holenderskog, Jerzyja Rylla i Janusza Bieńa.

Prva verzija encT_EX-a objavljena je 1997. godine, a omogućavala je konverziju bajt-za-bajt samo pomoću `xord` i `xchr` vektora, te označavanje znakova kao *ispisivih* (naredbe `\xordcode`, `\xchrcode` i `\xprncode`).

TCX tablice obnovljene su u 1998. godini. Ove obnovljene tablice izvršavale su istu funkciju kao encT_EX 1997. godine, pa se smatralo da više nema potrebe za daljnjim održavanjem encT_EX-a. Međutim, nakon uočavanja problema s nedostatkom podrške za ulazne datoteke kodirane u UTF-8, dodana je UTF-8 podrška za encT_EX. Tako obnovljena verzija encT_EX-a objavljena je u prosincu 2002. godine.

Verzija iz veljače 2003. godine uvodi sedam novih temeljnih naredbi: `\mubyte`, `\endmubyte`, `\mubytein`, `\mubyteout`, `\mubytelog`, `\specialout`, `\noconvert`. Uvedene naredbe omogućuju definiranje postavki za konverziju datoteka kodiranih u UTF-8.

U lipnju 2004. godine izašla je nova verzija encT_EX-a u kojoj su ispravljene neke pogreške, ali ne uvodi nove funkcionalnosti. Nova verzija je kompatibilna sa starom verzijom iz 1997. godine. Prema autoru, u encT_EX-u više neće dolaziti do većih promjena, te je zajamčena kompatibilnost eventualnih novih verzija s trenutnom verzijom, što ovaj paket čini pouzdanim za korištenje na duže vrijeme.

Ako su na računalu prisutni svi potrebni resursi, encT_EX se može aktivirati uključivanjem opcije `-enc` pri generiranju formata. Opcija `-enc` mora se koristiti u fazi generiranja formata jer encT_EX sprema svoje naredbe i podatke na datoteku s formatom. Kad se koristi tako dobiveni format, encT_EX se automatski inicijalizira iz datoteke s formatom i nije potrebno ponovno koristiti opciju `-enc`. Ako se koristi format u kojem encT_EX nije aktiviran te se isti pokušava upotrijebiti s `-enc` opcijom, ispiše se upozorenje i opcija se ignorira.

TCX tablice rade s istim vektorima kao i encT_EX u web2c distribuciji (vektori `xord` i `xchr`), što može dovesti do manjih konflikata. Ako se encT_EX koristi zajedno s TCX tablicom, TCX tablica može prebrisati početne vrijednosti `\xordcode`, `\xchrcode` i `\xprncode`. Te će početne vrijednosti biti dokumentirane kasnije u ovom poglavlju. Također, ako se uvedene vrijednosti pohrane u format pomoću encT_EX-a, te se TCX tablica koristi zajedno s takvim formatom, tada vrijednosti iz formata mogu biti prebrisane vrijednostima iz TCX tablice. S druge strane, naredbe `\xordcode`, `\xchrcode` i `\xprncode` mogu se bez problema koristiti za čitanje i spremanje vrijednosti nakon inicijalizacije TCX tablice.

Autor TeX-a D. Knuth pretpostavljao je da se parametri filtriranja postavki koje ovise o sustavu postavljaju na nivou izvornog koda. EncTeX pomiče zadavanje tih postavki iz nivoa izvornog koda na nivo izvršavanja. To je u skladu s činjenicom da je memorijske parametre TeX-a moguće zadati za vrijeme izvršavanja u modernim TeX distribucijama. Ovisno o korištenom sustavu, mogu se zadati specifične tablice konverzija. Zatim se može zadati `\let\xordcode=\undefined`, itd. (analogno za ostale temeljne naredbe uvedene u encTeX-u), te se može izvršiti `\dump`. Format tada ima tablice konverzija pohranjene kao sistemske specifikacije i korisnik ne može izvršavati daljnje promjene. Upotreba ovog formata jednaka je upotrebi originalnog TeX-a.

Što se tiče licenciranja programa, encTeX dodaje nove temeljne naredbe u TeX, pa se rezultirajući program ne može zvati tim imenom. Poruke na terminalu i rezultirajuća log datoteka dobivene pri pokretanju encTeX-a jasno naznačuju da je program modificirana verzija TeX-a. S obzirom na rast popularnosti UTF-8 kodiranja, korištenje encTeX-a i eventualni drugi načini modificiranja ulaznog procesora TeX-a nezaobilazni su ukoliko se želi spriječiti izumiranje 8-bitnog TeX sustava. Važno je reći da encTeX ima isto standardno ponašanje kao originalni TeX ako se nove naredbe nikad ne koriste.

2. Konverzija bajt-za-bajt

Svi tekstualni ulazi u TeX preslikavaju se xord vektorom u ulaznom procesoru (*oči* u TeXbook terminologiji). Ako znak ima kod x u korištenom sustavu, isti znak ima kod $y = \text{xord}[x]$ u TeX-u.

Svi tekstualni izlazi TeX-a na terminal, log datoteku i datoteke korištene s `\write` naredbom filtriraju se kroz `xchr` vektor i pomoću *ispisivosti* znaka. Ako znak s kodom y nije ispisiv, tada je izlaz u `^^code` notaciji (kao što je dokumentirano u TeXbook-u, str. 45). Ako je znak s kodom y ispisiv, tada je izlazni kôd tog znaka na terminalu i u tekstualnim datotekama $z = \text{xchr}[y]$.

Za pristup vektorima, proširenje encTeX uvodi tri nove temeljne naredbe sa sintaksom jednakom naredbi `\lccode`:

– `\xordcode i ... xord[i]`

– `\xchrcode i ... xchr[i]`

– znak s kodom i je ispisiv (na terminalu i log datoteci ne koristi se notacija `^^code`) ako i samo ako `\xprncode i > 0` ili je $i \in \{32, \dots, 126\}$.

Vrijednosti `\xordcode`, `\xchrcode` i `\xprncode` moguće je postaviti na cijele brojeve između 0 i 255. Tako postavljene vrijednosti su globalne. To znači da su postavke zadane unutar grupe imaju globalni efekt i nebitno je stavi li se prefiks `\global` ili ne.

Početne vrijednosti navedenih vektora u iniTeX stanju su:

– `\xordcode i = i`, za $i \in \{128, \dots, 255\}$,

– `\xchrcode i = i`, za $i \in \{128, \dots, 255\}$,

– `\xprncode i = 0`, za $i \in \{0, \dots, 31, 127, \dots, 255\}$,

– `\xprncode i = 1`, za $i \in \{32, \dots, 126\}$.

Vrijednosti `\xordcode i` i `\xchrcode i`, za $i \in \{0, \dots, 127\}$ ovise o sustavu, no na sustavima s ASCII kodiranjem vrijedi: `\xordcode i = i`, `\xchrcode i = i`.

3. Višebajtna konverzija

Od verzije iz prosinca 2002. encTeX ima opciju konvertiranja više bajtova u jedan bajt ili upravljačku riječ na razini procesiranja ulaza. Taj se *jedan bajt* ponovno pretva-

ra u originalnih *više bajtova* kad se procesira naredba `\write` ili kad T_EX ispisuje izlaz na terminal ili u log datoteku. Glavna namjena ovog paketa je korištenje za rad s ulaznim datotekama kodiranim u UTF-8. U tu svrhu bilo je potrebno dodijeliti vrijednosti `\catcodes`, `\uccodes` itd. slovima iz češke abecede, no neka su slova kodirana s dva bajta u UTF-8. EncT_EX omogućuje preslikavanje drugih kodova iz UTF-8 u upravljačke riječi, pa je broj UTF-8 kodova iz ulaznih datoteka neograničen.

Uvedeno je pet novih naredbi za upravljanje konverzijom:

- `\mubytein`,
- `\mubyteout`,
- `\mubytelog`,
- `\mubyte`,
- `\endmubyte`.

`\mubytein`, `\mubyteout` i `\mubytelog` su cjelobrojni registri sa standardnom vrijednosti nula: to znači da se ne izvršava nikakva konverzija čak i ako je tablica konverzija (stvorena pomoću `\mubyte`, `\endmubyte`) neprazna. Ako je vrijednost `\mubytein` pozitivna, izvršava se konverzija na razini ulaznog procesiranja pomoću tablice konverzija. Ako je vrijednost `\mubyteout` pozitivna, tada se aktivira konverzija za izlaz na `\write` datoteke pomoću iste tablice konverzija. Ako je vrijednost `\mubytelog` pozitivna, aktivira se konverzija za izlaz na terminal i u log datoteku.

Tablica konverzija je inicijalno prazna. Novi se redak u tablicu unosi parom naredbi `\mubyte`, `\endmubyte`, kako slijedi:

`\mubyte<prvi simbol><opcionalni razmak><opcionalni prefiks><niz bajtova>\endmubyte`.

Ovdje koristimo riječ *simbol* u značenju *token*, što je najmanja zasebna cjelina nastala leksičkom analizom programskog koda (u ovom slučaju: *razina procesiranja ulaza*). Niz tako dobivenih cjelina zatim prolazi sintaktičku analizu (u ovom slučaju: *razina procesiranja simbola*). Svaki *<niz bajtova>* konvertirat će se u *<prvi simbol>* prilikom procesiranja ulaza. Dvije su mogućnosti za *<prvi simbol>*: to može biti znak ili upravljačka riječ. Ako je *<prvi simbol>* znak, tada se njegova `\catcode` vrijednost ignorira i *<prvi simbol>* se interpretira kao *<bajt>*. Taj se *<bajt>* konvertira ponovno u *<niz bajtova>* u `\write` datotekama, log datoteci i terminalu.

Ako je *<prvi simbol>* upravljačka riječ, tada će se *<niz bajtova>* konvertirati u istu upravljačku riječ oblika jednog simbola na razini procesiranja ulaza. To znači da procesor simbola nikada ne mijenja tu upravljačku riječ. Procesor simbola ostaje u srednjem stanju nakon što je pročitana ta upravljačka riječ. Ako je `\mubyteout < 2`, tada se izlaz na `\write` datoteke ne konvertira natrag u *<niz bajtova>*, a upravljačka riječ se ekspandira kao i obično. Ako je `\mubyteout ≥ 2`, tada se upravljačka riječ deklarirana s `\mubyte` konvertira natrag u *<niz bajtova>* u parametrima naredbe `\write`. To funkcionira samo ako upravljačka riječ nije ekspandirana. Dakle, upravljačka riječ mora biti neekspandirajuća ili mora biti označena s `\noexpand`. Ako je `\mubyteout ≥ 3`, encT_EX automatski sprečava ekspanziju upravljačkih riječi deklariranih `\mubyte` naredbom.

Upravljačke riječi se ne konvertiraju ponovno u *<niz bajtova>* u log datotekama i u izlazu na terminal. Sintaksa i značenje stavke *<opcionalni prefiks>* bit će objašnjeni kasnije.

Upravljanje tablicom konverzije

Podaci pohranjeni u tablici s konverzijama su globalne vrijednosti, dok su vrijednosti registara `\mubytein`, `\mubyteout` i `\mubytelog` lokalne.

Naredbe `\mubyte`, `\endmubyte` rade vrlo slično dobro poznatom paru naredbi `\csname`, `\endcsname`. Razlika je u tome što se *<prvi simbol>* ne proširuje i što nakon simbola može slijediti *<jedan opcionalni razmak>*. Ako prilikom čitanja *<niz bajtova>* procesor naide na neku upravljačku riječ različitu od `\endmubyte`, ispiše se poruka o pogreški, kao u sljedećem primjeru:

```
! Missing \endmubyte inserted
\begtt
```

(Ubačen je nedostajući `\endmubyte`.)

Naredba `\mubyte` se ne izvršava na razini ekspanzije; radi se o naredbi za dodjeljivanje vrijednosti. Ako napišemo

```
\begtt
\edef\a{\mubyte X ABC \endmubyte}
```

tada naredba `\a` sadrži simbole `\mubyte X ABC\endmubyte`.

Primjeri:

```
\mubyte ^^c1
^^c3^^81 \endmubyte % \A
\mubyte ^^e1
^^c3^^a1 \endmubyte % \a
% itd.: implementacija UTF-8
```

```
\mubyte \endash ^^c4^^f6 \endmubyte %preslikavanje u upravljačku riječ
\mubyte \integral INT \endmubyte %primjer za ilustraciju
\mubytein=1 \mubyteout=1 %ovdje se aktiviraju konverzije
```

```
\def\endash {--} %ova definicija je dobra i za \write datoteke
\def\integral {\ifmmode \int \else \int \fi}
```

U ovom primjeru uneseno je više razmaka (ili tabulator) na mjesto za *<opcionalni razmak>* jer ti znakovi imaju `\catcode` vrijednost jednog razmaka te ih procesor simbola konvertira u jedan *<opcionalni razmak>*.

Ako se koristi kôd iz primjera, riječ "INTEGRAL" konvertira se u simbol `\integral` kojeg slijede slova "EGRAL". Tekst "INT nešto" konvertira se u simbol `\integral` kojeg slijede razmak i riječ "nešto". Mogu se napisati sljedeće konstrukcije: `\def INT{nešto}`, `\let INT=\foo`, itd.

Nakon `\show INT` dobivamo:

```
> \integral=macro:
-> \ifmmode \int \else \int \fi .
1.18 \show INT
```

te se `\string INT` ekspandira u: `\integral`.

Pretpostavimo da imamo deklaraciju `INT` kao u prethodnom primjeru i pretpostavimo da smo napisali `\INT`. Tada bi izlaz procesora simbola trebala biti prazna upravljačka riječ (`\csname\endcsname`) iza koje slijedi upravljačka riječ `\integral`. No u encTeX-u postoji iznimka kako bi se spriječilo miješanje s praznom upravljačkom riječi. `\INT` daje samo naredbu `\integral`; u ovom slučaju ignorira se *backslash*. Procesor simbola ostaje u srednjem stanju nakon što je pročitano `\INT`, te znak za slovo može uslijediti neposredno nakon toga.

Svojstva procesa konverzije

Ulaz se konvertira neposredno nakon postavljanja `\mubytein` na pozitivnu vrijednost, pa konverzija može započeti već u retku u kojem je zadana ta vrijednost.

`<niz bajtova>` konvertira se samo ako je cijeli `<niz bajtova>` uključen u jednoj liniji. Na kraju `<niz bajtova>` može se nalaziti znak `\endlinechar`.

Niz `^^c3^81` se ne konvertira u slovo `Á` čak ni ako se koristi kôd iz primjera. Razlog tome je što se konverzija `^^` vrši u procesoru simbola nakon `\mubyte` konverzije.

Konverzija `\xordcode` se izvršava prije `\mubyte` konverzije na strani ulaza, a konverzija `\xchrcode` se izvršava nakon konverzije `\mubyte` tijekom ispisa u datoteke ili na terminal. Na sljedećem dijagramu prikazan je slijed konverzija:

– ulazni tekst `-> \xordcode -> nadodan \endlinechar -> \mubyte ->`

`-> procesor simbola -> procesor ekspanzije`

`- \write argument -> procesor ekspanzije -> \mubyte -> \xchrcode -> izlaz`

`<niz bajtova>` se ne konvertira u oblik `^^` tijekom ispisa u datoteku čak ni ako `\xprncode` bajtova iz `<niz bajtova>` ima vrijednost nula. `<niz bajtova>` se više ne konvertira čak ni ako u njemu postoji znak koji se normalno konvertira po nekom pravilu u tablici s konverzijama.

Uzmimo da postoje dva ili više `<niz bajtova>` u tablici s konverzijama koji su jednaki ili počinju istim znakovima, pri čemu je jedan niz podniz drugog niza. Tada se konverzija u procesoru ulaza vrši na maksimalnom mogućem `<niz bajtova>`. To je svojstvo implementirano u verziji iz veljače 2003. Primjer:

```
\mubyte X A \endmubyte
\mubyte Y ABC \endmubyte
\mubyte \foo ABCD \endmubyte
```

Slovo A normalno se konvertira u X u ovom primjeru, ali ako neposredno nakon njega slijede slova BC, tada se ABC konvertira u Y, s iznimkom ABCD, koja se konvertira u `\foo`. Poredak postavki `\mubyte` u ovom primjeru nema nikakvog značaja.

Ako se više identičnih `<niz bajtova>` koristi u `\mubyte` zapisima, tada posljednji navedeni redak ima prednost, te se prethodni retci brišu.

Konverzija pri ispisu u log datoteku i na terminal

Izlaz na terminal i log datoteke ne konvertira se ako je vrijednost `\mubytelog` jednaka nuli. Ako je vrijednost `\xprncode` nekog znaka jednaka nuli, tada se znak ispisa u `^^A` ili `^^bc` obliku. Ako je vrijednost `\mubytelog` pozitivna, znakovi pohranjeni u tablici s konverzijama se konvertiraju u `<niz bajtova>` te se bajtovi iz tih `<niz bajtova>` ne konvertiraju u `^^` oblik. S druge strane, upravljačke riječi ostaju nepromijenjene u log datoteci i terminalu čak i ako je vrijednost `\mubytelog` pozitivna.

Konverzija za izlaz na terminal i na log datoteke se uključuje, odnosno isključuje postavljanjem vrijednosti `\mubytelog`. Ti se izlazi ne mogu odvojiti. Praktički, to znači da se ne može postići da je konverzija pri izlazu u log datoteku omogućena, a konverzija pri izlazu u terminal onemogućena.

Postoji poseban dio izlaza na terminal i log: ponovni ispis čitave linije s ulaza (npr., kad se prikazuje kontekst neke pogreške). Tu situaciju nazvat ćemo *ponovno ispisivanje linije* (*line-reprinting*) u svrhu sljedećeg odlomka.

Ako je vrijednost `\mubytein` jednaka nuli, tada *line-reprinting* radi kao i u standardnom T_EX-u. Ako je vrijednost `\mubytein` pozitivna te je vrijednost `\mubytelog`

jednaka nuli, tada *line-reprinting* uključuje izlaz ulaznog procesora encTeX-a. To znači da upravljačke riječi koje generira ulazni procesor mogu biti prikazane, čak i ako nisu prisutne u ulaznoj liniji. Ako su i `\mubytein` i `\mubytelog` pozitivne vrijednosti, tada line-reprinting radi bez ikakvih višebajtnih konverzija; koristi se samo xord kojeg neposredno slijedi xchr. U ovoj se situaciji ne koristi oblik $\hat{\hat{A}}$. Uočimo da poruka o greški može biti neobična kada su vrijednosti `\mubytein` i `\mubytelog` pozitivne:

```
\mubyte \cmd ABC \endmubyte \let\cmd=\undefined
\mubytein=1 \mubytelog=1
```

This is test of ABC and another text.

Dobivamo sljedeću poruku:

```
! Undefined control sequence.
1.3 This is test of ABC
and another and another text.
?
```

Pozivom `\show ABC` dobivamo više informacija:

```
> \cmd=undefined.
1.3 \show ABC
```

Brisanje zapisa iz tablice s konverzijama

Iz tablice konverzija moguće je obrisati samo zapise koji počinju s jednakim prvim bajtom kao *<niz bajtova>*. To se postiže naredbom

```
\mubyte <char> <char> \endmubyte.
```

Tako, na primjer, naredba

```
\mubyte A A \endmubyte
```

bríše sve nizove bajtova iz tablice s konverzijama koji počinju znakom A. Sljedeći kôd bríše cijelu tablicu s konverzijama:

```
{% \catcode' \hat@=12
\gdef \clearmbytes{ \bgroup \count255=1
\loop \uccode'X= \count255
\uppercase{ \mubyte XX \endmubyte}%
\advance \count255 by1
\ifnum \count255<256 \repeat
% \mubyte \hat@\hat@ \endmubyte
\egroup}
}
\clearmbytes
```

Ulazna i izlazna strana tablice s konverzijama

Tablica s konverzijama sastoji se od dva neovisna dijela: ulazne strane koju koristi ulazni procesor i izlazne strane koja se koristi pri izvršavanju `\write` ili tijekom ispisa u log datoteku i na terminal. Moguće je pohraniti zapis u samo jednu od navedenih strana koristeći neprazni *<opcionalni prefiks>*. Ako je *<opcionalni prefiks>* prazan, tada se isti zapis pohranjuje dvaput: na ulaznu i izlaznu stranu. Ako je *<opcionalni prefiks>* simbol catcode vrijednosti 8 (obično `_`), zapis se pohranjuje samo na ulaznu stranu tablice, a ako je *<opcionalni prefiks>* par simbola catcode vrijednosti 8 (obično `__`) tada se zapis pohranjuje samo na izlaznu stranu.

Ako opcionalni prefiks ima oblik `__` tada niz bajtova koji slijedi može biti prazan. U tom slučaju `encTEX` uklanja zapis koji odgovara simbolu `<prvi simbol>` iz izlazne strane tablice.

Kôd za brisanje tablice s konverzijama (naveden ranije) briše sve zapise iz ulazne strane tablice, ali iz izlazne strane brišu se samo zapisi koji se odnose na `<prvi simbol>` u jednobajtnom obliku. Zapis koji se odnosi na upravljačku riječ može se izbrisati iz izlazne strane samo naredbom `\mubyte \foo \endmubyte`.

Umetnute upravljačke riječi

Ako je `<prvi simbol>` upravljačka riječ, a `<opcionalni prefiks>` je simbol catcode vrijednosti 6 (obično znak #), iza kojeg slijedi `<broj>`, tada se `<broj>` bajtova čuva u ulaznom procesoru (što znači da se neće ponovno konvertirati), te se naznačena upravljačka riječ umeće prije `<niz bajtova>`, kao u sljedećem primjeru:

```
\def \abc{ABC}
\mubyte X BC \endmubyte \mubytein=1
\mubyte \foo \#3 \abc \endmubyte ABC is converted to \foo ABC
\mubyte \foo \#1 \abc \endmubyte ABC is converted to \foo AX
```

`<broj>` ima jednaku sintaksu kao `<number>` iz TeXbook-a. To znači da `<opcionalni razmak>` može imati ulogu separatora znamenki (vidi prethodni primjer).

Ako je `<broj>` jednak nuli, tada se umetne upravljačka riječ i cijeli `<niz bajtova>` ostaje nepromijenjen. Postiže se isti učinak kao kad je `<broj>` jednak duljini niza `<niz bajtova>`.

Prihvatljive vrijednosti za `<broj>` su 0 – 50. Negativne vrijednosti za `<broj>` se bez upozorenja ispisom tumače kao nula, a `<broj>` veći od 50 naznačuje da ostatak konvertirane linije neće biti promijenjen u ulaznom procesoru.

Slijedi više praktičnih primjera. Primijetimo, `<broj>` je ovdje veći od duljine niza `<niz bajtova>`.

```
\mubyte \warntwoytes \#2^^c3 \endmubyte
\mubyte \warntwoytes \#2^^c4 \endmubyte
\mubyte \warntwoytes \#2^^c5 \endmubyte
% itd...
\def \warntwoytes \#1\#2{ \message{WARNING: the UTF8 code:
\noconvert\#1 \noconvert\#2 is not defined i my macros.}}
```

U ovom primjeru koristi se nova temeljna naredba `\noconvert`.

Virtualna oznaka početka retka

Ako je vrijednost `\mubytein > 0` te je prvi bajt u `<niz bajtova>` jednak `\endlinechar` (što znači da `<niz bajtova>` ima oblik `<endlinechar><ostatak>`), tada ulazni procesor provjerava poklapa li se `<ostatak>` s početkom svakog pojedinog retka. Ako postoji poklapanje, izvršava se zadana konverzija, kao u sljedećem primjeru:

```
\bgroup \uccode'X= \endlinechar \uppercase{\gdef \echar{X}} \egroup
\mubyte \fooB \echar ABC \endmubyte %poklapanje s ABC na početku linije
\mubyte \fooE ABC \echar \endmubyte %poklapanje s ABC na kraju linije
\mubyte \fooW \spce \space ABC \space \endmubyte
%poklapanje s ABC uz razmak prije i nakon riječi
\mubyte \foo \# \echar XYZ \endmubyte %
%ako je XYZ na početku linije, ubacuje se \foo ispred njih
```

Sprečavanje ekspanzije u parametrima naredbe write

Ako je potrebno konvertirati upravljačke riječi u odgovarajuće nizove bajtova, tada nije poželjno da se dogodi ekspanzija. Takvo se ponašanje može spriječiti naredbom:

`\let\macro=\relax`, prije nego što `\write` započne ekspanziju svojeg parametra. No naredba `\write` radi asinkrono u većini situacija, te je moguće upravljati tisućama upravljačkih riječi koje su deklarirane kao UTF-8 kodovi. EncTeX nudi jednostavan alat za rješenje tog problema: ako je `\mubyteout ≥ 3`, tada encTeX daje značenje `\relax` svakoj upravljačkoj riječi deklariranom na izlaznoj strani tablice konverzija, prije nego što `\write` započne svoju ekspanziju, te im vraća originalno značenje neposredno nakon izvršenja naredbe `\write`. Primjer:

```
\mubyte \foo ABC \endmubyte \def\foo{macro body}
\mubyteout=2
\immediate \write16{testwrite: \foo} %ispisuje: 'testwrite: macro body'
\immediate \write16{testwrite: \noexpand \foo} %ispisuje: 'testwrite: ABC'
\mubyteout=3
\immediate \write16{testwrite: \foo} %ispisuje: 'testwrite: ABC'
\message{testmessage: \foo}
%ispisuje 'testmessage: macro body'
\message{testmessage: \noexpand \foo} %ispisuje: 'testmessage: \foo'
\edef\a{testedef: \foo}
%ekspandira se u sadržaj naredbe
\foo
\immediate \write16{ \meaning \foo}
%ispisuje '\relax'
\message{ \meaning \foo}
%ispisuje 'macro:->macro body'
```

Uočimo razliku između `\message` i `\immediate \write16`. Upravljačke riječi u parametru naredbe `\message` se uvijek ekspandiraju i nikad se ne konvertiraju u *<niz bajtova>*.

Moguće je postaviti *noexpand* zastavicu (samo za parametre naredbe `\write`) na bilo koju upravljačku riječ, te za nju nije potrebno deklarirati *<niz bajtova>*. U tu svrhu može se koristiti kôd poput sljedećeg:

```
\mubyte <upravljačka riječ> \relax \endmubyte.
```

Time se postiže jednak efekt kao naredbom

```
\mubyte <upravljačka riječ> \string <upravljačka riječ> \space \endmubyte,
```

no ovo rješenje zauzima više memorije jer TeX mora pohraniti *<niz bajtova>* kao string.

Mogu se definirati vlastite naredbe koje se ekspandiraju u jedan kôd u uobičajenoj upotrebi, a u drugi kôd u slučaju upotrebe kao parametar za `\write`. Preporučljivo je deklarirati upravljačku riječ `\writeparameter`:

```
\mubyte \writeparameter \relax \endmubyte \def\writeparameter{}
\def\mymacro{ \ifx \writeparameter \relax THIS CODE IS USED IN WRITE.
\else THIS CODE IS USED IN NORMAL EXPANSION. \fi}
```

Asinkrona naredba write i vrijednost mubyteout

Ako se ne koristi naredba `\immediate`, tada naredba `\write` prvo uzima parametre, a kasnije ih ekspandira i ispisuje. Naredba `\write` pohranjuje stvarne vrijednosti `\mubyteout` registra kad uzima parametre. Ta se vrijednost koristi kasnije kad se parametar ekspandira i ispisuje u datoteku.

Ovo svojstvo pruža mogućnost pisanja u više datoteka, pri čemu se kod pisanja u neke datoteke koristi konverzija u UTF-8, a kod ostalih nema konverzije (npr. ako se ta datoteka koristi kao ulaz za program koji ne može čitati datoteke kodirane s UTF-8). Može se zadati:

```
\newwrite \tocfile \newwrite \indexfile
\immediate \openout \tocfile= \jobname.toc
\immediate \openout \indexfile= \jobname.idx
\mubyteout=3
\write \tocfile{ovaj ce se parametar konvertirati u utf-8}
{ \mubyteout=0 \write \indexfile{ovaj parametar ostaje nepromijenjen}}
\write \tocfile{ovaj ce se parametar konvertirati u utf-8}
\end % sada se zbilja izvršavaju sve tri \write naredbe
```

4. Sažetak naredbi i opcija

Popis vrijednosti mubyteout

Osim vrijednosti 0, 1, 2 i 3, vrijednost `\mubyteout` registra može se postaviti i na `-1` ili `-2`. Sljedeća tablica sadržava značenja tih vrijednosti:

<code>\mubyteout</code>	<code><bajt>-><niz bajtova></code>	<code><upravljačka riječ>-><niz bajtova></code>	spriječena ekspanzija
0	isključeno	isključeno	isključeno
1	uključeno	isključeno	isključeno
2	uključeno	uključeno	isključeno
3	uključeno	uključeno	uključeno
-1	uključeno	isključeno	uključeno
-2	isključeno	isključeno	uključeno

Ako je aktivirana konverzija `<bajt> -> <niz bajtova>`, tada se konvertiraju svi tekstovi koji se ispisuju u `\write` datoteke, log datoteku i terminal. S druge strane, konverzija `<upravljačka riječ> -> <niz bajtova>` i spriječena ekspanzija odnose se samo na argumente naredbi `\write` i `\special`.

Popis argumenata naredbe special

U argumentima naredbe `\special` mogu se pojaviti obični (plain) tekstovi u jezicima različitim od engleskog. Ponekad je potrebno spremati te argumente kodirane u UTF-8, a `encTEX` pruža tu mogućnost. Argument naredbe `\special` procesira se s vrijednošću cjelobrojnog registra `\specialout`. Taj je registar uveden u `encTEX`-u i njegova je inicijalna vrijednost jednaka nuli.

```
\specialout=0 – nema konverzije, jednako kao kod originalnog TeX-a
\specialout=1 – izvršava se samo xchr konverzija
\specialout=2 – izvršava se samo \mubyteout konverzija
\specialout=3 – izvršava se \mubyteout konverzija, a potom i xchr konverzija
```

Naredba `\special` odmah ekspankira svoj argument. Ako je vrijednost `\specialout` jednaka 2 ili 3, tada se ekspanzija vrši `\mubyteout` vrijednošću na isti način kao i tijekom `\write` ekspanzije. Nadalje, naredba `\special` pohranjuje trenutne vrijednosti `\specialout` i `\mubyteout` registara i koristi ih u trenutku ispisa u `.dvi` datoteku.

Naredba noconvert

Naredba `\noconvert` uvedena u encTeX-u sprečava moguće konverzije sljedećeg znaka ili upravljačke riječi. Točnije: `\noconvert` je neekspandirajuća naredba i nema nikakvog utjecaja na ispis (slično kao `\relax`). Ako se koristi u argumentu za `\message` ili `\errmessage`, tada se upravljačka riječ ove naredbe ne ispisuje, a znak koji slijedi se ne konvertira u *<niz bajtova>* čak ni ako je vrijednost `\mubytelog` pozitivna te je znak zapisan na izlaznoj strani tablice konverzija.

Naredba `\noconvert` ima isti učinak u parametrima za naredbe `\write` i `\special`. Nadalje, ako iza nje dolazi upravljačka riječ, tada se ona normalno ispisuje čak i ako je vrijednost `\mubyteout` pozitivna te je upravljačka riječ zapisana na izlaznoj strani tablice konverzija.

Korištenje `\noconvert \noconvert` svodi se na jedan `\noconvert` na izlazu.

Naredba `\noconvert` normalno se ispisuje u log datoteku i terminal u ostalim slučajevima (kad se ne nalazi u parametrima za `\message` ili `\errmessage`) – npr., kod korištenja naredbe `\tracing`.

Cjelokupni popis encTeX naredbi

`\mubyte` – novi zapis u tablici konverzija

`\endmubyte` – separator za `\mubyte`

`\mubytein` – cjelobrojni registar; 0: suspregnuta višebajtna konverzija ulaza, 1 i više: aktivirana višebajtna konverzija ulaza.

`\mubyteout` – cjelobrojni registar; razina konverzije izlaza u parametrima za `\write` i `\special`.

`\mubytelog` – cjelobrojni registar; 0: suspregnuta višebajtna konverzija izlaza, 1 i više: aktivirana višebajtna konverzija izlaza.

`\specialout` – cjelobrojni registar, način konverzije parametra naredbe `\special`.

`\noconvert` – slično kao `\noexpand`, ali za proces konverzije.

`\xordcode` – pristup vektoru `xord`.

`\xchrcode` – pristup vektoru `xchr`.

`\xprncode` – pristup vektoru ispisivosti.

Slijedi popis vrijednosti opcionalnog prefiksa u naredbi `\mubyte`. Ovdje znak `#` znači da je simbol kategorije 6, a `-` znači da je simbol kategorije 7.

bez prefiksa – spremanje u ulaznu i izlaznu stranu tablice konverzija

`_` – spremanje samo u ulaznu stranu tablice konverzija

`__` – spremanje samo u izlaznu stranu tablice konverzija

`#<broj>` – spremanje u ulaznu stranu tablice konverzija; upravljačka riječ će biti umetnuta, *<broj>* bajtova neće se ponovno konvertirati

`\relax` – upravljačka riječ se neće ekspanirati u parametrima naredbe `\write`.

Postoji mogućnost da u sljedećim verzijama encTeX-a bude implementirano više prefiksa. Prvi znak prefiksa mora imati kategoriju koda različitu od 10, 11 i 12. To će se pravilo zadržati u nadolazećim verzijama, pa je dovoljno koristiti prvi znak iz *<niz bajtova>* s tom kategorijom ako nije potreban nikakav prefiks.

Lokalizacija T_EX-a za hrvatski književni jezik

Paket encT_EX sadrži tablice kodiranja uključene pomoću `\input` naredbe tijekom stvaranja formata. Te tablice podržavaju kodiranja koja se naširoko koriste u češkim tekstovima.

VI. Opis kreiranja formata hrtex

Hrvatska abeceda sadrži dva slova koja se ne nalaze u češkoj abecedi (pa nisu obuhvaćena paketom `encTEX`): Č i Đ, odnosno č i đ. U svrhu omogućavanja rada s tim slovima, bilo je potrebno:

1. Unijeti četiri nova retka u tablicu konverzija za UTF-8, respektivno za znakove Č, č, Đ, đ, pomoću naredbi `\mubyte ... \endmubyte`, kako bi ulazni procesor prepoznao navedene znakove pri čitanju datoteka. U tu svrhu napisana je datoteka `hrenc-u.tex`, po uzoru na datoteku `csenc-u.tex` iz paketa `encTEX`. Svaki uneseni redak sastoji se od para kodova, pri čemu lijeva strana sadrži kôd u internom T_EX kodiranju ISO 8859-2, dok desna sadrži kôd u UTF-8 kodiranju za dani znak.

2. Definirati koji će fontovi biti korišteni pri ispisivanju u datoteke. U tu svrhu napisana je datoteka `hrfonts.tex`, po uzoru na datoteku `csfonts.tex` iz paketa `encTEX`. U datoteci se redefinira T_EX naredba `\font` na način da se pri definiciji oznaka za Computer Modern fontove stvara *preusmjerenje* na odgovarajuće `hr-lm*` fontove.

3. U datoteci `lm-hr.map` opisano je kako se korišteni fontovi dobivaju rekodiranjem Latin Modern fontova pomoću datoteka `lm-hr.enc`, `lm-hrftt.enc` i `lm-hrsc.enc`. Datoteka `lm-hr.map` napisana je po uzoru na datoteku `lm-cs.map`. Struktura datoteke sastoji se redom od: stupca s imenima novih fontova (`hr-lm*`), stupca s punim nazivima odgovarajućeg Latin Modern fonta, stupca s imenom tablice korištene pri rekodiranju (imena tablica definirana su u datotekama s ekstenzijom `.enc`), stupca s odgovarajućom `.enc` datotekom i, na kraju, stupca s odgovarajućom `lm*.pfb` datotekom koja sadrži konkretne glifove iz skupine Latin Modern.

4. Napisati datoteke `lm-hr.enc`, `lm-hrftt.enc` i `lm-hrsc.enc`, koje sadrže tablice korištenih znakova respektivno za uobičajene, *typewriter* i *small caps* fontove. U tu svrhu, posuđene su datoteke za zamjenske fontove `lm-rep-cs*.enc` iz direktorija `/usr/share/doc/lmodern/font-substitution/`, u kojoj se nalaze potrebni retci za znakove iz ASCII tablice, sve češke/slovačke te još neke znakove. U tablicama su se već nalazili unosi za malo i veliko slovo Č, te je još bilo potrebno na odgovarajuće mjesto dodati unose za malo i veliko slovo Đ.

5. U skladu s datotekama `hrfonts.tex` i `lm-hr.map`, za definirane fontove generirati datoteke s ekstenzijom `.tfm` (T_EX Font Metric) potrebne za ispisivanje znakova u `.pdf/.dvi` dokument. Točnije, generirane su poveznice hrvatskih naziva (`hr-lm*.enc`) na konkretne datoteke analognih čeških/slovačkih naziva (`cs-lm*.enc`).

6. U sustav uključiti modificiranu verziju datoteke `plain.tex` te datoteku `slova.tex`, autora profesora Nenada Antonića, kojima se omogućava rad s dodatnim korisnim znakovima, kao i korištenje definicija za veličinu fontova po europskoj slovoslagarskoj tradiciji (*petit*, *borgis*, *garmond*, *cicero*, *mittel*, *canon*, *biscicero*).

Modifikacije u datoteci `plain.tex` su sljedeće:

– retci 399–493 u originalnoj datoteci zamijenjeni su retkom sadržaja `\input slova`. Izbačeni retci sadrže definicije oznaka raznih fontova raznih veličina za korištenje u dokumentima. U datoteci `slova.tex` uvode se drukčije oznake za te i druge fontove. Npr., u originalnoj datoteci uvodi se oznaka `\tenrm` za font Computer Modern Roman 10. Oznaka za taj font uvedena u datoteci `slova.tex` je `\xrm`, tj. dio koji označava veličinu je u obliku rimskog broja, a ne engleske riječi (kao u originalu). Također, uvedene su nove veličine u skladu s Cicero sustavom, koji se tradicionalno koristi u europskom slovoslagarstvu.

– redak 1099 u originalnoj datoteci mijenja se radi kompatibilnosti s gornjim promjenama.

- ubačena su tri nova retka kojima se postavlja vrijednost oznake `\rm` na `\xrm`.
- u retku 1241 u originalnoj datoteci promijenjena je oznaka samog formata iz `plain` u `HRplain`.

U svrhu kvalitetnog rastavljanja riječi na kraju retka, bilo je potrebno učiniti sljedeće:

7. Datoteka s obrascima za rastavljanje riječi za hrvatski jezik u paketu `hyph-utf8` zove se `hyph-hr.tex`. Ta je datoteka u ovom formatu nadjačana lokalnom datotekom `hyph-hr.tex` s obrascima za rastavljanje profesora Nenada Antonića. Ovi su obrasci dobiveni novim pristupom i znatno su manji od trenutnih obrazaca za hrvatski jezik prisutnih u `hyph-utf8`.

Zatim, bilo je potrebno:

8. Napisati datoteku `hrtex.tex`, po uzoru na `csplain.ini`, iz koje će se generirati format `hrtex`. `hrtex` omogućava korektnu obradu isključivo UTF-8 datoteka, odnosno, nisu omogućena neka druga kodiranja koja su bila omogućena `encTeX`-om.

9. Registrirati mapu `lm-hr.map` naredbom `updmap-sys --enable Map="lm-hr.map"`, čime se generira datoteka `10local.cfg`, te je omogućeno korištenje fontova navedenih u datoteci `lm-hr.map` uz kodiranje pomoću tablica u datotekama s ekstenzijom `.enc`.

10. Kreirati datoteku s formatom naredbom `pdftex -ini -enc -etex hrtex`.

Sve gore opisane datoteke (osim `10local.cfg`, koja se automatski generira na odgovarajućem mjestu) bilo je potrebno postaviti u direktorij `/usr/local/share/texmf/hrtex/`, u koji se, prema naputcima iz [7], trebaju spremati datoteke s lokalnim modifikacijama `TeX` sustava. Sljedeći je korak bio omogućiti korištenje formata s datotekama smještenima u spomenutom direktoriju iz bilo kojeg mjesta u sustavu.

11. Premjestiti stvorenu datoteku s formatom `hrtex.fmt` u direktorij

`/usr/local/share/texmf/web2c/` naredbom

```
mv /usr/local/share/texmf/hrtex/hrtex.fmt /usr/local/share/texmf/web2c/.
```

12. Omogućiti korištenje sintakse `hrtex dokument.tex` u radnom direktoriju po volji putem tekstualnog sučelja stvaranjem poveznice:

```
ln -s /usr/bin/pdftex /usr/local/bin/hrtex.
```

13. Dodati direktorij sa stvorenim poveznicama na `.tfm` datoteke u put traženja definiran varijablom `TFMFonts` u datoteci `/etc/texmf/texmf.d/55Fonts.cnf`, čime se omogućava nalaženje potrebnih `.tfm` datoteka pri korištenju formata `hrtex` iz bilo kojeg direktorija.

14. Dodati direktorij u kojem se nalaze kreirane `.enc` datoteke

(`/usr/local/share/texmf/hrtex/`) u put traženja definiran varijablom `ENCFONTS` u datoteci `/etc/texmf/texmf.d/85Misc.cnf`, čime se omogućava nalaženje potrebnih `.enc` datoteka pri korištenju formata `hrtex` iz bilo kojeg direktorija.

U nastavku se nalaze konkretnije upute kako omogućiti korištenje novog formata na računalu s operativnim sustavom baziranim na Linux-Debian distribuciji.

Postupak instalacije i korištenja formata hrtex

Za korištenje hrvatskih znakova s `plain TeX`-om u UTF-8 kodiranju pod operativnim sustavom baziranim na Linux-Debian distribuciji, potrebno je:

1. osigurati prisutnost paketa:

```
texlive-lang-czechslovak,
```

```
texlive-lang-croatian,
```

```
texlive-base-hyph-utf8,
```

```
texlive-base-enctex,  
lmodern.
```

2. u radni direktorij (`/usr/local/share/texmf/hrtex`) kopirati priložene datoteke (`hrenc-u.tex`, `hrfonts.tex`, `hyph-hr.tex`, `hrtex.tex`, `lm-hr.enc`, `lm-hrsc.enc`, `lm-hrtt.enc`, `lm-hr.map`, `slova.tex`, `plain.tex`)

3. u direktoriju (`/usr/local/share/texmf/hrtex/tfm`) generirati poveznice na `.tfm` datoteke oblika `cs-lm*`, potrebne za korištenje fontova:

```
for f in /usr/share/texmf/fonts/tfm/public/lm/cs-lm* ; do  
    fn=\$(basename "$f")  
    ln -s "$f" "/usr/local/share/texmf/hrtex/\${fn/cs/hr}"  
done
```

4. izvršiti `mktexlsr`

5. izvršiti `updmap-sys --enable Map="lm-hr".map`

6. izvršiti `mktexlsr`

7. izvršiti `pdftex -ini -enc -etex hrtex.tex`

8. izvršiti

```
mv /usr/local/share/texmf/hrtex/hrtex.fmt /usr/local/share/texmf/web2c/
```

9. izvršiti `ln -s /usr/bin/pdftex /usr/local/bin/hrtex`

10. izvršiti `mktexlsr`

11. u datoteci `/etc/texmf/texmf.d/85Misc.cnf` na kraj linije 5 dodati sadržaj:
`;$TEXMFLOCAL/hrtex/`

12. u datoteci `/etc/texmf/texmf.d/55Fonts.cnf` na kraj linije 9 dodati sadržaj:
`;$TEXMFLOCAL/hrtex/tfm/`

13. izvršiti `update-texmf`

14. izvršiti `mktexlsr`

15. korištenje: `hrtex dokument.tex`

Pomoću datoteka s internet stranice <http://web.studenti.math.hr/~majari/hr_{tex}> moguće je automatski provesti gore opisani postupak instalacije spremanjem spomenutih sadržaja u odgovarajući direktorij i pokretanjem instalacijske skripte.

Prilog 1: datoteka hrtex.tex

Ovo je datoteka pomoću koje se generira format `hrtex`. Unutar nje uključuju se datoteke `hrfonts.tex` i `hrenc-u.tex`, kao i (neposredno) lokalna verzija datoteke `plain.tex`. Ostatak koda preuzet je iz datoteke `csplain.ini`.

```

%%%%%%%% hrtex.tex plain TeX format for Croatian
%%%%%%%% based on csplain.ini, including etex extensions
%%%%%%%% 04.08.2011. MR/NA
% usage:
% pdftex -ini -enc -etex hrtex
% mv hrtex.fmt /usr/local/.../texmf/web2c/
% ln -s /usr/bin/pdftex /usr/local/bin/hrtex
% mktexlsr
\input hrfonts% re-defines primitive \font
\let\keptdump\dump \let\dump\relax
\input pdfetex.ini
\input eplain
\restorefont % original meaning of primitive \font
\input il2code % extra codes for czech / slovak letters in ISO-8859-2 encoding
\input plaina4 % \hsize and \vsize for A4
\everyjob=\expandafter{\the\everyjob
  \message{The format: hrplain <Jul. 2011>.}
  \message{The hr-fonts are preloaded and A4 size implicitly defined.}}
\input hrenc-u.tex % re-encoding using encTeX
\ifx\xpncode\undefined \else %% The encTeX is detected
  \count255=128 % All codes > 128 are printable:
  \loop \xpncode\count255=1 \advance\count255 by 1
  \ifnum \count255<256 \repeat
\fi
\ifnum\pdfoutput=0
  \message {jobname=hrplain with pdftex, DVI output initialised.}
  \pdfoutput=0
  \let\oripdfoutput=\pdfoutput \let\pdfoutput=\undefined
\else
  \message {jobname=pdfhrplain, PDF output initialised.}
  \openin0=pdfTEXconfig.tex
  \ifeof0 \message{WARNING: pdfTEXconfig.tex does not exist.
    I set \string\pdfoutput=1 only.}%
  \else \closein0 \input pdfTEXconfig.tex
  \fi
  \pdfoutput=1
\fi
\let\dump\keptdump \let\keptdump=\undefined
\dump

```

Prilog 2: datoteka hrfonts.tex

U ovoj datoteci definirano je kako se oznake fontova CM prevode u odgovarajuće hr-lm* fontove. Neke su oznake zadržale svoju CM vrijednost – npr. CM Math Italic i CM Math Symbols, jer ne postoji potreba za korištenjem hrvatskih znakova u tim fontovima.

```

%% hrfonts.tex, 04.08.2011. MR/NA
%% \font\xxx=cm* (no previous samples) \font\xxx=hr* (cm -> hr)

% CSFONTS.TEX, ver. 2.1, March 1996, by Petr Olsak (Ol\v s\'ak)
%
% This file re-defines the TeX primitive \font by the following way.
%
% You write: The meaning:
% \font\xxx=cmmi* ..... the same (math italic)
% \font\xxx=cmsy* ..... the same (math symbols)
% \font\xxx=cmbs* ..... the same (bold math symbols)
% \font\xxx=cmex* ..... the same (math extension)
% \font\xxx=cmte* ..... the same (math ext. tt)
% \font\xxx=cm* (no previous samples) \font\xxx=cs* (cm -> cs)
% \font\xxx=* (no previous samples) the same (no cm* fonts)
% \font\preloaded=*[scaled \magstep?] .... ignore loading %% ver. 2 feature

\ifx\transformfont\undefined \else \expandafter\endinput \fi

\catcode`\{=1 % left brace is begin-group character
\catcode`\}=2 % right brace is end-group character
\catcode`\#=6 % hash mark is macro parameter character

\def\transformfont#1#2#3#4#5 {%
  \if#1c%
    \if#2m%
      \if#3m%
        \if#4i%
          cmmi#5
        \else
          hr-lm#4#5%
        \fi
      \else
        \if#3b%
          \if#4s%
            cmbs#5
          \else
            hr-lmb#4#5
          \fi
        \else
          \if#3s%
            \if#4y%
              cmsy#5%
            \else
              \if#4l%
                hr-lmro#5
              \else
                hr-lmss#5
              \fi
            \fi
          \else
            \if#3e%
              \if#4x%
                cmex#5
              \else
                hr-lme#4#5%
              \fi
            \else
              \if#3t%
                \if#4e%
                  cmte#5
                \fi
              \fi
            \fi
          \fi
        \fi
      \fi
    \fi
  \fi
}

```

```

        \else%
        \if#4i%
        hr-lmri#5
        \else
        \if#4t%
        hr-lmtt#5
        \else
        cm#3#4#5
        \fi
\fi
        \fi
        \else
        hr-lm#3#4#5
        \fi
        \fi
        \fi
        \fi
        \else
        c#2#3#4#5
        \fi
        \else
        #1#2#3#4#5
        \fi}

\let\originalfont\font
\edef\tempb{\string\preloaded}
\def\font#1=#2 {\edef\tempa{\string#1}%
  \ifx\tempa\tempb\let\next=\ignoreload
  \else \def\next{\originalfont#1=\transformfont#2 }%
  \fi\next}
\long\def\ignoreload#1{\if#1s\let\next=\skipscaled
  \else\long\def\next{#1}\fi\next}
\def\skipscaled #1\magstep#2{}
\def\restorefont{\let\font\originalfont \let\originalfont\undefined
  \let\transformfont\undefined \let\restorefont\undefined}

% end of file HRFONTS.TEX

```

Prilog 3: datoteka hrenc-u.tex

Ova datoteka sastoji se od redaka oblika `\mubyte ... \endmubyte`, u koju su dodani retci za malo i veliko slovo Ć, te malo i veliko slovo Đ. Time se omogućava obrada tih znakova pri obradi datoteka u UTF-8 kodiranju, kako je opisano u poglavlju 5.

```

%% The re-encoding table for Croatian alphabet using encTeX, with ĆćĐđ,
%% 29.07.2011. MR/NA
%% based on Czech/Slovak format as described below.

%% The re-encoding table for Czech/Slovak alphabet using encTeX,
%% v.Sep.2005 (C) Petr Olšák
%% input: UTF8, internal TeX: ISO-8859-2
%% This file may be used while csplain, cseplain or cslatex is generated
%% example: tex -ini -enc \let\enc=u \input csplain.ini

% This is very simple setting of Czech and Slovak alphabet only.
% No special characters (such as dash, quotes etc.) used in UTF8 are
% declared here. No "UTF8 unknown code checking" is implemented here.
% The force is in the simplicity. The \mubyteout is set to 1 only.
% User can reset this value and add his own declaration.

\ifx\mubyte\undefined
  \errhelp={You has specified \let\enc=u but this works only with encTeX
  extension of TeX binary. See ftp://math.feld.cvut.cz/pub/olsak/encTeX.}
  \errmessage{The encTeX Feb2003 or later is not detected -- re-encoding is impossible}
  \endinput \fi

%%      TeX      input
\mubyte ^^c1 ^^c3^^81\endmubyte % \'A
\mubyte ^^e1 ^^c3^^a1\endmubyte % \'a
\mubyte ^^c9 ^^c3^^89\endmubyte % \'E
\mubyte ^^e9 ^^c3^^a9\endmubyte % \'e
\mubyte ^^cd ^^c3^^8d\endmubyte % \'I
\mubyte ^^ed ^^c3^^ad\endmubyte % \'i
\mubyte ^^d3 ^^c3^^93\endmubyte % \'O
\mubyte ^^f3 ^^c3^^b3\endmubyte % \'o
\mubyte ^^da ^^c3^^9a\endmubyte % \'U
\mubyte ^^fa ^^c3^^ba\endmubyte % \'u
\mubyte ^^dd ^^c3^^9d\endmubyte % \'Y
\mubyte ^^fd ^^c3^^bd\endmubyte % \'y
\mubyte ^^d4 ^^c3^^94\endmubyte % ^\O
\mubyte ^^f4 ^^c3^^b4\endmubyte % ^\o
\mubyte ^^c4 ^^c3^^84\endmubyte % \'A
\mubyte ^^e4 ^^c3^^a4\endmubyte % \'a
\mubyte ^^d6 ^^c3^^96\endmubyte % \'O
\mubyte ^^f6 ^^c3^^b6\endmubyte % \'o
\mubyte ^^dc ^^c3^^9c\endmubyte % \'U
\mubyte ^^fc ^^c3^^bc\endmubyte % \'u
\mubyte ^^c6 ^^c4^^86\endmubyte % veliko C'
\mubyte ^^e6 ^^c4^^87\endmubyte % malo c'
\mubyte ^^c8 ^^c4^^8c\endmubyte % \v C
\mubyte ^^e8 ^^c4^^8d\endmubyte % \v c
\mubyte ^^cf ^^c4^^8e\endmubyte % \v D
\mubyte ^^ef ^^c4^^8f\endmubyte % \v d
\mubyte ^^d0 ^^c4^^90\endmubyte % \Dj
\mubyte ^^f0 ^^c4^^91\endmubyte % \dj
\mubyte ^^cc ^^c4^^9a\endmubyte % \v E
\mubyte ^^ec ^^c4^^9b\endmubyte % \v e
\mubyte ^^c5 ^^c4^^b9\endmubyte % \' L
\mubyte ^^e5 ^^c4^^ba\endmubyte % \' l
\mubyte ^^a5 ^^c4^^bd\endmubyte % \v L
\mubyte ^^b5 ^^c4^^be\endmubyte % \v l
\mubyte ^^d2 ^^c5^^87\endmubyte % \v N
\mubyte ^^f2 ^^c5^^88\endmubyte % \v n
\mubyte ^^d8 ^^c5^^98\endmubyte % \v R
\mubyte ^^f8 ^^c5^^99\endmubyte % \v r

```



```
\mubyte ^^a9 ^^c5^^a0\endmubyte % \v S
\mubyte ^^b9 ^^c5^^a1\endmubyte % \v s
\mubyte ^^ab ^^c5^^a4\endmubyte % \v T
\mubyte ^^bb ^^c5^^a5\endmubyte % \v t
\mubyte ^^d9 ^^c5^^ae\endmubyte % \r U
\mubyte ^^f9 ^^c5^^af\endmubyte % \r u
\mubyte ^^ae ^^c5^^bd\endmubyte % \v Z
\mubyte ^^be ^^c5^^be\endmubyte % \v z
\mubyte ^^c0 ^^c5^^94\endmubyte % \' R
\mubyte ^^e0 ^^c5^^95\endmubyte % \' r

\global\everyjob=\expandafter{\the\everyjob
  \message{The UTF8->ISO-8859-2 re-encoding of Croatian alphabet activated by encTeX}}

\mubytein=1 \mubyteout=1 \mubytelogs=1

\endinput
```

Prilog 4: datoteka lm-hr.map

Datoteka `lm-hr.map` sadrži imena novouvedenih fontova `hr-lm*`, te podatke o tome kako dobiti fontove korištenjem odgovarajućih postojećih resursa (datoteke Latin Modern fontova) i ispravnog kodiranja (novouvedene tablice kodiranja, `lm-hr*.enc`). Uvedeni su analogoni svih fontova navedenih u poglavlju 3.

```

%%
%% lm-cs.map modified into lm-hr.map
%% 04.08.2011., MR/NA
%%

% This file belongs to the Latin Modern package. The work is released under
% the GUST Font License. See the MANIFEST-Latin-Modern.txt and
% README-Latin-Modern.txt files for the details. For the most recent version of
% this license see http://www.gust.org.pl/fonts/licenses/GUST-FONT-LICENSE.txt
% or http://tug.org/fonts/licenses/GUST-FONT-LICENSE.txt

% Prepared by Thomas Esser and Staszek Wawrykiewicz (May 14, 2005).
% Touched by Bogusław Jackowski (April 13, 2006).
% Support for CS (Czech and Slovak) text fonts in Type1 format:
% map lines which replace the new cstext.map (or parts of an old cs.map).
% 21.01.2008: PostScript internal font names changed

hr-lmb10    LMRomanDemi10-Regular    "enclmhr ReEncodeFont"    <lm-hr.enc <lmb10.pfb
hr-lmbx10   LMRoman10-Bold          "enclmhr ReEncodeFont"    <lm-hr.enc <lmbx10.pfb
hr-lmbx12   LMRoman12-Bold          "enclmhr ReEncodeFont"    <lm-hr.enc <lmbx12.pfb
hr-lmbx5    LMRoman5-Bold            "enclmhr ReEncodeFont"    <lm-hr.enc <lmbx5.pfb
hr-lmbx6    LMRoman6-Bold            "enclmhr ReEncodeFont"    <lm-hr.enc <lmbx6.pfb
hr-lmbx7    LMRoman7-Bold            "enclmhr ReEncodeFont"    <lm-hr.enc <lmbx7.pfb
hr-lmbx8    LMRoman8-Bold            "enclmhr ReEncodeFont"    <lm-hr.enc <lmbx8.pfb
hr-lmbx9    LMRoman9-Bold            "enclmhr ReEncodeFont"    <lm-hr.enc <lmbx9.pfb
hr-lmbxo10  LMRomanSlant10-Bold       "enclmhr ReEncodeFont"    <lm-hr.enc <lmbxo10.pfb
hr-lmbxi10  LMRoman10-BoldItalic      "enclmhr ReEncodeFont"    <lm-hr.enc <lmbxi10.pfb
hr-lmcs10   LMRomanCaps10-Regular     "enclmhrsc ReEncodeFont"  <lm-hrsc.enc <lmcs10.pfb
hr-lmdunh10 LMRomanDunh10-Regular     "enclmhr ReEncodeFont"    <lm-hrsc.enc <lmdunh10.pfb
hr-lmtti10  LMMono10-Italic           "enclmhrtt ReEncodeFont"  <lm-hrsc.enc <lmtti10.pfb
hr-lmr10    LMRoman10-Regular         "enclmhr ReEncodeFont"    <lm-hr.enc <lmr10.pfb
hr-lmr12    LMRoman12-Regular         "enclmhr ReEncodeFont"    <lm-hr.enc <lmr12.pfb
hr-lmr17    LMRoman17-Regular         "enclmhr ReEncodeFont"    <lm-hr.enc <lmr17.pfb
hr-lmr5     LMRoman5-Regular         "enclmhrsc ReEncodeFont"  <lm-hrsc.enc <lmr5.pfb
hr-lmr6     LMRoman6-Regular         "enclmhr ReEncodeFont"    <lm-hr.enc <lmr6.pfb
hr-lmr7     LMRoman7-Regular         "enclmhr ReEncodeFont"    <lm-hr.enc <lmr7.pfb
hr-lmr8     LMRoman8-Regular         "enclmhr ReEncodeFont"    <lm-hr.enc <lmr8.pfb
hr-lmr9     LMRoman9-Regular         "enclmhr ReEncodeFont"    <lm-hr.enc <lmr9.pfb
hr-lmro10   LMRomanSlant10-Regular    "enclmhr ReEncodeFont"    <lm-hr.enc <lmro10.pfb
hr-lmro12   LMRomanSlant12-Regular    "enclmhr ReEncodeFont"    <lm-hr.enc <lmro12.pfb
hr-lmro8    LMRomanSlant8-Regular     "enclmhr ReEncodeFont"    <lm-hr.enc <lmro8.pfb
hr-lmro9    LMRomanSlant9-Regular     "enclmhr ReEncodeFont"    <lm-hr.enc <lmro9.pfb
hr-lmtto10  LMMonoSlant10-Regular     "enclmhrtt ReEncodeFont"  <lm-hrsc.enc <lmtto10.pfb
hr-lmss10   LMSans10-Regular         "enclmhr ReEncodeFont"    <lm-hr.enc <lmss10.pfb
hr-lmss12   LMSans12-Regular         "enclmhr ReEncodeFont"    <lm-hr.enc <lmss12.pfb
hr-lmss17   LMSans17-Regular         "enclmhr ReEncodeFont"    <lm-hr.enc <lmss17.pfb
hr-lmss8    LMSans8-Regular          "enclmhr ReEncodeFont"    <lm-hr.enc <lmss8.pfb
hr-lmss9    LMSans9-Regular          "enclmhr ReEncodeFont"    <lm-hr.enc <lmss9.pfb
hr-lmssbx10 LMSans10-Bold            "enclmhr ReEncodeFont"    <lm-hr.enc <lmssbx10.pfb
hr-lmssdc10 LMSansDemiCond10-Regular "enclmhr ReEncodeFont"    <lm-hr.enc <lmssdc10.pfb
hr-lmssso10 LMSans10-Oblique         "enclmhr ReEncodeFont"    <lm-hr.enc <lmssso10.pfb
hr-lmssso12 LMSans12-Oblique         "enclmhr ReEncodeFont"    <lm-hr.enc <lmssso12.pfb
hr-lmssso17 LMSans17-Oblique         "enclmhr ReEncodeFont"    <lm-hr.enc <lmssso17.pfb
hr-lmssso8  LMSans8-Oblique          "enclmhr ReEncodeFont"    <lm-hr.enc <lmssso8.pfb
hr-lmssso9  LMSans9-Oblique          "enclmhr ReEncodeFont"    <lm-hr.enc <lmssso9.pfb
hr-lmssq8   LMSansQuot8-Regular      "enclmhr ReEncodeFont"    <lm-hr.enc <lmssq8.pfb
hr-lmssqo8  LMSansQuot8-Oblique      "enclmhr ReEncodeFont"    <lm-hr.enc <lmssqo8.pfb
hr-lmtcsc10 LMMonoCaps10-Regular     "enclmhrtt ReEncodeFont"  <lm-hrsc.enc <lmtcsc10.pfb
hr-lmri10   LMRoman10-Italic         "enclmhr ReEncodeFont"    <lm-hr.enc <lmri10.pfb
hr-lmri12   LMRoman12-Italic         "enclmhr ReEncodeFont"    <lm-hr.enc <lmri12.pfb

```

hr-lmri7	LMRoman7-Italic	"enclmhr ReEncodeFont"	<lm-hr.enc <lmri7.pfb
hr-lmri8	LMRoman8-Italic	"enclmhr ReEncodeFont"	<lm-hr.enc <lmri8.pfb
hr-lmri9	LMRoman9-Italic	"enclmhr ReEncodeFont"	<lm-hr.enc <lmri9.pfb
hr-lmtt10	LMMono10-Regular	"enclmhrtt ReEncodeFont"	<lm-hrtt.enc <lmtt10.pfb
hr-lmtt12	LMMono12-Regular	"enclmhrtt ReEncodeFont"	<lm-hrtt.enc <lmtt12.pfb
hr-lmtt8	LMMono8-Regular	"enclmhrtt ReEncodeFont"	<lm-hrtt.enc <lmtt8.pfb
hr-lmtt9	LMMono9-Regular	"enclmhrtt ReEncodeFont"	<lm-hrtt.enc <lmtt9.pfb
hr-lmu10	LMRomanUnsl10-Regular	"enclmhr ReEncodeFont"	<lm-hr.enc <lmu10.pfb
hr-lmvt10	LMMonoProp10-Regular	"enclmhrtt ReEncodeFont"	<lm-hrtt.enc <lmvt10.pfb

Prilog 5: datoteka lm-hr.enc

Datoteka `lm-hr.enc` sadrži tablicu znakova (*vektor kodiranja*) za uobičajene tekstualne fontove. Sastoji se od retka koji sadrži ime tablice (to se ime koristi u `lm-hr.map` datoteci), nakon kojeg slijedi 256 oznaka znakova, jedna po retku. Redni broj retka u tablici je ujedno pozicija kodiranja odgovarajućeg znaka u ISO 8859-2 kodiranju. Na analogan način načinjene su i datoteke `lm-hrsc.enc` i `lm-hrtt.enc`, koje sadrže tablice za *small caps*, odnosno *typewriter* fontove.

```
% This file belongs to the Latin Modern package. The work is released under
% the GUST Font License. See the MANIFEST-Latin-Modern.txt and
% README-Latin-Modern.txt files for the details. For the most recent version of
% this license see http://www.gust.org.pl/fonts/licenses/GUST-FONT-LICENSE.txt
% or http://tug.org/fonts/licenses/GUST-FONT-LICENSE.txt
```

<code>/enclmhr[</code>	<code>/zero</code>
<code>/Gamma</code>	<code>/one</code>
<code>/Delta</code>	<code>/two</code>
<code>/Theta</code>	<code>/three</code>
<code>/Lambda</code>	<code>/four</code>
<code>/Xi</code>	<code>/five</code>
<code>/Pi</code>	<code>/six</code>
<code>/Sigma</code>	<code>/seven</code>
<code>/Upsilon</code>	<code>/eight</code>
<code>/Phi</code>	<code>/nine</code>
<code>/Psi</code>	<code>/colon</code>
<code>/Omega</code>	<code>/semicolon</code>
<code>/ff</code>	<code>/exclamdown</code>
<code>/fi</code>	<code>/equal</code>
<code>/fl</code>	<code>/questiondown</code>
<code>/ffi</code>	<code>/question</code>
<code>/ffl</code>	<code>/at</code>
<code>/dotlessi</code>	<code>/A</code>
<code>/dotlessj</code>	<code>/B</code>
<code>/grave</code>	<code>/C</code>
<code>/acute</code>	<code>/D</code>
<code>/caron</code>	<code>/E</code>
<code>/breve</code>	<code>/F</code>
<code>/macron</code>	<code>/G</code>
<code>/ring</code>	<code>/H</code>
<code>/cedilla</code>	<code>/I</code>
<code>/germandbls</code>	<code>/J</code>
<code>/ae</code>	<code>/K</code>
<code>/oe</code>	<code>/L</code>
<code>/oslash</code>	<code>/M</code>
<code>/AE</code>	<code>/N</code>
<code>/OE</code>	<code>/O</code>
<code>/Oslash</code>	<code>/P</code>
<code>/suppress</code>	<code>/Q</code>
<code>/exclam</code>	<code>/R</code>
<code>/quotedblright.cm</code>	<code>/S</code>
<code>/numbersign</code>	<code>/T</code>
<code>/dollar</code>	<code>/U</code>
<code>/percent</code>	<code>/V</code>
<code>/ampersand</code>	<code>/W</code>
<code>/quoteright</code>	<code>/X</code>
<code>/parenleft</code>	<code>/Y</code>
<code>/parenright</code>	<code>/Z</code>
<code>/asterisk</code>	<code>/bracketleft</code>
<code>/plus</code>	<code>/quotedblleft.cm</code>
<code>/comma</code>	<code>/bracketright</code>
<code>/hyphen</code>	<code>/circumflex</code>
<code>/period</code>	<code>/dotaccent</code>
<code>/slash</code>	<code>/quoteleft</code>
	<code>/a</code>

/b	/.notdef
/c	/.notdef
/d	/.notdef
/e	/Scaron
/f	/.notdef
/g	/Tcaron
/h	/.notdef
/i	/.notdef
/j	/Zcaron
/k	/.notdef
/l	/.notdef
/m	/.notdef
/n	/.notdef
/o	/.notdef
/p	/.notdef
/q	/lcaron
/r	/.notdef
/s	/.notdef
/t	/agrave
/u	/scaron
/v	/.notdef
/w	/tcaron
/x	/.notdef
/y	/.notdef
/z	/zcaron
/endash	/.notdef
/emdash	/Racute
/hungarumlaut	/Acute
/tilde	/.notdef
/dieresis	/.notdef
/.notdef	/Adieresis
/.notdef	/Lacute
/.notdef	/Cacute
/.notdef	/.notdef
/.notdef	/Ccaron
/.notdef	/Eacute
/.notdef	/.notdef
/.notdef	/.notdef
/.notdef	/Ecaron
/.notdef	/Iacute
/.notdef	/.notdef
/.notdef	/Dcaron
/.notdef	/Dcroat
/perthousand	/.notdef
/.notdef	/Ncaron
/.notdef	/Oacute
/.notdef	/Ocircumflex
/.notdef	/.notdef
/.notdef	/Odieresis
/.notdef	/.notdef
/.notdef	/Rcaron
/.notdef	/Uring
/.notdef	/Uacute
/.notdef	/.notdef
/Agrave	/Udieresis
/.notdef	/Yacute
/.notdef	/.notdef
/.notdef	/.notdef
/hyphen.dup	/racute
/ogonek	/aacute
/guillemotleft	/.notdef
/guillemotright	/.notdef
/.notdef	/adieresis
/.notdef	/lacute
/.notdef	/cacute
/.notdef	/.notdef
/.notdef	/ccaron
/Lcaron	/eacute

Lokalizacija T_EX-a za hrvatski književni jezik

```
/.notdef
/.notdef
/ecaron
/iacute
/.notdef
/dcaron
/dcroat
/.notdef
/ncaron
/oacute
/ocircumflex
/.notdef

/odieresis
/.notdef
/rcaron
/uring
/uacute
/.notdef
/udieresis
/yacute
/quotedblbase.cs
/quotedblright.cs
] def
```

Prilog 6: datoteka slova.tex

U ovoj datoteci definirane su oznake za razne fontove i korisne znakove. U nastavku slijedi tekst preuzet iz komentara originalne datoteke `slova.tex`.

Osnovnih 16 familija matematičkih znakova

Ove se familije koriste i kao osnovna slova u tekstu, ali prvenstvena je prednost za fina matematička smanjivanja i uvećanja.

- 0. rm: roman, tj. znakovi s tipkovnice i uspravna velika slova grčkog alfabeta
- 1. mit: matematički kurziv, mala grčka slova
- 2. cal: lijepo pisana velika slova i matematički simboli
- 3. <ex>: veliki operatori (nije još eksplicitno pozivano)
- 4. it: tekstualni kurziv
- 5. sl: nakošena tekstualna slova
- 6. bf: masna slova
- 7. tt: tipkaći stroj
- 8. sf: sans serif
- 9. mib: masni matematički simboli, raspored kao i 1 (može se koristiti `{\mib\alpha}` i sl.)
- 10. bsy: kao 2, samo masno (`{\bsy A}` daje masno pisano A)
- 11. fm: fraktur (velika i mala slova)
- 12. fb: masni fraktur
- 13. sam: AMS simboli A
- 14. sbm: AMS simboli B
- 15. sfb: sans serif bold

Gornji se znakovi mogu koristiti u sljedećim priređenim veličinama (veličine su izražene u broju *anglosaksonskih* točaka, a ne kontinentalnih):

- petit: 8
- borgis: 9
- garmond: 10
- hypocicero: 11 (izmišljeno ime za popularnu veličinu)
- cicero: 12
- mittel: 14 (preciznije, 14,4pt)
- hyppertertia: 17 (preciznije, 17,28pt; ime je izmišljeno, tertia – 16dd)
- canon: 20 (preciznije, 20,736)
- biscicero: 24 (preciznije, 24,8832; nema punu funkcionalnost)

Manje veličine: colonel (7), nonpareille (6), perl (5), ... nemaju smisla za matematički tekst; naravno, indeksi se koriste do 5pt; ali su definirani za normalne veličine $\geq 8pt$.

```
% aktivirati ako se koristi plain TeX, a ne iniTeX
%\catcode'@=11 % borrow the private macros of PLAIN (with care)

\def\@magscale#1{ scaled \magstep #1}
\def\@halfmag{ scaled \magstephalf}
\def\@ptscale#1{ scaled #100}

%
%
% TEN point X
%
\font\xrm = cmr10 % roman
```

```

\font\xmi = cmmi10 % math italic
\skewchar\xmi ='177 % for placement of accents
\font\xsy = cmsy10 % math symbols
\skewchar\xsy ='60 % for placement of math accents
\font\xex = cmex10 % extended symbols
\font\xmib = cmmib10 % bold math italic
\font\xbsy = cmbsty10 % bold symbols
\font\xit = cmti10 % text italic
\font\xsl = cmsl10 % slanted
\font\xbf = cmbx10 % extended bold
% \font\xbfs = cmbxsl10 % extended bold slanted
\font\xtt = cmitt10 % typewriter
\hyphenchar\xtt = -1 % suppress hyphenation in \tt font
% \font\xtti = cmitt10 % italic typewriter
% \font\xtts = cmslitt10 % slanted typewriter
\font\xsf = cmss10 % sans serif
\font\xsfi = cmssi10 % italic sans serif
\font\xsfb = cmssbx10 % bold sans serif
% \font\xsc = cmcsc10 % small caps
% \font\xuit = cmu10 % unslanted italic
% \font\xbfti = cmbxti10 % bold text italic
% \font\xbfnx = cmb10 % bold (not extended!)
% \font\xdunh = cmdunh10 % Dunhill
% \font\xff = cmff10 % funny font
% \font\xfi = cmfi10 % funny font italic
% \font\xssdc = cmssdc10 % sans serif bold condensed
% \font\xtcsc = cmtcsc10 %
% \font\xtex = cmtex10 % stanford keyboard typewriter
% \font\xvtt = cmvtt10 % 'v' typewriter
% \font\xbxcd = cmbxcd10 % nepotpun, samo brojevi i velika slova
% \font\xscscsl = cmcscsl10 % small caps slanted
% \font\xssbxo = cmssbxo10 %
% \font\xvtti = cmvtti10 % 'v' typewriter italic

% other fonts (not cm)
% fraktur
% \font\xeuex = euex10 % extended math symbols
\font\xeufb = eufb10 % fraktur bold
\font\xeufm = eufm10 % fraktur
% \font\xeurb = eurb10 % stylus bold
% \font\xeurm = eurm10 % stylus
% \font\xeusb = eusb10 % nepotpuno, masna velika pisana slova
% \font\xeusm = eusm10 % nepotpuno, velika pisana slova

% additional math symbols
\font\xmsam = msam10 % additional math symbols
\font\xmsbm = msbm10 % bb bold & additional symbols
% \font\xbbold = bbold10 % blackboard bold

% cyrillic
% \font\xcyb = wncyb10 %
% \font\xcyi = wncyi10 %
% \font\xcyr = wncy10 %
% \font\xcyssc = wncyssc10 %
% \font\xcyssf = wncyssf10 %

%
%
% ELEVEN point XI
%
\font\xirm = cmr10 % @halfmag
\font\ximi = cmmi10 % @halfmag
\skewchar\ximi ='177
\font\xisy = cmsy10 % @halfmag
\skewchar\xisy ='60
\font\xiex = cmex10 % @halfmag
\font\ximib = cmmib10 % @halfmag

```



```

\font\xibsy = cmbsty10      \@halfmag
\font\xiit  = cmti10       \@halfmag
\font\xisl  = cmsl10       \@halfmag
\font\xibf  = cmbx10       \@halfmag
\font\xitt  = cmtt10       \@halfmag
  \hyphenchar\xitt = -1
\font\xisf  = cmss10       \@halfmag
\font\xisfb = cmssbx10     \@halfmag
\font\xieufb = eufb10     \@halfmag
\font\xieufm = eufm10     \@halfmag
\font\ximsam = msam10     \@halfmag
\font\ximsbm = msbm10     \@halfmag

%
%
% TWELVE point XII
%
\font\xiirm = cmr12
\font\xiimi = cmmi12
  \skewchar\xiimi = '177
\font\xiisy = cmsy10 \@magscale1
  \skewchar\xiisy = '60
\font\xiidx = cmex10 \@magscale1
\font\xiimib = cmmib10 \@magscale1
\font\xiibsy = cmbsty10 \@magscale1
\font\xiit  = cmti12
\font\xisl  = cmsl12
\font\xibf  = cmbx12
\font\xitt  = cmtt12
  \hyphenchar\xitt = -1
\font\xisf  = cmss12
\font\xisfb = cmssbx10 \@magscale1

\font\xieufb = eufb10 \@magscale1
\font\xieufm = eufm10 \@magscale1

% additional math symbols
\font\xiimsam = msam10 \@magscale1
\font\xiimsbm = msbm10 \@magscale1
% \font\xiibbold = bbold12

%
%
% FOURTEEN point XIV
%
\font\xivrm = cmr12 \@magscale1
\font\xivmi = cmmi12 \@magscale1
  \skewchar\xivmi = '177
\font\xivsy = cmsy10 \@magscale2
  \skewchar\xivsy = '60
\font\xivdx = cmex10 \@magscale2
\font\xivmib = cmmib10 \@magscale2
\font\xivbsy = cmbsty10 \@magscale2
\font\xivit  = cmti12 \@magscale1
\font\xivsl  = cmsl12 \@magscale1
\font\xivbf  = cmbx12 \@magscale1
\font\xivtt  = cmtt12 \@magscale1
  \hyphenchar\xivtt = -1
\font\xivsf  = cmss12 \@magscale1
\font\xivsfb = cmssbx10 \@magscale2
\font\xiveufb = eufb10 \@magscale2
\font\xiveufm = eufm10 \@magscale2
% additional math symbols
\font\xivmsam = msam10 \@magscale2
\font\xivmsbm = msbm10 \@magscale2

%
%

```

```

% SEVENTEEN point XVII
%
\font\xviirm = cmr17
\font\xviimi = cmmi12 \@magscale2
\skewchar\xviimi ='177
\font\xviisy = cmsy10 \@magscale3
\skewchar\xviisy ='60
\font\xviiex = cmex10 \@magscale3
\font\xviimib = cmmib10\@magscale3
\font\xviibsy = cmbsty10\@magscale3
\font\xviiit = cmti12 \@magscale2
\font\xviisl = cmsl12 \@magscale2
\font\xviibf = cmbx12 \@magscale2
\font\xviitt = cmtt12 \@magscale2
\hyphenchar\xviitt = -1
\font\xviisf = cmss17
\font\xviisfb = cmssbx10 \@magscale3
\font\xviieufb = eufb10 \@magscale3
\font\xviieufm = eufm10 \@magscale3

% additional math symbols
\font\xviimsam = msam10 \@magscale3
\font\xviimsbm = msbm10 \@magscale3

%
%
% TWENTY point XX
%
\font\xxrm = cmr17 \@magscale1
\font\xxmi = cmmi12 \@magscale3
\skewchar\xxmi ='177
\font\xxsy = cmsy10 \@magscale4
\skewchar\xxsy ='60
\font\xxmib = cmmib10\@magscale4
\font\xxbsy = cmbsty10\@magscale4
\font\xxit = cmti12 \@magscale3
\font\xxsl = cmsl12 \@magscale3
\font\xxbf = cmbx12 \@magscale3
\font\xxtt = cmtt12 \@magscale3
\hyphenchar\xxtt = -1
\font\xxsf = cmss17 \@magscale1
\font\xxsfb = cmssbx10 \@magscale4

%
%
% TWENTY FIVE point XXV
%
\font\xxvrm = cmr17 \@magscale2
\font\xxvmi = cmmi12 \@magscale4
\skewchar\xxvmi ='177
\font\xxvsv = cmsy10 \@magscale5
\skewchar\xxvsv ='60
\font\xxvit = cmti12 \@magscale4
\font\xxvsl = cmsl12 \@magscale4
\font\xxvbf = cmbx12 \@magscale4
\font\xxvtt = cmtt12 \@magscale4
\hyphenchar\xxvtt = -1
\font\xxvsf = cmss17 \@magscale2

\newskip\ttglue

%
%
% PETIT VIII
%
\def\petit{\def\rm{\fam\z@\viiirm}%
\textfont0=\viiirm \scriptfont0=\virm \scriptscriptfont0=\vrm
\textfont1=\viiimi \scriptfont1=\vimi \scriptscriptfont1=\vimi

```

```

\def\mit{\fam\@ne}
  \textfont2=\viiiis \scriptfont2=\visy \scriptscriptfont2=\vsv
\def\cal{\fam\tw@}
  \textfont3=\viiiex \scriptfont3=\viiey \scriptscriptfont3=\viiey
% italics
  \def\it{\fam\itfam\viiiit}%
  \textfont\itfam=\viiiit
% slanted
  \def\sl{\fam\slfam\viiiisl}%
  \textfont\slfam=\viiiisl
% bold
  \def\bf{\fam\bffam\viiiibf}%
  \textfont\bffam=\viiiibf \scriptfont\bffam=\vibf
  \scriptscriptfont\bffam=\vbf
% typewriter
  \def\tt{\fam\ttfam\viiiitt}%
  \textfont\ttfam=\viiiitt
% sans serif
  \def\sf{\fam\sffam\viiiisf}%
  \textfont\sffam=\viiiisf \scriptfont\sffam=\visf
  \scriptscriptfont\sffam=\vsf
% bold math italics
  \def\mib{\fam\mibfam\viiiimib}%
  \textfont\mibfam=\viiiimib \scriptfont\mibfam=\vimib
  \scriptscriptfont\mibfam=\vmib
% bold math symbols
  \def\bsy{\fam\bsyfam\viiiibsy}%
  \textfont\bsyfam=\viiiibsy \scriptfont\bsyfam=\vibsy
  \scriptscriptfont\bsyfam=\vbsy
% fraktur
  \def\fm{\fam\fmfam\viiiieufm}%
  \textfont\fmfam=\viiiieufm \scriptfont\fmfam=\vieufm
  \scriptscriptfont\fmfam=\veufm
% bold fraktur
  \def\fb{\fam\fbfam\viiiieufb}%
  \textfont\fbfam=\viiiieufb \scriptfont\fbfam=\vieufb
  \scriptscriptfont\fbfam=\veufb
% AMS symbols A
  \def\sam{\fam\samfam\viiiisam}%
  \textfont\samfam=\viiiisam \scriptfont\samfam=\vimsam
  \scriptscriptfont\samfam=\vmsam
% AMS symbols B
  \def\sbm{\fam\sbmfam\viiiisbm}%
  \textfont\sbmfam=\viiiisbm \scriptfont\sbmfam=\vimsbm
  \scriptscriptfont\sbmfam=\vmsbm
% sans serif bold
  \def\sfb{\fam\sfbfam\viiiisfb}%
  \textfont\sfbfam=\viiiisfb \scriptfont\sfbfam=\viisfb
  \scriptscriptfont\sfbfam=\viisfb
%
  \tt \ttglue=.5em plus.25em minus.15em
  \normalbaselineskip=9pt
  \setbox\strutbox=\hbox{\vrule height8.5pt depth3.5pt width\z@}%
  \normalbaselines\rm}

%
%
% BORGIS IX
%
\def\borgis{\def\rm{\fam\z@\ixrm}%
  \textfont0=\ixrm \scriptfont0=\virm \scriptscriptfont0=\vrm
  \textfont1=\ixmi \scriptfont1=\vimi \scriptscriptfont1=\vimi
\def\mit{\fam\@ne}
  \textfont2=\ixsy \scriptfont2=\visy \scriptscriptfont2=\vsv
\def\cal{\fam\tw@}
  \textfont3=\ixex \scriptfont3=\viiey \scriptscriptfont3=\viiey

```

```

% italics
  \def\it{\fam\itfam\ixit}%
  \textfont\itfam=\ixit
% slanted
  \def\sl{\fam\slfam\ixsl}%
  \textfont\slfam=\ixsl
% bold
  \def\bf{\fam\bffam\ixbf}%
  \textfont\bffam=\ixbf \scriptfont\bffam=\vibf
  \scriptscriptfont\bffam=\vbf
% typewriter
  \def\tt{\fam\ttfam\ixtt}%
  \textfont\ttfam=\ixtt
% sans serif
  \def\sf{\fam\sffam\ixsf}%
  \textfont\sffam=\ixsf \scriptfont\sffam=\visf
  \scriptscriptfont\sffam=\vsf
% bold math italics
  \def\mib{\fam\mibfam\ixmib}%
  \textfont\mibfam=\ixmib \scriptfont\mibfam=\vimib
  \scriptscriptfont\mibfam=\vmib
% bold math symbols
  \def\bsy{\fam\bsyfam\ixbsy}%
  \textfont\bsyfam=\ixbsy \scriptfont\bsyfam=\vibsy
  \scriptscriptfont\bsyfam=\vbsy
% fraktur
  \def\fm{\fam\fmfam\ixeufm}%
  \textfont\fmfam=\ixeufm \scriptfont\fmfam=\vieufm
  \scriptscriptfont\fmfam=\veufm
% bold fraktur
  \def\fb{\fam\fbfam\ixeufb}%
  \textfont\fbfam=\ixeufb \scriptfont\fbfam=\vieufb
  \scriptscriptfont\fbfam=\veufb
% AMS symbols A
  \def\sam{\fam\samfam\ixmsam}%
  \textfont\samfam=\ixmsam \scriptfont\samfam=\vimsam
  \scriptscriptfont\samfam=\vmsam
% AMS symbols B
  \def\sbm{\fam\sbmfam\ixmsbm}%
  \textfont\sbmfam=\ixmsbm \scriptfont\sbmfam=\vimsbm
  \scriptscriptfont\sbmfam=\vmsbm
% sans serif bold
  \def\sfb{\fam\sfbfam\ixsfb}%
  \textfont\sfbfam=\ixsfb \scriptfont\sfbfam=\viisfb
  \scriptscriptfont\sfbfam=\viisfb
%
  \tt \ttglue=.5em plus.25em minus.15em
  \normalbaselineskip=11pt
  \setbox\strutbox=\hbox{\vrule height8.5pt depth3.5pt width\z@}%
  \normalbaselines\rm}

%
%
% GARMOND X
%
\def\garmond{\def\rm{\fam\z@\xrm}%
  \textfont0=\xrm \scriptfont0=\viirm \scriptscriptfont0=\vrm
  \textfont1=\xmi \scriptfont1=\viimi \scriptscriptfont1=\vmi
\def\mit{\fam\@ne}
  \textfont2=\xsy \scriptfont2=\viisy \scriptscriptfont2=\vsy
\def\cal{\fam\tw@}
  \textfont3=\xex \scriptfont3=\viiox \scriptscriptfont3=\viiox
% italics
  \def\it{\fam\itfam\xit}%
  \textfont\itfam=\xit

```

```

% slanted
  \def\sl{\fam\slfam\xsl}%
  \textfont\slfam=\xsl
% bold
  \def\bf{\fam\bffam\xbf}%
  \textfont\bffam=\xbf \scriptfont\bffam=\viibf
  \scriptscriptfont\bffam=\vbf
% typewriter
  \def\tt{\fam\ttfam\xtt}%
  \textfont\ttfam=\xtt
% sans serif
  \def\sf{\fam\sffam\xsf}%
  \textfont\sffam=\xsf \scriptfont\sffam=\viisf
  \scriptscriptfont\sffam=\vsf
% bold math italics
  \def\mib{\fam\mibfam\xmib}%
  \textfont\mibfam=\xmib \scriptfont\mibfam=\viimib
  \scriptscriptfont\mibfam=\vmib
% bold math symbols
  \def\bsy{\fam\bsyfam\xbsy}%
  \textfont\bsyfam=\xbsy \scriptfont\bsyfam=\viibsy
  \scriptscriptfont\bsyfam=\vbsy
% fraktur
  \def\fm{\fam\fmfam\xeufm}%
  \textfont\fmfam=\xeufm \scriptfont\fmfam=\viieufm
  \scriptscriptfont\fmfam=\veufm
% bold fraktur
  \def\fb{\fam\fbfam\xeufb}%
  \textfont\fbfam=\xeufb \scriptfont\fbfam=\viieufb
  \scriptscriptfont\fbfam=\veufb
% AMS symbols A
  \def\sam{\fam\samfam\xmsam}%
  \textfont\samfam=\xmsam \scriptfont\samfam=\viimsam
  \scriptscriptfont\samfam=\vmsam
% AMS symbols B
  \def\sbm{\fam\sbmfam\xmsbm}%
  \textfont\sbmfam=\xmsbm \scriptfont\sbmfam=\viimsbm
  \scriptscriptfont\sbmfam=\vmsbm
% sans serif bold
  \def\sfb{\fam\sfbfam\xsfb}%
  \textfont\sfbfam=\xsfb \scriptfont\sfbfam=\viisfb
  \scriptscriptfont\sfbfam=\viisfb
%
  \tt \ttglue=.5em plus.25em minus.15em
  \normalbaselineskip=12pt
  \setbox\strutbox=\hbox{\vrule height8.5pt depth3.5pt width\z@}%
  \normalbaselines\rm}

%
%
% HYPOCICERO XI
%
\def\hypocicero{\def\rm{\fam\z@\xirm}%
  \textfont0=\xirm \scriptfont0=\viiirm \scriptscriptfont0=\virm
  \textfont1=\ximi \scriptfont1=\viiimi \scriptscriptfont1=\vimi
\def\mit{\fam\@ne}
  \textfont2=\xisy \scriptfont2=\viiisy \scriptscriptfont2=\visy
\def\cal{\fam\tw@}
  \textfont3=\xiex \scriptfont3=\viiiex \scriptscriptfont3=\viex
% italics
  \def\it{\fam\itfam\xiit}%
  \textfont\itfam=\xiit
% slanted
  \def\sl{\fam\slfam\xisl}%
  \textfont\slfam=\xisl
% bold
  \def\bf{\fam\bffam\xibf}%

```

```

        \textfont\bffam=\xibf \scriptfont\bffam=\viibf
        \scriptscriptfont\bffam=\vibf
% typewriter
    \def\tt{\fam\ttfam\xitt}%
    \textfont\ttfam=\xitt
% sans serif
    \def\sf{\fam\sffam\xisf}%
    \textfont\sffam=\xisf \scriptfont\sffam=\viiisf
    \scriptscriptfont\sffam=\visf
% bold math italics
    \def\mib{\fam\mibfam\ximib}%
    \textfont\mibfam=\ximib \scriptfont\mibfam=\viiiimib
    \scriptscriptfont\mibfam=\vimib
% bold math symbols
    \def\bsy{\fam\bsyfam\xibsy}%
    \textfont\bsyfam=\xibsy \scriptfont\bsyfam=\viiiibsy
    \scriptscriptfont\bsyfam=\vibsy
% fraktur
    \def\fm{\fam\fmfam\xieufm}%
    \textfont\fmfam=\xieufm \scriptfont\fmfam=\viiiieufm
    \scriptscriptfont\fmfam=\vieufm
% bold fraktur
    \def\fb{\fam\fbfam\xieufb}%
    \textfont\fbfam=\xieufb \scriptfont\fbfam=\viiiieufb
    \scriptscriptfont\fbfam=\vieufb
% AMS symbols A
    \def\sam{\fam\samfam\ximsam}%
    \textfont\samfam=\ximsam \scriptfont\samfam=\viiiisam
    \scriptscriptfont\samfam=\vimsam
% AMS symbols B
    \def\sbm{\fam\sbmfam\ximsbm}%
    \textfont\sbmfam=\ximsbm \scriptfont\sbmfam=\viiiisbm
    \scriptscriptfont\sbmfam=\vimsbm
% sans serif bold
    \def\sfb{\fam\sfbfam\xisfb}%
    \textfont\sfbfam=\xisfb \scriptfont\sfbfam=\viiiisfb
    \scriptscriptfont\sfbfam=\viisfb
%
    \tt \ttglue=.5em plus.25em minus.15em
    \normalbaselineskip=13,2pt
    \setbox\strutbox=\hbox{\vrule height8.5pt depth3.5pt width\z@}%
    \normalbaselines\rm}

%
%
% CICERO XII
%
\def\cicero{\def\rm{\fam\z@\xiirm}%
    \textfont0=\xiirm \scriptfont0=\ixrm \scriptscriptfont0=\viirm
    \textfont1=\xiimi \scriptfont1=\ixmi \scriptscriptfont1=\viimi
\def\mit{\fam\@ne}
    \textfont2=\xiisy \scriptfont2=\ixsy \scriptscriptfont2=\viisy
\def\cal{\fam\tw@}
    \textfont3=\xiiey \scriptfont3=\ixey \scriptscriptfont3=\viiey
% italics
    \def\it{\fam\itfam\xiit}%
    \textfont\itfam=\xiit
% slanted
    \def\sl{\fam\slfam\xiisl}%
    \textfont\slfam=\xiisl
% bold
    \def\bf{\fam\bffam\xiibf}%
    \textfont\bffam=\xiibf \scriptfont\bffam=\ixbf
    \scriptscriptfont\bffam=\viibf
% typewriter
    \def\tt{\fam\ttfam\xiitt}%

```

```

\textfont\ttfam=\xiitt
% sans serif
\def\sf{\fam\sffam\xiisf}%
\textfont\sffam=\xiisf \scriptfont\sffam=\ixsf
\scriptscriptfont\sffam=\viisf
% bold math italics
\def\mib{\fam\mibfam\xiimib}%
\textfont\mibfam=\xiimib \scriptfont\mibfam=\ixmib
\scriptscriptfont\mibfam=\viimib
% bold math symbols
\def\bsy{\fam\bsyfam\xiibsy}%
\textfont\bsyfam=\xiibsy \scriptfont\bsyfam=\ixbsy
\scriptscriptfont\bsyfam=\viibsy
% fraktur
\def\fm{\fam\fmfam\xiieufm}%
\textfont\fmfam=\xiieufm \scriptfont\fmfam=\ixeufm
\scriptscriptfont\fmfam=\viieufm
% bold fraktur
\def\fb{\fam\fbfam\xiieufb}%
\textfont\fbfam=\xiieufb \scriptfont\fbfam=\ixeufb
\scriptscriptfont\fbfam=\viieufb
% AMS symbols A
\def\sam{\fam\samfam\xiimsam}%
\textfont\samfam=\xiimsam \scriptfont\samfam=\ixmsam
\scriptscriptfont\samfam=\viimsam
% AMS symbols B
\def\sbm{\fam\sbfam\xiimsbm}%
\textfont\sbfam=\xiimsbm \scriptfont\sbfam=\ixmsbm
\scriptscriptfont\sbfam=\viimsbm
% sans serif bold
\def\sfb{\fam\sfbfam\xiisfb}%
\textfont\sfbfam=\xiisfb \scriptfont\sfbfam=\ixsfb
\scriptscriptfont\sfbfam=\viisfb
%
\tt \ttglue=.5em plus.25em minus.15em
\normalbaselineskip=14,3pt
\setbox\strutbox=\hbox{\vrule height8.5pt depth3.5pt width\z@}%
\normalbaselines\rm}

%
%
% MITTEL XIV
%
\def\mittel{\def\rm{\fam\z@\xivrm}%
\textfont0=\xivrm \scriptfont0=\xirm \scriptscriptfont0=\viiiirm
\textfont1=\xivmi \scriptfont1=\ximi \scriptscriptfont1=\viiiimi
\def\mit{\fam\@one}
\textfont2=\xivsy \scriptfont2=\xisy \scriptscriptfont2=\viiiisy
\def\cal{\fam\tw@}
\textfont3=\xivex \scriptfont3=\xiex \scriptscriptfont3=\viiiex
% italics
\def\it{\fam\itfam\xivit}%
\textfont\itfam=\xivit
% slanted
\def\sl{\fam\slfam\xivsl}%
\textfont\slfam=\xivsl
% bold
\def\bf{\fam\bffam\xivbf}%
\textfont\bffam=\xivbf \scriptfont\bffam=\xivbf
\scriptscriptfont\bffam=\viiiibf
% typewriter
\def\tt{\fam\ttfam\xivtt}%
\textfont\ttfam=\xivtt
% sans serif
\def\sf{\fam\sffam\xivsf}%
\textfont\sffam=\xivsf \scriptfont\sffam=\xivsf

```

```

        \scriptscriptfont\sffam=\viiiisf
% bold math italics
    \def\mib{\fam\mibfam\xivmib}%
    \textfont\mibfam=\xivmib \scriptfont\mibfam=\ximib
    \scriptscriptfont\mibfam=\viiiimib
% bold math symbols
    \def\bsy{\fam\bsyfam\xivbsy}%
    \textfont\bsyfam=\xivbsy \scriptfont\bsyfam=\xibsy
    \scriptscriptfont\bsyfam=\viiiibsy
% fraktur
    \def\fm{\fam\fmfam\xiveufm}%
    \textfont\fmfam=\xiveufm \scriptfont\fmfam=\xieufm
    \scriptscriptfont\fmfam=\viiiieufm
% bold fraktur
    \def\fb{\fam\fbfam\xiveufb}%
    \textfont\fbfam=\xiveufb \scriptfont\fbfam=\xieufb
    \scriptscriptfont\fbfam=\viiiieufb
% AMS symbols A
    \def\sam{\fam\samfam\xivmsam}%
    \textfont\samfam=\xivmsam \scriptfont\samfam=\ximsam
    \scriptscriptfont\samfam=\viiiisam
% AMS symbols B
    \def\sbm{\fam\sbmfam\xivmsbm}%
    \textfont\sbmfam=\xivmsbm \scriptfont\sbmfam=\ximsbm
    \scriptscriptfont\sbmfam=\viiiimsbm
% sans serif bold
    \def\sfb{\fam\sfbfam\xivsfb}%
    \textfont\sfbfam=\xivsfb \scriptfont\sfbfam=\xisfb
    \scriptscriptfont\sfbfam=\viiiisfb
%
    \tt \ttglue=.5em plus.25em minus.15em
    \normalbaselineskip=17pt
    \setbox\strutbox=\hbox{\vrule height8.5pt depth3.5pt width\z@}%
    \normalbaselines\rm}

%
%
% HYPPERTERTIA XVII
%
\def\hyppertertia{\def\rm{\fam\z@\xviirm}%
    \textfont0=\xviirm \scriptfont0=\xivrm \scriptscriptfont0=\xiirm
    \textfont1=\xviimi \scriptfont1=\xivmi \scriptscriptfont1=\xiimi
\def\mit{\fam\@ne}
    \textfont2=\xviisy \scriptfont2=\xivsy \scriptscriptfont2=\xiisy
\def\cal{\fam\tw@}
    \textfont3=\xviiox \scriptfont3=\xivex \scriptscriptfont3=\xiiox
% italics
    \def\it{\fam\itfam\xviiit}%
    \textfont\itfam=\xviiit
% slanted
    \def\sl{\fam\slfam\xviisl}%
    \textfont\slfam=\xviisl
% bold
    \def\bf{\fam\bffam\xviibf}%
    \textfont\bffam=\xviibf \scriptfont\bffam=\xivbf
    \scriptscriptfont\bffam=\xiibf
% typewriter
    \def\tt{\fam\ttfam\xviitt}%
    \textfont\ttfam=\xviitt
% sans serif
    \def\sffam{\fam\sffam\xviisf}%
    \textfont\sffam=\xviisf \scriptfont\sffam=\xivsf
    \scriptscriptfont\sffam=\xiisf
% bold math italics
    \def\mib{\fam\mibfam\xviimib}%
    \textfont\mibfam=\xviimib \scriptfont\mibfam=\xivmib
    \scriptscriptfont\mibfam=\xiimib

```



```

% bold math symbols
\def\bsy{\fam\bsyfam\xviibsy}%
\textfont\bsyfam=\xviibsy \scriptfont\bsyfam=\xivbsy
\scriptscriptfont\bsyfam=\xiibsy
% fraktur
\def\fm{\fam\fmfam\xviieufm}%
\textfont\fmfam=\xviieufm \scriptfont\fmfam=\xiveufm
\scriptscriptfont\fmfam=\xiieufm
% bold fraktur
\def\fb{\fam\fbfam\xviieufb}%
\textfont\fbfam=\xviieufb \scriptfont\fbfam=\xiveufb
\scriptscriptfont\fbfam=\xiieufb
% AMS symbols A
\def\sam{\fam\samfam\xviimsam}%
\textfont\samfam=\xviimsam \scriptfont\samfam=\xivmsam
\scriptscriptfont\samfam=\xiimsam
% AMS symbols B
\def\sbm{\fam\sbfam\xviimsbm}%
\textfont\sbfam=\xviimsbm \scriptfont\sbfam=\xivmsbm
\scriptscriptfont\sbfam=\xiimsbm
% sans serif bold
\def\sfb{\fam\sfbfam\xviisfb}%
\textfont\sfbfam=\xviisfb \scriptfont\sfbfam=\xivsfb
\scriptscriptfont\sfbfam=\xiisfb
%
\tt \ttglue=.5em plus.25em minus.15em
\normalbaselineskip=21pt
\setbox\strutbox=\hbox{\vrule height8.5pt depth3.5pt width\z@}%
\normalbaselines\rm}

%
%
% CANON XX
%
\def\canon{\def\rm{\fam\z@\xxrm}%
\textfont0=\xxrm \scriptfont0=\xviirm \scriptscriptfont0=\xivrm
\textfont1=\xxmi \scriptfont1=\xviimi \scriptscriptfont1=\xivmi
\def\mit{\fam\@one}
\textfont2=\xxsy \scriptfont2=\xviisy \scriptscriptfont2=\xivsy
\def\cal{\fam\tw@}
\textfont3=\xivex \scriptfont3=\xviex \scriptscriptfont3=\xivex
% italics
\def\it{\fam\itfam\xxit}%
\textfont\itfam=\xxit
% slanted
\def\sl{\fam\slfam\xxsl}%
\textfont\slfam=\xxsl
% bold
\def\bf{\fam\bfam\xxbf}%
\textfont\bfam=\xxbf \scriptfont\bfam=\xviibf
\scriptscriptfont\bfam=\xivbf
% typewriter
\def\tt{\fam\ttfam\xttt}%
\textfont\ttfam=\xttt
% sans serif
\def\sffam{\fam\sffam\xxsf}%
\textfont\sffam=\xxsf \scriptfont\sffam=\xviisf
\scriptscriptfont\sffam=\xivsf
% bold math italics
\def\mib{\fam\mibfam\xxmib}%
\textfont\mibfam=\xxmib \scriptfont\mibfam=\xviimib
\scriptscriptfont\mibfam=\xivmib
% bold math symbols
\def\bsy{\fam\bsyfam\xxbsy}%
\textfont\bsyfam=\xxbsy \scriptfont\bsyfam=\xviibsy
\scriptscriptfont\bsyfam=\xivbsy

```

```

% fraktur
%   \def\fm{\fam\fmfam\xxeufm}%
%   \textfont\fmfam=\xxeufm \scriptfont\fmfam=\xviieufm
%   \scriptscriptfont\fmfam=\xiveufm
% bold fraktur
%   \def\fb{\fam\fbfam\xxeufb}%
%   \textfont\fbfam=\xxeufb \scriptfont\fbfam=\xviieufb
%   \scriptscriptfont\fbfam=\xiveufb
% AMS symbols A
%   \def\sam{\fam\samfam\xxmsam}%
%   \textfont\samfam=\xxmsam \scriptfont\samfam=\xviimsam
%   \scriptscriptfont\samfam=\xivmsam
% AMS symbols B
%   \def\sbm{\fam\sbfam\xxmsbm}%
%   \textfont\sbfam=\xxmsbm \scriptfont\sbfam=\xviimsbm
%   \scriptscriptfont\sbfam=\xivmsbm
% sans serif bold
%   \def\sfb{\fam\sfbfam\xxsfb}%
%   \textfont\sfbfam=\xxsfb \scriptfont\sfbfam=\xviisfb
%   \scriptscriptfont\sfbfam=\xivsfb
%
%   \tt \ttglue=.5em plus.25em minus.15em
%   \normalbaselineskip=27pt
%   \setbox\strutbox=\hbox{\vrule height8.5pt depth3.5pt width\z@}%
%   \normalbaselines\rm}

%
%
% BISCICERO

\def\biscicero{\def\rm{\fam\z@\xxvrm}%
  \textfont0=\xxvrm \scriptfont0=\xxrm \scriptscriptfont0=\xviirm
  \textfont1=\xxvmi \scriptfont1=\xxmi \scriptscriptfont1=\xviimi
\def\mit{\fam\@ne}
  \textfont2=\xxvsy \scriptfont2=\xxsy \scriptscriptfont2=\xviisy
\def\cal{\fam\tw@}
%   \textfont3=\xviieex \scriptfont3=\xviieex \scriptscriptfont3=\xviieex
%
%   \def\it{\fam\itfam\xxvit}%
%   \textfont\itfam=\xxvit
%
%   \def\sl{\fam\slfam\xxvsl}%
%   \textfont\slfam=\xxvsl
%
%   \def\bf{\fam\bffam\xxvbf}%
%   \textfont\bffam=\xxvbf \scriptfont\bffam=\xxbf
%   \scriptscriptfont\bffam=\xviibf
%
%   \def\tt{\fam\ttfam\xxvtt}%
%   \textfont\ttfam=\xxvtt
%
%   \def\sf{\fam\sffam\xxvsf}%
%   \textfont\sffam=\xxvsf \scriptfont\sffam=\xxsf
%   \scriptscriptfont\sffam=\xviisf
%
%   \tt \ttglue=.5em plus.25em minus.15em
%   \normalbaselineskip=32pt
%   \setbox\strutbox=\hbox{\vrule height8.5pt depth3.5pt width\z@}%
%   \normalbaselines\rm}

\newfam\itfam
\newfam\slfam
\newfam\bffam
\newfam\ttfam
\newfam\sffam

```

```
\newfam\mibfam  
\newfam\bsyfam  
\newfam\fmfam  
\newfam\fbfam  
\newfam\samfam  
\newfam\sbfam  
\newfam\sfbfam
```

```
%  
%\catcode'@=12%  
%
```

```
\endinput
```

Literatura

- [1] VLADIMIR ANIĆ, JOSIP SILIĆ: *Pravopisni priručnik hrvatskoga ili srpskoga jezika, Školska knjiga, 1986.*
- [2] DAVID BAUSUM: *T_EX Reference Manual, Kluwer Academic Publishers, 2002.*
- [3] K. BERRY, O. WEBER, T. HOEKWATER: *Kpathsea library, <http://tug.org/kpathsea>*
- [4] YANNIS HARALAMBOUS: *Fonts and Encodings, O'Reilly, 2007.*
- [5] B. JACKOWSKI, J. M. NOWACKI: *Latin Modern Family of Fonts – The Technical Documentation, <http://www.gust.org.pl/projects/e-foundry/latin-modern>*
- [6] DONALD E. KNUTH: *The T_EXbook, Addison Wesley, 1986.*
- [7] F. KÜSTER, R. LEWIS, N. PREINING, R. STUBNER, F. ROUGON: *The Debian T_EX sub-policy, <http://people.debian.org/preining/TeX/Debian-TeX-Policy>*
- [8] MOJCA MIKLAVEC, ARTHUR REUTENAUER: *Putting the Cork back in the bottle – Improving Unicode support in T_EX, <http://tug.org/tex-hyphen>*
- [9] PETR OLŠÁK: *EncT_EX, The Extension of T_EX for Input Re-encoding, <http://www.olsak.net/encTex.html>*

Sažetak

U ovom radu opisana je izrada i korištenje novog $\text{T}_{\text{E}}\text{X}$ formata `hrtex`, koji omogućuje obradu datoteka s hrvatskim znakovima kodiranih po UTF-8 standardu pomoću 8-bitnog $\text{T}_{\text{E}}\text{X}$ alata `pdftex`. Unaprijeđeno je i rastavljanje riječi na kraju retka. Osim postupka izrade i korištenja novog formata, opisane su podloge na kojima počiva rad i koje su poslužile kao literatura, uputa i praktičan primjer pri nastanku ovog rada. Radi se o već poznatim $\text{T}_{\text{E}}\text{X}$ paketima `encT_{\text{E}}\text{X}` i `hyph-utf8` i skupini fontova Latin Modern, a ukratko je opisan i sam program $\text{T}_{\text{E}}\text{X}$ i kodiranje UTF-8, te je dana njegova usporedba s drugim kodiranjima. K tome, priloženi su ispisi datoteka kojima je realiziran novi format.

Summary

In this study the newly created T_EX format `hrtex` is described. Its purpose is to allow the processing of input files containing Croatian characters in the UTF-8 character encoding by the 8-bit T_EX tool, `pdfhtex`. Some improvements concerning word hyphenation have also been made. Beside the descriptions of creating and using the new format, this study also contains descriptions of the bases on which the new format relies, which have been utilized as study material and as practical examples during format creation: T_EX packages `encTEX` and `hyph-utf8`, and Latin Modern collection of fonts. The T_EX program itself has been described in short notes, as well as the UTF-8 encoding, along with a comparison with other encodings. Listings of files used to implement the format are also attached.

Životopis

Rođena sam 9. srpnja 1988. godine u Kopru, R. Slovenija. Od rođenja živim u Buzetu gdje sam završila osnovnu školu i opću gimnaziju. Preddiplomski studij matematike na Prirodoslovno-matematičkom fakultetu – Matematičkom odsjeku upisala sam 2006. godine, a Diplomski studij računarstva i matematike 2009. godine.