

Magyar nyelvű szöveg szedése Magyar \LaTeX -hel

Szabó Péter
<pts@fazekas.hu>

2009. április 5.

Kivonat

Az eredeti \LaTeX csak angol nyelvű szövegek szedésére van felkészítve. A magyar szerzők magyar ékezetes betűket, automatikus szóelválasztást, tipográfiát, irodalomjegyzéket, helyesírás-ellenőrzőt és a tárgymutató szócikkeinek magyar betűrendbe szedését igénylik. Ez a dokumentum bemutatja a Magyar \LaTeX szoftvergyűjteményben megtalálható és az egyéb \LaTeX kiegészítő csomagokat és egyéb segédprogramokat, melyekkel magyar dokumentumok szedése megvalósítható.

1. Telepítés

Előre bocsátjuk, hogy angol nyelvtudás nélkül igen nehéz a \LaTeX -et használni. A \TeX parancsszavak és a külső segédprogramok továbbra is angol nyelvűek, és dokumentáció nagy része sincs magyarra fordítva. A \LaTeX -hel dolgozó szerző magyar nyelvű dokumentumot készít egy angol szoftverkörnyezetben. A nyelvi nehézségek áthidalhatók, ha az olvasó áttanulmányozza a magyar nyelven megjelent, \LaTeX -ről szóló könyvek egyikét (egy lista található a <http://www.math.bme.hu/latex/> címen, és letölthető a [4], a klasszikus bevezető magyar fordítása is), vagy beiratkozik egy \LaTeX -et oktató tanfolyamra vagy egyetemi kurzusra. A továbbiakban feltételezzük, hogy az olvasó alapszinten ismeri a \LaTeX -et, és már használta is, de magyar nyelvű dokumentumok szedésével kapcsolatban még nincs tapasztalata.

A magyar szedéshez szükséges egy működő \TeX disztribúció (UNIX alatt $teTeX$, Windows alatt $MikTeX$ ajánlott), továbbá a Magyar \LaTeX szoftvergyűjtemény, amelynek legfrissebb változata letölthető a fenti webcímről. A továbbiakban feltételezzük, hogy az olvasó a Magyar \LaTeX -et letöltötte, és egy üres könyvtárba kicsomagolta. A bemutatott csomagok és programok többségét elegendő az aktuális (a `.tex` dokumentumot tartalmazó) könyvtárba másolni. Ahol ettől eltérés van, ott külön jelezzük.

Az alábbi lista a \LaTeX magyar használatát segítő szoftvereket tartalmazza, csökkenő fontossági sorrendben. Ha egy szoftver nem része a Magyar \LaTeX -nek, azt külön jelezzük.

- **Babel**: a \LaTeX többnyelvű csomagkészlete. Minden támogatott nyelvhez egy `.ldf` és egy `*hyph*.tex` file-t tartalmaz. Nem szükséges telepítenünk, a legtöbb \TeX disztribúció része. Nem része a Magyar \LaTeX -nek. E dokumentum írásakor a 3.7-es verzió a legelterjedtebb, és csak latin betűs nyelvekre működik gond nélkül.
- **magyar.ldf**: a Babel-ben a magyar tipográfiai szabályok (nagy részét) valósítja meg. Egy régi és hibás verziója része minden \TeX disztribúciónak, helyette a Magyar \LaTeX -ben található, friss változatot (1.5) érdemes használni, a `magyar.ldf` file-t az aktuális könyvtárba másolva. Részletes leírását a 2. szakaszban.
- **huhyp.tex** vagy **huhyp3.tex**: a Babel-ben a magyar szóelválasztási szabályokat tartalmazza. A 3-as verziót Mayer Gyula 1998-ban fejezte be. Bár minden \TeX disztribúció része, alapértelmezésben ki van kapcsolva, ezért aktiválnunk kell, lásd a 2.8. alszakaszban. Az aktiválás kezdők számára nem könnyű feladat, de túl kell esni rajta, mert a szóelválasztás a dokumentumok jó minőségű szedése szempontjából alapvető fontosságú.
- **huhypc.tex**: a szóelválasztást leíró file 4-es verziója, melyet Mayer Gyula fejlesztett 2002-ben. Jóval több összetett szót választ el helyesen, mint az elődje. Az idegen összetett szavakat az összetétel határán választja el (pl. `szin-kron`).
- **huhypf.tex**: a szóelválasztást leíró file 4-es verziója, melyet Mayer Gyula fejlesztett 2002-ben. Az idegen összetett szavakat fonetikusan választja el (pl. `szink-ron`).
- **lafmtgen.pl**: a fenti szóelválasztást megvalósító file-ok telepítését segítő Perl script UNIX-os a $teTeX$ -hez. Használatát lásd a 2.8. alszakaszban.
- **huplain.bst**: magyar irodalomjegyzék szedési szabályait tartalmazó stílusfile $BibTeX$ -hez
- **husort.pl**: a `makeindex` tárgymutató-feldolgozó programot helyettesítő Perl script. Megvalósítja a magyar betűrendbe sorolást. A `makeindex` parancs helyett ezt kell majd futtatnunk. Telepítését és részletes leírását lásd a 4.1. alszakaszban.

- `ccs_extract.pl`: a hosszú kettős mássalhangzókat tartalmazó szavak elválasztását segítő Perl script. Használatát lásd a 2.8. alszakaszban.
- `magyar.xdy`: a Xindy tárgymutató-feldolgozó programhoz magyarítás. Elegendő az aktuális könyvtárba másolnunk. Használatát lásd a 4.3. alszakaszban.
- `CM-Super`: az EC fontok Type 1 formátumú változata, minőségi PDF-ek készítésénél lesz rá szükség. Több, mint 50 MB a mérete, és viszonylag bonyodalmas a telepítése. A CTAN-ról (<http://www.ctan.org/>) tölthető le. Nem része a Magyar \LaTeX -nek.
- `MagyarISpell`: az ISpell UNIX-os helyesírás-ellenőrző program magyar modulja. Telepítésével és használatával kapcsolatban lásd a dokumentációját. Lehetséges interaktív módon is használni, de úgy igen kényelmetlen. Ehelyett érdemes egy arra alkalmas szövegszerkesztőn (*text editor*) belülről meghívni. Ilyen szövegszerkesztők UNIX-ra: Emacs, a Vim új verziói, OpenOffice, Lyx, a KDE szövegszerkesztői. Debian GNU/Linux-on a `apt-get install ihungarian` paranccsal telepíthető, más rendszerekre is szabadon letölthető. Nem része a Magyar \LaTeX -nek.
- `HunSpell`: a MySpell UNIX-os helyesírás-ellenőrző magyarított változata, a MagyarISpell utódja. Képes az ISpell-kompatibilis módban működni. Használata hasonló a MagyarISpell-hez. Szabadon letölthető. Nem része a Magyar \LaTeX -nek.
- `MSpell`: a Morphologic cég ingyenes magyar helyesírás-ellenőrző programja Linux-ra. Az ISpell-et képes helyettesíteni, de interaktív használata igen kényelmetlen. Letölthető a cég honlapjáról, telepítési útmutatást lásd ott. Nem része a Magyar \LaTeX -nek.

`bibtex8` nyolcbites BibTeX, amely be tudja olvasni a `csf` fájlokat. a `huplain.bst` mellett nincs rá szükség. Nem része a Magyar \LaTeX -nek, de a honlapról letölthető. A CTAN-ról (<http://www.ctan.org/>) is letölthető, a `biblio/bibtex/8-bit` könyvtárból.

2. Kezdőlépések, a magyar.ldf

2.1. A Babel betöltése

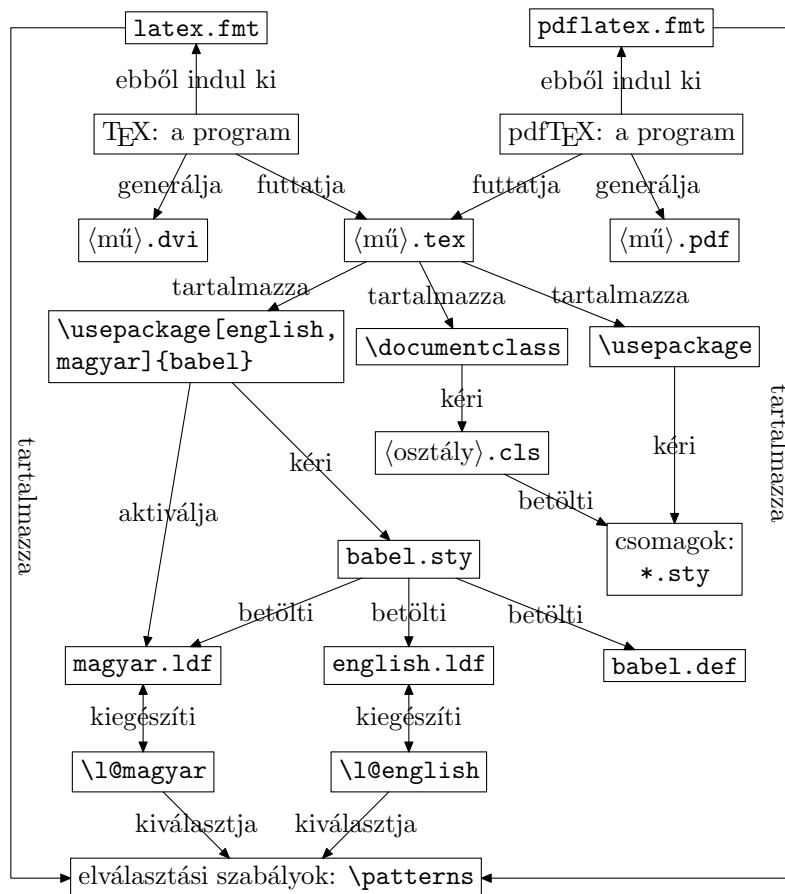
Egy egyszerű, angol nyelvű \LaTeX dokumentum (`.tex` file) forrása és lefordított képe:

<pre> 1 \documentclass{article} 2 \begin{document} 3 \section{Welcome} 4 Hello, World!\ 5 Current language is: \languagenam 6 \end{document} </pre>	<pre> 1 Welcome Hello, World! Current language is: nohyphenation 1 </pre>
---	--

A magyar nyelv használatához be kell töltenünk a Babel csomagot, és meg kell adnunk a használni kívánt nyelveket. Az alapértelmezett nyelvet adjuk meg utoljára. A `magyar.ldf` 1.5-ös verziójának működését a `\magyarOptions` makró definiálásával módosíthatjuk. Például:

<pre> 1 \documentclass{article} 2 \usepackage[T1]{fontenc} 3 % ^^^ vagy \usepackage{t1enc} 4 \usepackage[latin2]{inputenc} 5 \def\magyarOptions{defaults=prettiest} 6 \usepackage[french,magyar]{babel} 7 % ^^^ magyar az alapnyelv 8 \begin{document} 9 {\selectlanguage{french} \section{Francais}} 10 \section{Üdvözlő} 11 Szia, {\languagenam} Világ!\ 12 \lowercase{Árvíztűrő tükörfúrógép}\ 13 \uppercase{\'Arv\' izt\H{u}r\H{o}} 14 t"uk"orf\'ur\'og\'ep} 15 \end{document} </pre>	<pre> 1 Francais 2. Üdvözlő Szia, magyar Világ! árvíztűrő tükörfúrógép ÁRVÍZTŰRŐ TÜKÖRFÚRÓGÉP 1 </pre>
---	--

A példában a `fontenc.sty` csomagot a T1 kódolással azért kellett betöltenünk, hogy a \TeX az ékezetes betűket tartalmazó szavakat is el tudja választani. Az `inputenc.sty` csomagra pedig azért volt szükség, hogy a `.tex` file-ba ékezetes betűket is írassunk. Nem baj, ha a szövegszerkesztőnkben (*text editor*) kalapos ú és hullámvonalas ő betű jelenik meg, a lefordított dokumentumban jók lesznek az ékezetek – ezt a `latin2` bemeneti kódolás



1. ábra. A Babel és a különböző nyelvek betöltése

garantálja. Az Unicode elterjedésével egyre több szerző szeretné UTF-8 kódolásban megírni a `.tex` forrást. Ezt megteheti az `ucs` csomaggal, amely a CTAN-ról letölthető.

A fenti példában figyeljük meg, hogy a `magyar.ldf` a fejezetszám után pontot tesz, míg a francia (és az alapértelmezett angol) tipográfiában nincs pont. A `magyar.ldf` számos egyéb ponton is a magyar tipográfiát követi, de nem lehet mindent automatizálni, ezért ajánlott, hogy a dokumentum szerzője tisztában a magyar tipográfiai szabályokkal.

A nyelvek között `\selectlanguage` parancssal válthatunk, mint ahogy a példa mutatja. Azt is láthatjuk a példából, hogy a csoport (a dokumentum `{ és }` közé csoportosított része) végén visszaáll a csoport megnyitása előtti nyelv.

A Babel és az `.ldf` file-ok betöltését a \LaTeX környezetben az 1. ábra szemlélteti.

2.2. Ékezetes betűk

Bármilyen latin betűre bármilyen ékezetet elhelyezhetünk az az 1. táblázatban látható parancsokkal. Figyeljünk arra, hogy ha magát a betűt is `\`-sel kezdődő parancs hozza elő, akkor az `\uppercase` és `\lowercase` nem fog változtatni rajta; helyettük inkább a `\MakeUppercase` és `\MakeLowercase` parancsok ajánlottak. Az ékezetek többsége leveszi az `i`-ről a pontot (pl. `\'i`), de amelyik nem, ott `i` helyett `\i`-t kell gépelnünk, `j` helyett pedig `\j`-t – ezeket a táblázat külön jelzi. Célszerűbb szóközök helyett kapcsos zárójeleket használni (a táblázatban is így van), mert ezáltal biztosan egymás mellett marad a betű és az ékezet, akkor is, ha pl. szövegszerkesztőnkben újratördeljük a bekezdést.

Az előző példadokumentum az összes magyar ékezetes betűt tartalmazza. Ezeket az 1. táblázat parancsaival is begépelhetjük, de ha `inputenc.sty`-t használunk, akkor jóval kényelmesebb a parancsok helyett csak ennyit gépelni: *Árvíztűrő tükörfúrógép*. Bárhogy is gépeljük be a betűket, a `fontenc.sty`-t T1 kódolással használva a \TeX automatikusan el tudja választani a magyar szavakat (kis- és nagybetűk közötti különbség nem számít). Ha azonban a szó idegen ékezetes betűt is tartalmaz (pl. `û, õ, ä`), a \TeX nem fogja magától elválasztani, és szükség esetén nekünk kell `\-` parancsokat elhelyeznünk a töréspontokban, például `ähn\ -lich` vagy `gar\ -\c{c}on`, vagy rábízhathatjuk az elválasztást a Babel-re a megfelelő nyelv kiválasztásával: `\selectlanguage{french} gar\c{c}on`.

ì	\`i	í	\'i	î	\^i	ï	\~{i}
ī	\={i}	ı	\.i	ï	\"i	ı̇	\c{i}
ÿ	\u{i}	ÿ	\v{i}	ÿ	\H{i}	ı̈	\d{i}
ı̇	\b{i}	ı̈	\t{i}	ö	\t{oo}	ı̇	\r{i}
œ	\oe{}	Œ	\OE{}	æ	\ae{}	Æ	\AE{}
å	\aa{}	Å	\AA{}	ß	\ss{}		
noalign ø \o{}							
		Ø	\O{}	ı	\l{}	Ł	\L{}
ı	\i{}	ı	\j{}	ı	!'	ı	?'
ı	\textexclamdown			ı	\textquestiondown		
þ	\th{}	Þ	\TH{}	ı	\k{i}		
đ	\dh{}	Đ	\DH{}	đ	\dj{}	Đ	\DJ{}
ŋ	\ng{}	Ŋ	\NG{}			SS	\SS{}

Az alsó csoport csak `\usepackage{t1enc}` mellett működik.

1. táblázat. A \LaTeX ékezetek és különleges betűk

2.3. Betöltési opciók használata

```
\def\magyarOptions {<kulcs1>=<érték1>,<kulcs2>=<érték2>,...}
\PassOptionsToPackage{<kulcs1>=<érték1>,<kulcs2>=<érték2>,...}{magyar.ldf}
```

Az 1.5-ös verziótól kezdve a `magyar.ldf`-et használó szerzők a csomag működését testreszabhatják. Ehhez a preambulumban a `\usepackage[...]{babel}` előtt ki kell adni a `\def\magyarOptions{...}` parancsot vagy a `\PassOptionsToPackage` parancsot, a fenti módon. Ha túl későn adjuk ki őket, a `magyar.ldf` figyelmeztet erre a konzolon, de az opciókat már nem tudja figyelembe venni. Az kulcs-érték párok, vagyis az opciók megadási sorrendje lényeges. Ha több `\def\magyarOptions` szerepel a dokumentumban, akkor csak a legutolsó (de még a `\usepackage[...]{babel}` előtti) számít. Ha több `\PassOptionsToPackage` szerepel, akkor mindegyik számít, futási sorrendben (ami legtöbbször megegyezik a file-béli sorrenddel). Az egy parancson belüli opciólistában a sorrend balról jobbra halad.

Ha egy opció többször szerepel, akkor a sorrend szerinti legutolsó értéke lép érvénybe. Van egy-két olyan opció (pl. `defaults=`), amely más opciók értékét módosítja.

Az alszakasz további részét az első olvasáskor ajánlott átugrani.

A `magyar.ldf` az alábbi alapbeállításokat kínálja:

defaults=over-1.4. Ez az alapértelmezés. Az a `magyar.ldf` 1.4-es verziójának tipográfiájától csak kevésbé, néhány, de fontos ponton tér el. Az új, más csomagokkal össze nem akadó parancsokat elérhetővé teszi, de nem kapcsolja be. Kijavítja az előző verzió legfontosabb megvalósítási hibáit. Célja, hogy az összes, az 1.4-es verzióhoz készült dokumentum leforduljon vele, és csak kevés, a sor- és oldalhatárokat általában nem érintő látható eltérés legyen az 1.4-es verzióhoz képest. Ha ehelyett a szerző célja az, hogy a dokumentum a magyar tipográfiai szabályokat minél inkább kövesse, akkor helyette a `defaults=hu-min` ajánlott.

defaults=compat-1.4. Nem változtat (lehetőség szerint) a `magyar.ldf` 1.4-es verziójának tipográfiáján, de kijavít számos megvalósítási hibát és más \LaTeX csomagokkal való összeférhetetlenséget. Ezt ajánljuk azoknak, akik egy régi dokumentumot szeretnének vizuális változás nélkül az új `magyar.ldf`-fel újrafordítani. A teljes egyezés nem garantált, a kompatibilitás a `magyar.ldf`-nek csak az opciók által befolyásolt részeire terjed ki. Elrejti az 1.4-es verzió után definiált parancsokat is (pl. `\told`, `\emitdate`).

defaults=safest. Szinte mindent kikapcsol, hatása olyan, mintha a `magyar.ldf`-et be se töltöttük volna. Hibakeresésnél érdemes ezt választani. Tehát ha váratlan hibaiüzenetet kapunk, vagy ha arra gyanakszunk, hogy a `magyar.ldf` összeakad valamely másik csomaggal, akkor váltsunk át `defaults=safest-re`, és ha a probléma továbbra is fennáll, akkor nem a `magyar.ldf`-fel volt baj.

defaults=prettiest. Minden új szolgáltatást bekapcsol, a magyar tipográfiához a lehető legjobban illeszkedik, bár még nem felel meg neki teljesen. A `magyar.ldf` fejlesztése során keletkezendő új szolgáltatásokat ez az alapbeállítás aktiválni fogja. Vigyázat! Az itt bekapcsolt szolgáltatások esetleg összeakadhatnak más csomagokkal.

defaults=hu-min. Az alapbeállításoktól annyiban tér el, hogy nem az 1.4-es verzióval való kompatibilitásra figyel, hanem az opciókat úgy állítja be, hogy azok ne mondjanak ellent a magyar tipográfiai alapelveknek. (Ennek ellenére előfordulhat, hogy a keletkező dokumentum egy olyan ponton eltér az alapelvektől, melyet nem szabályoz opció.) Ha a szabályok választást tesznek lehetővé, akkor ez az alapbeállítás a legegyszerűbben megvalósítható és legkevesebb kompatibilitási problémát előhozó értékek közül

válogat. Ha a `magyar.ldf` nem felelt meg egy szabálynak, de egy későbbi fejlesztése után már megfelel, akkor a változást ez az alapbeállítás aktiválni fogja. A magyar tipográfiát érintő opciók és jelenleg beállított értékeik a következők: `activespace=safe`, `amstocnumskip=\enskip`, `frenchspacing=yes`, `captions=hu`, `classmod=yes`, `figurecaptions=hu`, `footnote=huplain`, `frenchspacing=yes`, `hyphenmins=22`, `labelenums=hu-a`, `labelitems=hu`, `longcaption=centered`, `mathhucomma=fix`, `mathbrk=fix`, `postpara=diamond`, `postsubpara=star`, `postdescription=dot`, `sectiondot=safe`, `tablecaptions=hu`, `theoremtitle=hu`, `titles=\enskip`.

Az alábbiakban ismertetésre kerülnek a `magyar.ldf` betöltési opciói. Minden opció után az az érték található, amelyet az opció akkor kap, ha öt `defaults=over-1.4` alapértelmezett alapbeállítás mellett kihagyjuk a `\def\magyarOptions{...}`-ből. Az opció jelentésének ismertetése után az alapérték és az egyéb lehetséges értékek hatásainak részletes ismertetése következik. Az értékeket az opció nevével (kulcs) `=`-lel kell elválasztani, ezért az értékek a magyarázatban `=`-lel kezdődnek. Az opciók többségénél csak néhány előre megadott értékből lehet választani; az ettől való eltérést mindig jelezzük.

Vannak opciók, melyek egy-egy parancs definiálását szabályozzák. Ha értékük `=yes`, akkor az adott parancsot a `magyar.ldf` betöltésekor (újra)definiálja, `=no` érték esetén nem, `=weak` érték esetén pedig csak akkor definiálja a parancsot, ha az a `magyar.ldf` betöltésekor még nem létezett.

Következzen tehát az összes opció leírása.

defaults=over-1.4. Az alapbeállításokat jelöli ki. Célszerű a többi opció előtt megadni. További értékei: `=compact-1.4`, `=safest`, `=prettiest`. Részletes leírásukat lásd fent.

safest=no. Kerülendő. `=no` értékének nincs hatása, `=yes` értéke pedig `defaults=safest`-tel egyezik meg.

accents=defaulthigh. Alapértékének nincs hatása, `=low` értéke a `babel.def`-ben definiált `\umlautlow` makrót használva az „ö” és más betűk dupla pontos ékezetét viszi lejjebb, a betűhöz közelebb. `=high` értéke pedig némileg feljebb viszi az ékezetet. A három pozíció, `=low`, `=defaulthigh`, `=high` sorrendben: ö, ö, ö. Bár a magyar tipográfia a `=low`-t javasolja, ennek csak OT1 fontkódolás mellett van hatása, továbbá a magyar szavak automatikus elválasztása T1 fontkódolást követel meg, ezért inkább tartózkodjunk ennek az opciónak a használatától.

active=safe. Tegyen-e a `magyar.ldf` aktívva egy karaktert? Az aktív karakterekről a 2.6. alszakaszban olvashatunk részletesen. `=none` értéke nem tesz aktívva karaktert. `=onlycs` sem aktivál karaktert, de bevezeti a `shorthandcs=` opcióban megadott nevű parancsot, amely ugyanazt nyújtja (hosszabb gépelés árán), mint egy aktívva tett karakter. `=safe` aktívva teszi a `activeprefix=` opcióban megadott karaktert, és a `shorthandcs=` opcióban megadott parancsot is. `=problematic` hatása megegyezik `=safe`-ével, de ezt máshogy, egyéb csomagokkal valószínűleg összeakadva éri el (a `magyar.ldf` 1.4-es verziója működött így). A `=safe` kezdetben (`\begin{document}` előtt) és nem magyar nyelvű szövegrész esetén, kompatibilitási okokból a karakter aktívságát megszünteti. Ez persze hiábavaló, ha betöltjük például a `frenchb.ldf`-et, amely a fordított aposztrófot végig aktívna hagyja. Probléma esetén érdemes a Babel-t minél később betölteni.

activeprefix=babelopt2. Kiválasztja, hogy melyik karakter legyen aktív. `=none` értéke egyetlen karaktert sem tesz aktívva, `=grave` a fordított aposztrófot (`'`), `=acute` a sima aposztrófot (`'`), `=quotedbl` az írógép idézőjel (`"`) aktiválja. `=babelopt2` az `'`-et teszi aktívva, ha a Babel `\usepackage[activeacute,...]{babel}` paranccsal lett betöltve, egyébként pedig `'`-et. `=babelopt3` az `'`-et aktiválja, ha a Babel `\usepackage[activeacute,...]{babel}` paranccsal lett betöltve, a `'`-et aktiválja `\usepackage[activegrave,...]{babel}` esetén, egyébként pedig az `"`-et. Az alapértelmezés megegyezik az 1.4-es `magyar.ldf` viselkedésével, amely mindig a `'`-ot aktiválta. Az aktív karakterek használatát részletesen tárgyalja a 2.6. alszakasz.

activespace=none. Legyen-e egy rövid szóköz kihagyva minden kettőspont, pontosvessző, kérdőjel és felkiáltójel előtt? A magyar tipográfia megköveteli ezt a kihagyást, melyet a `=safe` érték meg is valósít a fenti négy karakter aktiválásával. Az alapértelmezett `=none` érték nem hagy ki helyet, mivel az aktiválás előre nem látható kompatibilitási problémákhoz vezethet. Elérhető még a `=problematic` érték, amely a `=safe`-et valósítja meg olyan összeférhetlenségi problémákkal, ahogy a a `magyar.ldf` 1.4-es verziója tette (volna).

afterindent=unchanged. Kell-e szakasz cím után az első sort beljebb kezdeni? Az alapérték ezt a dokumentumosztályra bízza, a `=force=yes` érték mindenképpen beljebb kezd (`\indent`), a `=force=no` pedig a margón kezd (`\noindent`). A magyar tipográfia mindkét fajta kezdést megengedi.

amslevelfix=fix-ams. Megjavítsuk-e az ábrák jegyzékét (`\listoffigures`) és egyéb jegyzékeket AMS dokumentumosztályok (pl. `\documentclass{amsart}`) esetén, hogy ne legyen „Overfull \hbox...” hibaüzenet. `=fix-ams` csak akkor javítja meg, ha egy AMS dokumentumosztály van betöltve, `=fix-all` mindenképpen megkísérli a javítást, `=unchanged` nem javít.

Az opció nem kötődik szorosan a magyar nyelvhez.

amspostsectiondot=no. Bizonyos AMS dokumentumosztályok (`amsproc.cls`, `amsbook.cls` és `amsart.cls`) ponttal fejezik be a strukturális egységek címsorait, amit a magyar tipográfia megtilt. Ha az opció értéke nem `unchanged=`, akkor eltünteti ezt a pontot.

amstocnumskip=\enskip. Előírja (a `\tocsection` és hasonló parancsok átdefiniálásával), hogy AMS dokumentumosztályok magyar nyelvű bejegyzéseinek számaikat a tartalomjegyzékben mekkora kihagyás válassza el a fejezet nevétől. Tetszőleges nem üres \LaTeX kód megadható. A magyar tipográfia az `\enskip`-et javasolja az AMS által használt `\quad` helyett. Ha üres értéket adunk meg, akkor nem nyúl a `\tocsection`-höz.

amstocnumlang=all. Alapértéke beállítja, hogy az `amstocnumskip=` opció a nem magyar nyelvű fejezetek tartalomjegyzék-beli címsoraira is vonatkozzon. A `=hu` érték csak a magyar nyelvűekre alkalmazza az `amstocnumskip=` opciót.

az=weak. A határozott névelőt (*a/az*) automatikusan kirakni képes `\az` parancsot és az őt használó `\aref` és egyéb parancsokat definiálja, a részleteket lásd a 2.11. alszakaszban. Egyéb értékei: `=no` és `=yes`.

captions=hu. Ha értéke nem `=unchanged`, akkor a `magyar.ldf` a járulékos fejezetneveket és egyéb feliratokat magyarul jeleníti meg. Például a `\bibname` makrót „Bibliography”-ról átírja „Irodalomjegyzék”-re, és a képaláírásokban sem „figure” jelenik majd meg, hanem „ábra”.

chapternumber=unchanged. Ha értéke `=huordinal`, akkor a fejezetszámok magyarul betűzve jelennek meg, például „harmadik fejezet”. Ha az érték `=Huordinal`, akkor nagybetűvel kezdődik, például „Negyedik fejezet”. Az alapérték nem változtat a számozáson, ami a \LaTeX osztályok többségében arab számokkal történik. Lásd még 2.12. alszakasz.

classmod=yes. A `magyar.ldf` ismer néhány \LaTeX osztályt (ezek: a `book.cls`, a `report.cls`, az `article.cls`, a `letter.cls`), melyek viselkedését képes módosítani. Az ismert osztályokban a `\part` és `\chapter` parancsok, a tartalomjegyzékek és a fejlécek tipográfiáját módosítja, ha `classmod=yes`-t adunk meg. `classmod=unchanged` esetén módosítás nem történik. Bár ismeri a `letter.cls`-t is, a levél fejlécét és egyéb formai elemeit nem a magyar szabályok szerint helyezi el, csak a címsort módosítja. A fejléceket – technikai okokból – nem módosítja `toclang=unchanged` opció esetén.

dottedtocline=fix. Alapértéke a `\@dottedtocline` parancs átdefiniálásával engedélyezi, hogy a tartalomjegyzékek pontozott sorainak végén az oldalszám balra túllógjon a számára fenntartott `\@pnumwidth` távolságon. Az `=unchanged` érték nem változtat a `\@dottedtocline` alap \LaTeX -parancson. Az opció nem kötődik szorosan a magyar nyelvhez.

emitdate=weak. A dátumok magyar helyesírás szerinti megjelenítését végző, több formából választani engedő és a ragozást is támogató `\emitdate` parancsot definiálja, a részleteket lásd a 2.16. alszakaszban. Egyéb értékei: `=no` és `=yes`. Lásd még a `hutoday=` opciót.

extras=yes. A magyar nyelv aktívvá válásakor a Babel számos beállítást módosít az `\extrasmagyar` makró futtatásával (és amikor egy másik nyelv lesz aktív, akkor előbb a `\noextrasmagyar` makró fut le). A `magyar.ldf` betöltődésekor a fenti két makróba építi bele saját nyelvspecifikus beállításait. Az opció `=yes` alapértéke engedi érvényre jutni ezeket a beállításokat, a `=no` érték pedig letiltja őket. Az `extras=no` használata egyedül hibakeresési céllal javasolt, mert épp a `magyar.ldf` fő célját hiúsítja meg. Hibakeresés esetén érdemes a `extras=no`-t is tartalmazó `defaults=safest` alapbeállításokból kiindulni.

figurecaptions=hu.

footnote=yes. A lábjegyzetek megjelenítési stílusát szabályozza. `=yes` értéke elérhetővé teszi a `\footnotestyle` parancsot, mellyel a 2.18. alszakaszban leírt módon testreszabható a lábjegyzetek megjelenítése. `=huplain` értéke ezen felül még a magyar tipográfiának megfelelő, arab számokat használó stílust állít be, `=starplain` értéke pedig csillagokat használó stílust állít be, ami szintén megfelel a magyar tipográfiának. Az `=unchanged` érték nem változtat a lábjegyzeteken, és a `\footnotestyle` parancsot sem definiálja.

frenchspacing=unchanged. `=yes` értéke az írásjelek utáni szóköz szélességét a magyar tipográfiának megfelelően egyenletesre állítja (`\frenchspacing`), amíg a magyar nyelv aktív. `=no` értéke a magyar nyelv aktívvá válásakor az írásjeltől függő változó szóközszélességet ír elő. A `=unchanged` érték nem változtat a beállításokon. További leírását lásd még a 2.4. alszakaszban.

hang=weak. A szimbólummal kezdődő, függő bekezdések nyitására szolgáló `\hang` parancsot definiálja, a részleteket lásd a 2.5. alszakaszban. Egyéb értékei: `=no` és `=yes`.

hunnewlabel=yes. A generált oldalszámok és fejezetszámok névelővel ellátását segítő `\hunnewlabel` segédmakrót definiálja, a részleteket lásd a 2.11. alszakaszban. `=no` értéke nem generál `\hunnewlabel`-eket, `=only-hu` értéke pedig csak akkor generál, ha a `\label` parancs hívásakor a magyar nyelv aktív.

hunumbers=yes. A nem túl nagy egész számokat magyarul kibetűzni képes `\@hunumeral` és egyéb parancsokat definiálja, a részleteket lásd a 2.12. alszakaszban. Egyéb értéke: `=no`.

hutoday=yes. Ha értéke =yes, akkor a dátumok magyar megjelenítéséért felelős parancsokat definiálja, =no érték esetén pedig nem módosít az alapbeállításokon. A szóban forgó parancsok: \today, ami a \make-title hatására megjelenik; \ondatemagyar és \ontoday, melyek az -án/-én raggal ellátott \today-nek felelnek meg, de a Babel sehol nem használja őket; és végül \SafeToday, amely a L^AT_EX kernelnek a Babel által felüldefiniált \today parancsával egyenértékű. További leírás olvasható a 2.16. alszakaszban. A magyar nyelvű dátummegjelenítésnek egy gazdagabb, ragozást is megengedő formája az \emitdate parancs, melyet az emitdate= opcióval lehet engedélyezni.

hyphenation=huhyp3. Előírja, hogy a T_EX milyen szabálykészlet szerint válasza el a magyar szavakat. Az opció =huhyp3 értéke az 1998 óta klasszikusnak számító 3-as verziót használja, amely kevés összetett szót tud helyesen elválasztani, =composite értéke a huhyp.tex 2002-ben készült 4-es verzióját (vagy újabbat) használja, amely már az összetett szavak többségét helyesen választja el, az idegen szóösszetétel-határokat is követi (pl. **szin-kron**) és az MHSz§231 és 238 szabványok szerint készült. A =phonetic érték az MHSz§224-et követő, az idegen szóösszetételeket (pl. **szink-ron**) figyelmen kívül hagyó 4-es verziót (vagy újabbat) használja. A =none érték letiltja az automatikus szóelválasztást. A hyphenation= opció csak a magyar szavak elválasztására van hatással. Az elválasztásról részletesen ír a 2.8. alszakasz.

hyphenmins=22. Ha értéke két számjegy, akkor a bal illetve jobb oldali számjegy azt adja meg, hogy a magyar nyelvű szavak bal illetve jobb oldalán legalább hány karakter közé nem szabad automatikusan elválasztójelet tenni. A Babel és a L^AT_EX alapbeállítása 23, a magyar.ldf-é pedig 22, tehát például „Aliz” a bal oldala miatt nem választható el, „Sára” viszont igen, mivel mindkét oldalon megvan a két karakter. (A „fáraó” szó a jobb oldala miatt csak egy helyen választható el, az „évei” szó egyik elválasztási pontja a bal, a másik pedig a jobb széléhez van túl közel.) Az elválasztásról további részletek olvashatók a 2.8. alszakaszban. A hyphenmins= a Babel-en keresztül nyelvváltáskor a beépített T_EX \lefthyphenmin és \righthyphenmin számlálókat módosítja. Ezekhez nem érdemes kézzel hozzányúlni, mert nyelvváltáskor a visszaíródik a nyelvre jellemző érték.

labelenums=unchanged.

labelitems=unchanged.

longcaption=justified. !! kettőspontot is ponttá

mathbrk=define. A =define alapérték a matematikai szimbólumok sortörését támogató \MathBrk parancsot definiálja, a =fix érték ezen felül még a magyar tipográfiai szokásokhoz igazítja néhány matematikai bináris operátor és bináris reláció sortöréssel kapcsolatos viselkedését. Az =unchanged érték nem definiál és nem módosít semmit. A részleteket lásd a 2.10. alszakaszban. A =fix érték együttműködik a nath.sty és az amsmath.sty csomagokkal, de kockázatos használni, mert esetleg más csomagokkal összeakadhat.

mathhucomma=define. !! both places

mathfactorial=define. A =define alapérték a matematikai faktoriálisjelet helyes térközzel beszuró \factorial parancsot definiálja, parancsot definiálja, a =fix érték ezen felül még átírja a felkiáltójel képleten belüli jelentését, \factorial-ra. Az =unchanged érték nem definiál és nem módosít semmit. A részleteket lásd a 2.10. alszakaszban. A nath.sty eme opciótól függetlenül helyes térközzel helyezi el a faktoriálisjelet. A =fix érték együttműködik a nath.sty és az amsmath.sty csomagokkal, de kockázatos használni, mert esetleg más csomagokkal összeakadhat.

Ha az activespace= és a mathfactorial= opciókat együtt használjuk, akkor a felkiáltójelre matematikai képletben a mathfactorial=, szövegben pedig az activespace= van hatással.

Az opció nem kötődik szorosan a magyar nyelvhez.

mathmuskips=unchanged. =hu értéke a matematikai képletekben levő térközöket igazítja a magyar tipográfiához, a =latex érték a L^AT_EX alapbeállításait erősíti meg, míg az =unchanged érték nem változtat a térközök beállításán. A részleteket lásd a 2.10. alszakaszban.

mathreal=weak. A képletekben a tizedespontot tizedes vesszővel helyettesítő \MathReal parancsot definiálja, a részleteket lásd a 2.10. alszakaszban. Egyéb értékei: =no és =yes.

mond=weak. A párbeszédekben új szereplő megszólalását bevezető \mond parancsot definiálja, a részleteket lásd a 2.5. alszakaszban. Egyéb értékei: =no és =yes.

openqq=maybedown. Azt adja meg, hogy a ‘ ‘ és ‘ ’ parancsok hogyan viselkedjenek. (Ha az activeprefix= más karaktert jelölt ki aktívnek, akkor a fenti két parancs első karaktere értelemszerűen megváltozik). Az =unchanged érték nem kezeli parancsként a fenti karaktersorozatokat, tehát ‘ ‘ valószínűleg egy angol dupla nyitó felső idézőjelet fog eredményezni (“) – feltéve, hogy a beírásakor aktuális fontban van ilyen ligatúra; a ‘ ’ pedig valószínűleg két angol szimpla idézőjelet (‘) jelenít meg. Az =up érték ‘ ‘-re angol dupla nyitó felső idézőjelet (“), ‘ ’-re pedig ennek alsó változatát (,) adja, a =down érték ennek épp fordítottja: ‘ ‘-re (,) és ‘ ’-re (“). Az opció hatása természetesen csak akkor érvényesül, ha a magyar nyelv aktív.

A =maybedown alapérték active=none esetén =unchanged-del, egyéb esetben =down-nal ekvivalens. Ennek első látásra talán meghökkenítő mellékhatása, hogy a forrásban ‘ ‘-nek gépelt jel alsó idézőjelként jelenik

meg. Ez az 1.4-es magyar.ldf-fel való kompatibilitás miatt van így, továbbá azért, mert pl. a CM fontokban nincs ligatúra a magyar nyitó idézőjelnek („), de így ligatúra nélkül is begépelhetők ‘-ként.

Új dokumentum írásakor az javasolható, hogy a `openqq=` alapbeállításon ne változtassunk, és az idézőjeleket a `\textqq` paranccsal helyezzük el (lásd a 2.5. alszakaszban).

partnumber=unchanged. Hasonló a `chapternumber=-`höz, de a részek, kötetek (`\part`) számozására vonatkozik, például „Ötödik rész”.

postdescription=unchanged.

postpara=unchanged.

postsubpara=unchanged.

refstruc=weak. A dokumentum strukturális egységeire hivatkozó `\refstruc` parancsot definiálja, a részleteket lásd a 2.14. alszakaszban. Egyéb értékei: `=no` és `=yes`.

sectiondot=safe. !! Bizonyos AMS dokumentumosztályok (`amsproc.cls`, `amsbook.cls` és `amsart.cls`) ponttal fejezik be a strukturális egységek címsorait, amit a magyar tipográfia megtilt. Ha az opció értéke nem `none=`, akkor eltünteti ezt a pontot is.

shorthandcs=\shu. Annak a parancsnak a nevét írja elő, amely az aktív karakterrel (lásd a 2.6. alszakaszban) egyenértékű. Ha értéke `=none`, akkor nem lesz ilyen parancs. Ettől eltérő csak akkor adható meg, ha nem `active=none` van beállítva. Ha egy parancs van megadva (ajánlott), pl. `shorthandcs=\shu`, akkor az eldobja az első paraméterét, így például a `\shu&` és `'|` ugyanazt fogja eredményezni (feltéve, hogy `'` az aktív karakter). Ha viszont a `shorthandcs=` értéke egy backslash nélküli név, akkor nincs ilyen eldobás: tehát például `shorthandcs=susu` esetén a `\susu|` és `'|` jelenti majd ugyanazt.

A kétfajta parancs közül az első paraméterét eldobó ajánlott, mert ezáltal se a `po'ttyan`, se a `po\shu'¬ttyan` szóba nem kell szóközt írni, míg a `po\susu_ttyan`-ba kéne.

shortrefcmds=yes. `=yes` értéke a magyar.ldf 1.4-es verziójával való kompatibilitás miatt alternatív, háromkarakteres neveket definiál a névelővel hivatkozó parancsoknak: `\aref` → `\azr`, `\Aref` → `\Azr`, `\apageref` → `\azp`, `\Apageref` → `\Azp`, `\acite` → `\azc`, `\Acite` → `\Azc`. A `=no` érték e hárombetűs parancsokat nem definiálja. A határozott névelőről szóló 2.11. alszakaszban további részletek olvashatók.

suggestions=yes.

tablecaptions=hu.

textqq=weak. Az idézőjeleket (macskakörmöket és egyéb jeleket) generáló `\textqq` parancsot definiálja, a részleteket lásd a 2.5. alszakaszban. Egyéb értékei: `=no` és `=yes`.

theoremtitle=hu. A tételszerű környezetek címsorainak a formátumát határozza meg. Az `=unchanged` érték nem változtat az alapbeállításokon, tehát a dokumentumba valószínűleg az angol tipográfia szerinti „Theorem 1.2” kerül. A `=hu` érték megcseréli a sorrendet és elhelyezi a pontokat: „1.2. tétel.” fog megjelenni. A részleteket lásd 2.15. alszakaszban.

titles=\enskip.

toclang=good.

told=weak. Az egész számok toldalékolását végző `\told` parancsot definiálja, a részleteket lásd a 2.13. alszakaszban. Egyéb értékei: `=no` és `=yes`.

varioref=hu.

A különböző nyelvek .ldf file-jai közül alig van olyan, ami hosszabb 14kB-nál. A magyar.ldf legújabb, 1.5-ös verziója túllépi a 120kB-ot is. Ez a kód azonban nem töltődik be teljes egészében: a szerző az opciók megadásával megválaszthatja, hogy mely szolgáltatásokat kíván igénybe venni, és azokat milyen beállításokkal kívánja használni. A kihagyott szolgáltatások nem lassítják a L^AT_EX-et, és a memóriaigényt sem növelik.

A Babel által kínált `\languageoptions` parancsot a magyar.ldf nem használja, mert ennek hatása túl későn érvényesül, és így segítségével nem lehetne bizonyos kódrészleteket kihagyni.

2.4. `\frenchspacing` – a pont utáni szóköz szélessége

Az angolszász tipográfia bizonyos karakterek után a normálisnál szélesebb szóközt ír elő. Ezek a karakterek: `. ? !` (háromszoros szóközzel), `:` (kétszeres), `;` (másfélszeres) és `,` (1.25-szoros). A magyar hagyományok viszont egyenletes szóközszélességet javasolnak, melyet a L^AT_EX `\frenchspacing` parancsával aktiválhatunk. A `\nonfrenchspacing` parancs (vagy a csoport vége) visszaállítja a változó szóközszélességet. A magyar.ldf `frenchspacing=` betöltési opciójában megadható, hogy magyar nyelvre váltásnál hogyan változzon ez a tulajdonság. Az `=unchanged` nem változtat semmit, a `=yes` érték `\frenchspacing`-et állít be a magyar nyelvre, a `=no` pedig `\nonfrenchspacing`-et. Az opció által létrehozott változás csak addig érvényes, amíg a magyar nyelv aktív.

2.5. Kiemelések, idézetek, megszólalások

<code>\textqq{<szövegközi idézet>}</code>	∈ magyar.ldf, <code>textqq=yes</code>
<code>\emph{<bekezdésen belüli kiemelés>}</code>	
<code>{\em <több bekezdésen át tartó kiemelés>}</code>	

Macskakörmöt használunk, ha valakinek a gondolatait idézzük, vagy valakinek a sajátos szóhasználatát adjuk vissza. A macskakörmön belüli idézésre befelé álló lúdlábat használunk (azon belül pedig 9-es alakú félidézőjelet). Például: Róza így gondolkozott: „bár Géza a kerület »legóvatosabb söfőrjének« mondja magát, biztonságosabb busszal menni”. Ma általános, de kerülendő gyakorlat, hogy bizonytalan megfogalmazás esetén is macskakörmöt használunk. Helyette érdemes egy más, macskakörmöt nem igénylő megoldást keresni. A `\textqq` parancsok egymásba ágyazhatók, és mindig az idézet szintjének megfelelő idézőjel kerül a dokumentumba. Ha nyelvet váltunk (`\selectlanguage`), akkor a `\textqq` angol tipográfia szerint fog működni, függetlenül a nyelvtől. Hosszabb idézeteket `\textqq` helyett a `quote` vagy `quotation` környezetbe helyezzük el.

<code>\selectlanguage{magyar}\strut\\</code>	
így emelek ki: az <code>\emph{Auróra}</code> cirkáló\\	így emelek ki: az <i>Auróra</i> cirkáló
és: <code>\emph{még a \emph{Titanic} is}</code> \\	és: <i>még a Titanic is</i>
így idézek: <code>\textqq{külső}</code> \\	így idézek: „külső
<code>\textqq{középső \textqq{belső} középső}</code> \\	»középső 'belső' középső«”
<code>\selectlanguage{french}---\\</code>	—
<code>\textqq{angol külső \textqq{angol belső}}</code>	“angol külső ‘angol belső’”

A különböző idézőjeleket előhozó parancsokat az alábbi példa szemlélteti. A három- és négybetűs parancsok csak a Babel betöltése mellett használhatók, és mind OT1, mind T1 fontkódolásban elérhetők. Az OT1 fontkódolásból hiányoznak a szimpla és dupla lúdlábak (ezeket a Babel hasonló szimbólumokkal helyettesíti), továbbá a hüvelyket jelölő " helyett is `\textquotedblright` jelenik meg.

<code>foo\\</code>	foo
<code>bar\\</code>	bar

Az alábbi parancsok csak T1 fontkódolásban érhetőek el, de nem a Babel definiálja őket. Használatuk magyar szövegben nem javasolt, helyettük inkább a Babel betöltése után a fenti példa három- és négybetűs parancsai ajánlottak.

<code>\quotedblbase~\quotesinglbase\\</code>	” ’
<code>\guillemotleft~\guillemotright\\</code>	« »
<code>\guilsinglleft~\guilsinglright</code>	‹ ›

Művek címeit, és egyéb, tárgyakra vonatkozó tulajdonneveket dőlt betűvel ajánlott szedni (`\emph`). Szokásos még irodalmi szövegben egész bekezdéseket kiemelni, ha azok álomban, vagy a főszövegtől eltérő helyszínen játszódnak (`\em`). Például:

<code>a \emph{János vitéz} elbeszélő költemény\par</code>	a <i>János vitéz</i> elbeszélő költemény
<code>\parindent3mm{\em Egy deszkán találta magát,</code>	<i>Egy deszkán találta magát, amely a tenger hullámain</i>
<code>amely a tenger hullámain zötykölődött.</code>	<i>zötykölődött.</i>
<code>\mond Hol a \emph{Titanic}? -- kérdezte,</code>	– <i>Hol a Titanic? – kérdezte, de nem kapott választ.</i>
<code>de nem kapott választ.\par</code>	– <i>Ez nem lehet – szólalt meg ismét. – Öt perce még a</i>
<code>\mond Ez nem lehet -- szólalt meg ismét.</code>	<i>kabinomban voltam.</i>
<code>-- Öt perce még a kabinomban voltam.}</code>	

<code>\mond <kimondott szöveg></code>	∈ magyar.ldf, <code>mond=yes</code>
<code>\hang{<bekezdéskezdő jel>}<bekezdés></code>	∈ magyar.ldf, <code>hang=yes</code>

Ha egy szereplő a műben megszólal, az általa mondott szöveget új bekezdésben, gondolatjellel kezdjük. A gondolatjel után nem nyúló szóközt kell hagyni. Ezt valósítja meg a `\mond` parancs, melyre példát fent láthattunk. A kimondott szövegbe gondolatjelek közt leírást is ékelhetünk, melyet ponttal kell lezárni. A kimondott szöveg végére szükség esetén ki kell tenni a kérdőjelet vagy a felkiáltójelet, de a pontot tilos. Az fenti példa ezt is szemlélteti.

Ad hoc szervezésű függő felsorolásokhoz lehet hasznos a `\hang` parancs, amely az új bekezdés első sorát a bal margón kezdi, a további sorokat pedig a megadott jel mögött. Ha lehetséges, használjuk helyette a `\mond`-ot, vagy az `itemize`, `enumerate` vagy `description` környezetek valamelyikét. A `\hang` listakörnyezeteken kívül használandó, ebben (is) különbözik az `\item`-től. Például:

<code>\hang{-- }Első sor\\</code>	– Első sor
<code>második sor.\par</code>	második sor.
<code>\hang{\${\bullet}\circ\$ }Egy másik.\par</code>	●○ Egy másik.

2.6. Az aktív karakter

Alapértelmezésben a magyar.ldf a fordított aposztróf (‘) karaktert teszi aktívvá (ez megváltoztatható az `active- \rightarrow prefix=` opcióval), továbbá definiálja az `\shu` parancsot, és ezután az `\shu‘` és a `‘` ugyanarra használható.

Vegyük figyelembe, hogy az aktív karakter (a Babel terminológiájában *shorthand*) az öt követő karaktert makróparaméterként veszi, tehát például `‘=` írható így: `‘{=}` vagy így is: `‘_{}=`. Ezek közül a legrövidebb forma ajánlott. Ha `\string‘`-et gépelünk, az aktív karakter speciális jelentése eltűnik. Így például `\string‘\string‘` egy angol nyitó idézőjelet eredményez, feltéve, hogy az aktuális fontban van ilyen ligatúra.

A magyar.ldf aktív karaktere az alábbi funkciókat nyújtja:

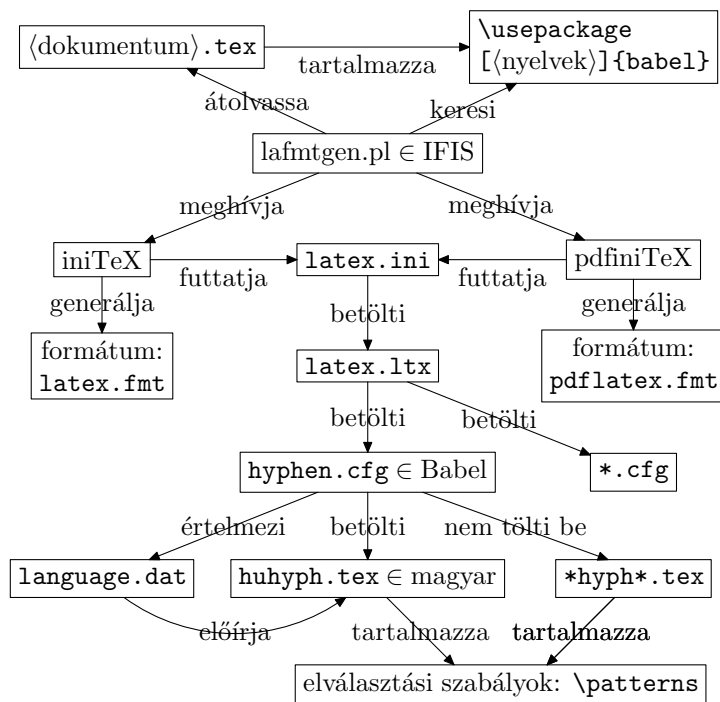
```
‘ ‘      „ magyar nyitó idézőjel, openqq= miatt eltérhet
‘ ’      “ angol nyitó idézőjel, openqq= miatt eltérhet
‘ =      szón belüli kötőjel1
         \showhyphens{tudja-e tudja‘=e} → tudja-e tud-ja-e
‘ _      szón belüli feltételes kötőjel
         \showhyphens{bohém\-életre bohém‘_életre} → bohém-életre bo-hém-élet-re
‘ --     gondolatjel kis szóközzel, személynevek elválasztására
         Kiss Előd‘--Nagy Pál → Kiss Előd–Nagy Pál
‘ -      ugyanaz, mint ‘=, kivéve matematikai képletben, vagy ha így folytatódik: ‘--
         matematikai képletben törhető szóköz: $1<2, ‘-3<4$ → 1 < 2, 3 < 4
‘ |      fontos kötőjel, sor végén és elején is megjelenik
         \showhyphens{nátrium‘|klorid} → nát-ri-um-||-klo-rid
‘ <      « francia nyitó idézőjel (guillemot vagy guillemet)
‘ >      » francia záró idézőjel (guillemot vagy guillemet)
‘ "      szón belüli \allowbreak, résszavak elválasztása automatikus
         \showhyphens{kutatás‘"+“fejlesztés} → ku-ta-tás+fej-lesz-tés
‘ ~      ligatúrát nem képező kötőjel, pl. ‘~‘~‘~ → ---
‘ ccs    sor közepén „ccs”, elválasztva „cs-cs”
         \showhyphens{loccsan szétlo‘ccsantja} → loccsan szét-locs-csant-ja
‘ CCS    sor közepén „CCS”, elválasztva „CS-CS”
‘ cxy    sor közepén „cxy”, elválasztva „cs-xy”; de csak akkor hasznos, ha xy = cs.
‘ ddz    hasonló az előzőkhöz, pl. „eddzünk”
‘ DDZ    hasonló az előzőkhöz
‘ ddzs   hasonló az előzőkhöz, nem szerepel értelmes magyar szóban
‘ DDZS   hasonló az előzőkhöz
‘ ggy    hasonló az előzőkhöz, pl. „poggyász”
‘ GGY    hasonló az előzőkhöz
‘ lly    hasonló az előzőkhöz, pl. „Kodállyal”
‘ LLY    hasonló az előzőkhöz
‘ nny    hasonló az előzőkhöz, pl. „mennyei”
‘ NNY    hasonló az előzőkhöz
‘ ssz    hasonló az előzőkhöz, pl. „vissza”
‘ SSZ    hasonló az előzőkhöz
‘ tty    hasonló az előzőkhöz, pl. „pottyán”
‘ TTY    hasonló az előzőkhöz
‘ zzs    hasonló az előzőkhöz, pl. „rizzsel”
‘ ZZS    hasonló az előzőkhöz
```

A kettős mássalhangzókat kivéve a fenti funkciók más nyelvek .ldf-jei, különösen az `ukraineb.ldf` alapján készültek.

2.7. Címsorok

```
!! figurecaptions=hu
   longcaption=justified
   postpara=unchanged
   postsubpara=unchanged
   sectiondot=safe
   titles=
   amstocnumskip=
   amstocnumlang=all
```

¹ Abban különbözik a sima kötőjeltől (-), hogy ha szerepel egy szóban, akkor a \TeX a résszavakat még megpróbálja elválasztani.



2. ábra. Az elválasztási szabályok és a \LaTeX formátum kapcsolata

2.8. Automatikus szóelválasztás és a huhyph.tex

Ha kíváncsiak vagyunk, hogy a \TeX hol tud elválasztani egy szót, használjuk a `\showhyphens{<szó>}` parancsot. A parancs kimenete a konzolon (abban az ablakban, melyben a \TeX fut) megjelenik. Érdekes utána rögtön valami hibásat gépelni, (pl. `\HIBA`), hogy a \TeX fordítás közben megálljon, és nyugodtan megtekinthessük az lehetséges elválasztási pozíciókat.

A \LaTeX a konzolra írt első sorok egyikében jelzi, hogy mely nyelvek szavait képes automatikusan elválasztani, például: `Babel <v3.7h> and hyphenation patterns for american, french, german, ngerman, magyar, nohyphenation, loaded..` Ha ez a kiírás nem tartalmazza a magyar nyelvet, de mégis betöltöttük, akkor a konzolon egy figyelmeztetés (*warning*) is megjelenik: `Package babel Warning: No hyphenation patterns were loaded for the language 'magyar'.` Ekkor persze már a `\showhyphens{asztal}` is kötőjel nélkül írja ki az `asz-tal-t`. Ha ez a helyzet, akkor aktiválnunk kell a magyar elválasztási szabályokat.

A \LaTeX csak akkor fogja az ékezetes betűket tartalmazó szavakat elválasztani, ha T1 fontkódolást használunk. Ez biztosítható például a `tlenc.sty` csomag betöltésével, vagyis a preambulumban kiadott `\usepackage{tlenc}` sorral. Például `\showhypens{Árvíztűrő Tükörfúrógép}`-vel ellenőrizhetjük, hogy minden ékezetes betűt jól gépeltünk-e be. Ha az elválasztás hiányos, akkor módosítani kell vagy az `inputenc.sty` betöltési opcióin, vagy a `.tex` file szerkesztéséhez használt program mentési opcióin.

A \TeX -et a szerzője úgy tervezte, hogy csak az ún. formátumba (`.fmt` file) beleégetett elválasztási szabályokat ismerje. Hiába van meg tehát a `huhyph.tex` vagy a `huhyph2.tex` a magyar szabályokkal, ha ezek tartalma nem került bele a formátumba, azaz nincs aktiválva. A \LaTeX formátum generálásáról és benne az elválasztási szabályok (`\patterns`) szerepéről áttekintést ad a 2. ábra. Nem kell értenünk azonban az ábra minden részletét ahhoz, hogy aktiválni tudjuk a magyar nyelvet.

Az aktiválás egy egyszer elvégzendő feladat, melynek pontos lépései függenek a \TeX disztribúciónktól. Mik \TeX -ben a *Start menü*-ből hozzuk elő a Mik \TeX beállításait, és ott az elválasztásnak (Hyphenation) fenntartott fülön vegyük fel a magyar nyelvet mind a pdf \LaTeX -hez, mind a \LaTeX -hez. Ezután néhány másodpercig dolgozik a gép, újragenerálja a formátumokat. Ha kész van, a fent leírt módon egy példadokumentumban ellenőrizzük, hogy sikeres volt-e az aktiválás.

Ha UNIX alatt `teTeX`-et használunk, akkor rendszergazdaként (`root`) a `texconfig` paranccsal aktiválhatunk nyelveket. A parancs futtatása előtt ne felejtjük el beállítani az `EDITOR` környezeti változót egy olyan szövegszerkesztőre, melyet kezelni tudunk, azaz például így érdemes indítani a `texconfig`-ot: `su - -c 'env EDITOR=mcedit texconfig'`. A hamarosan megjelenő menüből válasszuk a `HYPHEN`, majd a `latex` menüpontot. Elindul a szövegszerkesztőnk. A `%magyar`-ral kezdődő sor elejéről töröljük ki a százalékjelet. (Ha eredetileg nem volt ott százalékjel, akkor írjuk be, mentsük el, lépünk ki, válasszuk újra a `HYPHEN`-t és a `latex`-et, és írjuk be újra a százalékjelet.) Mentsük el a módosításokat, majd várjuk meg, amíg az `iniTeX` lefut, és sokmindent kiír a képernyőre. Lépünk ki a `texconfig`-ból. Próbáljuk ki egy példaprogramon, hogy működik-e már a magyar elválasztás.

Ha `teTeX`-et használunk, de nincs rendszergazdai jogosultságunk, akkor futtassuk a `lafmtgen.pl` Perl script-et

a Magyar \LaTeX -ből. Először írjuk meg a `.tex` dokumentum preambuláját, hogy a `\usepackage[...]{babel}` sorban az összes szükséges nyelv szerepeljen, majd adjuk ki a `lafmtgen.pl --latex <dokumentum>.tex` parancsot. Ha pdf \LaTeX -et (is) használunk, akkor `--latex` helyett `--pdflatex` kapcsolóval (is) hívjuk. A parancs az aktuális könyvtárban létrehozza a megfelelő `.fmt` formátumfile-t. Ha más könyvtárban is szeretnék \LaTeX dokumentumokat fordítani, akkor ott is meg kell hívnunk a script-et.

A magyar nyelv többjegyű mássalhangzói elválasztáskor különválnak (pl. mennyi \rightarrow meny-nyi). A \TeX eredendő korlátai miatt a `huhyph.tex`-ben leírt elválasztási szabályok nem találják meg ezeket az elválasztási pontokat. Ha mégis itt kívánnánk egy szót elválasztani, akkor egy fordított aposztróffal jelöljük meg a többjegyű mássalhangzót, pl. `me'nyyi`, `időho'sszabbítás`. A `magyar.ldf` 1.5-ös verziója óta az így megjelölt kettős mássalhangzókon kívül a \TeX el tudja másról is választani a szót, például: `\showhyphens{időho'sszabbítás}` \rightarrow `idő-hosz-szab-bí-tás` (feltéve, hogy az ékezetes betűk elválasztásához szükséges `tlenc.sty` csomag be van töltve).

Az összetett szavakat az összetétel határán kell elválasztani. Ez a szabály erősebb annál, hogy egy mássalhangzót kell átrakni a következő szótagba. Így például a `vadember vad-em-ber`-ként szótagolandó. A `huhyph.tex` emellett több, mint 1700 ilyen kivételt felsorol, sőt, még olyanokat is, mint például `ha-lász-zsák-mány`, amit egyébként a `zsz` karaktersorozat miatt nem tudna elválasztani. A kivételek listája nyilván nem lehet teljes, például hiányzik belőle a `télapó`. Ez a dokumentumban elhelyezett `\hyphenation{tél-a-pó tél-a-pót}` hívással vehető fel a kivételek közé. Sajnos az összes toldalékolt alakot és az összetett szavakban szereplő előfordulásokat is külön meg kell adni, ezért a `\hyphenation` használata fáradságos lehet. További hátrány, hogy kettős mássalhangzót tartalmazó kivételek nem adhatók meg, tehát `\hyphenation{me'nyiség meny-nyi-ség}` egyike sem fogja a kettős mássalhangzók határán elválasztani a `mennyiség` szót. Egyedül a `\hyphenation{mennyi-ség}` helyes, de ez sem választ el a határon, továbbá fölösleges is, mert ezt a `huhyph.tex` le tudja vezetni a szabályokból. A `\hyphenation` által létrehozott kivételista globális, felvett elemek nem távolíthatóak el.

Az elválasztási kivételekkel az alábbiak szerint érdemes eljárni. A dokumentumot a szerző a kivételekre és a kettős mássalhangzókra való tekintet nélkül gépeli be. A lektor észreveszi, ha valahol szebb bekezdéskép lenne kialakítható egy kettős mássalhangzó elválasztásával. A szerző ezután a szövegszerkesztő (*text editor*) cseré funkciójával a szó összes előfordulását cseréli '-t tartalmazóra. Ha a lektor elválasztási hibát vesz észre, a szerző a dokumentum elején felveszi a szót a `\hyphenation` kivételek közé, ügyelve arra, hogy a kettős mássalhangzókat ne válassza el. A szerző a biztonság kedvéért még felvehet néhány ragozott alakot, de nem törekszik a teljességre.

A hozzáértők megpróbálkozhatnak a `huhyph.tex` kivétellistájának bővítésével. Ennek előnye, hogy a toldalékolt alakok és az összetett szavak jobban illeszthetők. Érdemes ekkor a `huhyph.tex`-et átmásolni `huhyph0.tex` névre, a másolatot módosítani, és a `language.dat`-ban is átírni a filenevet. Emlékeztetünk arra, hogy ezután a formátumot újra kell generálni ini \TeX -hel a 2. ábrának megfelelően.

A kettős mássalhangzók '-fal való ellátását segíti a Magyar \LaTeX részét képező a `css_extract.pl` Perl script. Ha például a `mű.tex` (esetleg többször) tartalmazza a `loccsan`, `hosszú`, `leggyakoribb` és `mennyi` szavakat, akkor a `css_extract.pl mű.tex` parancs hatására az alábbi sorok a `mű.tex` végéhez fűződnek:

```
%% ho'sszú 5
%% le'gyakoribb 10
%% lo'ccsan 6
%% me'nyyi 3
```

A `css_extract.pl` elvárja, hogy `latin2`-es kódolásban legyenek az általa feldolgozott file-ok. Az ékezetes betűket \-es formájukban nem ismeri fel. Ha a `.tex` file preambulájában eleve `\usepackage[latin2]{inputenc}` szerepel, és az minden ékezetes betű egyetlen karakter, akkor a `css_extract.pl` biztosan jól fog működni.

A fenti listában a számok arról tudósítanak, hogy a file-ban hányszor fordult elő '-ok nélkül az adott szó. Ezután a szerzőnek a szövegszerkesztőben a megfelelő '-ainak egyik aposztrófát törölnie kell, ha a cserét végre szeretné hajtani. Példánkban a `le'gyakoribb` változatlanul hagyandó, mivel szóösszetételről és nem dupla kétjegyű mássalhangzóról van szó, a másik három szó viszont módosítandó a file végén: `ho'sszú`, `lo'ccsan` és `me'nyyi`. A `css_extract.pl mű.tex` parancs újbóli lefuttatása $5 + 6 + 3$ aposztrófbeszúrást végez a file belsejében. Mostmár újra lehet fordítani a `mű.tex`-et, és a \LaTeX el fogja választani a dupla kétjegyű mássalhangzókat is. A dokumentum bővülése után a `css_extract.pl` egyszeri futtatása beszúrja az aposztróft a már ismert szavakba, az ismeretlen szavakat pedig felveszi két aposztróffal a file végére. Ezek előfordulásait a fent leírt módon, a file szerkesztése után a `css_extract.pl` ismételt futtatásával cserélhetjük.

A `css_extract.pl` több file-ban is képes egyszerre cserélni. Például a `css_extract.pl mű.tex *.tex` hívás az aktuális könyvtár összes `.tex` kiterjesztésű file-ját feldolgozza (UNIX alatt), de a statisztikát a `mű.tex` végén készíti el. Nem baj, hogy a `*.tex` tartalmazza a `mű.tex` filenevet is: ez csak annyi változást hoz, hogy a számadatokba a `mű.tex` szavai többszörösen számítanak bele.

Jelen dokumentum írásának egy köztes fázisában a `css_extract.pl` az alábbi dupla kétjegyű mássalhangzókat tartalmazó szavakat találta: `ho'sszabb ho'sszú ho'sszúak ho'sszúságú kla'sszikus 0 kulcsszavakat le'gyakoribb le'gyorsabban lo'ccsan me'llyel me'nyyi na'gyá ö'ssze ö'sszefűzésére ö'sszeget ö'sszehasonlítás ö'sszehasonlításakor ö'sszehasonlításnál ö'sszes ö'sszesítve ö'→`

sszetett ö’sszetételnél ö’sszevetnünk ö’sszevonását ö’sszevonja paran’ccsal paranc’sszavak paranc’sszót ‘ttype tudá’sszintje vála’sszuk ve’sszövel vi’ssza vi’sszaáll vi’sszautasítja vi’sszük.

2.9. A sortörés irányítása

Az alábbi, a sortöréssel és az elválasztással kapcsolatos legfontosabb parancsok közül a ‘-fal kezdődőek csak akkor elérhetőek, ha a a magyar.ldf activeprefix= opcióval lett betöltve, és a magyar nyelv éppen aktív.

`\break` vagy `\penalty-1000`. Az adott ponton mindenképpen eltöri a sort. A sortörés előtti sort a \TeX kizárja a jobb margóig. Használható függőleges módban is, ekkor sortörés helyett oldaltörést végez.

`\hfill\break`. Mint a `\break`, de az előző sort nem húzza szét a jobb margóig.

`\leavevmode\nobreak\hskip0pt plusfil\null\hskip0pt plus-1fil\relax`. Megenged (de nem kényszerít ki) egy sortörést ezen a ponton, és sortörés esetén a felső sort balra igazítja. Hosszabb, eltörhetetlen szövegrészek előtt praktikus alkalmazni, például akkor, ha az irodalomjegyzékben egy webcím jelenik meg. Ha a főszövegbe egy hosszabb, eltörhetetlen szövegrész kerül, azt érdemes – a tartalomtól függően – ábraként, táblázatként, forrásszöveggént (`verbatim` környezet) vagy kiemelt egyenletként (`displayed equation`) beszúrni, mivel a balra zárt sorok rontják az oldalképet, a ritka sorok (`Underfull \hbox`) pedig ezen felül nehezen olvashatóvá is teszik a szöveget.

`\discretionary{<jobb>}{<bal>}{<egybe>}`. Sortörés esetén a felső sor jobb oldalán `<jobb>`, az alsó sor bal oldalán `<bal>` jelenik meg. Ha a \TeX nem itt töri el a sort, akkor `{<egybe>}` fog megjelenni. Sortörés esetére a \TeX `\penalty\exhyphenpenalty`-t helyez el, ha a `<jobb>` üres, egyébként pedig `\penalty\hyphenpenalty`-t. Ha egy `\discretionary` sorhatárra kerül, akkor a sorok oldalakra tördelésekor a \TeX egy `\penalty\brokenpenalty`-t szúr be. Példák: `időho\discretionary{sz-}{sz}{ssz}abbítás` vagy `időhos\discretionary{z-}{z}{sz}zabbítás`.

Ha azt szeretnénk, hogy az előtte levő szórészt a \TeX próbálja saját maga elválasztani, tegyünk elé `\nobreak`-et. Ha a jobbra levő szórészt is el szeretnénk választani, tegyünk a parancs mögé `\nobreak\hskip0pt\relax`-ot (vagy `\makeatletter` után `\nobreak\hskip\z@skip`-et), hogy a `\showhyphens`-ben ne jelenjen meg szóköz. Tehát az „időhosszabbítás” végleges alakja `időhos\nobreak\discretionary{z-}{z}\nobreak\hskip\z@skip szabbítás`, de ezt a a magyar.ldf aktív karaktere segítségével rövidebben is begépelhetjük: `időho’sszabbítás`.

- . Egy olyan kötőjel, amely után el szabad törni a sort. A \TeX a kötőjel után automatikusan `\discretionary{-}{-}{-}`-t helyez el.

`\leavevmode\hbox{-}`. Nemtörő kötőjel (olyan kötőjel, amely után sortörés nem megengedett). A \TeX az őt körülvevő szavakat sem próbálja elválasztani. Ha mégis ez lenne a cél, akkor a már fent ismertetett módon (`\leavevmode\nobreak\hbox{-}\nobreak\hskip\z@skip`) lehet elérni. A `\leavevmode` arra szolgál, hogy a \TeX a `\hbox`-ot vízszintes módba pakolja. Ha a nemtörő kötőjel nem a bekezdés elején van, a `\leavevmode` elhagyható.

\- . Feltételes kötőjel. Egyenértékű `\discretionary{-}{-}{-}`-l-el. A feltételes kötőjelet tartalmazó szavakat a \TeX egyéb pontokon nem próbálja elválasztani, tehát a `\showhyphens{gyermek\egészségügy}` \rightarrow `gyermek-egészségügy` lesz.

‘_ vagy `\leavevmode\nobreak-\nobreak\hskip\z@skip`. Olyan feltételes kötőjel, amely nem zavarja az őt körülvevő szórészek elválasztását. Például `\showhyphens{gyermek\nobreak-\nobreak\hskip\z@skip egészségügy}`, vagy a magyar.ldf aktív karakterét használó `\showhyphens{gyermek‘_egészségügy}` ezt eredményezi: `gyermek-egész-ség-ügy`. Az `egész`-t alapértelmezésben azért nem választja el a \TeX , mert az `egészségügy`-et tekinti egy szónak, és a magyar.ldf `hyphenmins=22` alapbeállítása miatt csak a szó második karaktere után hajlandó elválasztani.

Matematikai módban nincs sok haszna, mert ekkor is szöveges feltételes kötőjelként működik, tehát nem azonos a `\discretionary{-}{-}{-}`-l-el, mert az egy feltételes kivonás (aminek szintén nincs sok haszna).

‘= vagy `\leavevmode\nobreak-\hskip\z@skip`. Szón belüli kötőjel, amely nem akadályozza meg, hogy a \TeX a szó többi részét megpróbálja elválasztani. Például: `\showhyphens{tudja-e tudja‘=e}` \rightarrow `tudja-e tudja-e`.

‘| vagy `\leavevmode\nobreak-\discretionary{-}{-}{-}\nobreak\hskip\z@skip`. Fontos kötőjel. Sorhatáron a felső sor végén és az alsó sor elején is megjelenik. Akkor használjuk, ha szeretnénk hangsúlyozni, hogy az épp elválasztott szót kötőjellel kell írni, például `\showhyphens{nátrium|klorid}` \rightarrow `nát-rium-||-klo-rid`.

`\nobreak` vagy `\penalty1000`. Az adott ponton nem engedi eltörni a sort, vagyis az őt követő `\hskip`-et nem törhetővé változtatja. Az egymást követő `\penalty`-k közül a legkisebb jut érvényre, tehát a `\break`–`\nobreak` és a `\nobreak``\break` egyaránt azonnali sortörést idéz elő. A `\nobreak` használható függőleges módban is, ekkor az oldaltörést akadályozza meg.

`\allowbreak` vagy `\penalty0`. Az adott ponton megenged egy sortörést, de nem kényszeríti ki. Főszövegen belüli matematikai képletekben hasznos. Egy hosszabb, nem szöveges adatot, például egy webcímet bárhol el szabad törni, s bár ilyenkor az `\allowbreak` igen jó szolgálatot tesz, helyette szebb, ha valamilyen `\discretionary`-val jelezzük, hogy a sortörés helyén tilos szóközt hagyni. Például a CTAN webcímét (<http://www.ctan.org/>) elválasztás szempontjából így praktikus begépelni:

```
\def\URLdefs{\frenchspacing          | Próba Próba Próba Próba Próba http://www.ctan.~
\def\|{\nobreak\hskip\z@skip        | org/ http://www.ctan.org/ http://www.ctan.org/
\discretionary{\hbox{\$ \not$}}{-}{%
\nobreak\hskip\z@skip}%
\def\b##1{\| \ifx\b##1\empty
\else\expandafter##1\fi}%
\def~##1{\b\string##1\b}%
}
Próba Próba Próba Próba Próba
{\URLdefs\|tt http:~/~/www~.ctan~.org\|/}
{\URLdefs\|tt http:~/~/www~.ctan~.org\|/}
{\URLdefs\|tt http:~/~/www~.ctan~.org\|/}
```

A példában `\|` felel meg az `\allowbreak`-nek, de sortörésnél egy `/`szimbólumot helyez el. Két oldalán engedélyezi a szóelválasztást, de mivel a `\tt` font `\hyphenchar` értéke -1 , ezért a \TeX nem fog egyéb elválasztójeleket beszúrni. A `\b` makró a `\|`-at hívja, de `\b\b` esetén is csak egyszer. A `~` makró pedig az őt követő karaktert úgy jeleníti meg, hogy előtte és utána is engedélyezett a `\|` szerinti sortörés. Az URL utolsó `/` jelét azért nem lehet `~`-lal megjeleníteni, mert a második `\b`-nek már nincs argumentuma. A `~` definíciójában szereplő `\string` bizonyos ASCII karakterek speciális jelentését szünteti meg (de az alábbiakkal még probléma lehet: `\%{}`).

Szerencsére létezik egy `url.sty` csomag, amelynek az egyszerű `\url{http://www.ctan.org/}` parancsa automatikusan elhelyezi a `\allowbreak`-et – sajnos azonban `\discretionary`-t nem tud.

`\MathBrk{<szimbólum>}`. Matematikai képlekben olyan szimbólumot helyez el, ami sortörés esetén a következő sor elején megismétlődik. A további részleteket a 2.10. alszakaszban.

`' -` vagy `\mskip2.4mu``plus3.6mu``minus1.8mu`. Matematikai képletben a `_`-nél hajszálnyival keskenyebb, elválasztható szóközt helyez el, feltéve, ha matematikai írásjel (*punctuation*) után szerepel. Az `\mskip` csak matematikai módban működik, a `' -` pedig szöveges módban más eredményez, és csak akkor elérhető, ha a magyar nyelv aktív. Példa:

```
\def\mspace{\mskip2.4mu plus3.6mu minus1.8mu}
$x=1,\mspace y\ge z,\mspace f(x,y)=5$\\
$x=1,' - y\ge z,' - f(x,y)=5$\\
$x=1$, $y\ge z$, $f(x,y)=5$
$$x=1,\mspace y\ge z,\mspace f(x,y)=5$$
```

$$\left. \begin{array}{l} x = 1, y \geq z, f(x, y) = 5 \\ x = 1, y \geq z, f(x, y) = 5 \\ x = 1, y \geq z, f(x, y) = 5 \\ x = 1, y \geq z, f(x, y) = 5 \end{array} \right\}$$

2.10. Matematikai képletek

A \LaTeX alapbeállításai túl bő térközöket hagynak a képletekben. A térközök csökkenthetők a `magyar.ldf` `math-muskips=hu` opcióval való betöltésével. Ez az opció a `\thickmuskip`, `\medmuskip` és `\thinmuskip` \TeX belső távolságok állításával igyekszik megfelelni a magyar tipográfiának – bár teljesen nem tud, mert a magyar tipográfia a \TeX térköz-generálásától lényegesen eltérően építi fel a szabályokat, melyek követése ráadásul kevésbé tetszetős képleteket eredményezne, mint a fenti szűkítő opció.

<code>\MathBrk {<matematikai szimbólum>}</code>	<code>∈ magyar.ldf, mathbrk=</code>
<code>\MathBrkAll{<matematikai szimbólum>}</code>	<code>∈ magyar.ldf, mathbrk=</code>

A magyar tipográfia a főszövegen belüli belső képletekben a bináris relációk és a bináris operátorok körül (a `.` szorzásjellet és `/` perjellet kivéve) megengedi a sortörést, de sortörés esetén az operátort illetve relációt a következő sor elején meg kell ismétlni. Erre szolgál a `\MathBrk` parancs, amely egyetlen szimbólumot vár paraméterül, ami vagy egyetlen karakter (pl. `+`), vagy egy parancs (pl. `\cup`). A `\MathBrk` csak akkor elérhető, ha a `magyar.ldf`-et megfelelő `mathbrk=` opcióval töltöttük be. A `\MathBrk` parancs csak akkor módosítja a szimbólumot, ha a magyar nyelv aktív, a `\MathBrkAll` parancs viszont mindig.

A szorzásjel körüli elválasztás megakadályozható `\nobreak\cot\nobreak` gépelésével `\cdot` helyett, vagy a `\cdot` parancs átdefinálásával: `\DeclareRobustCommand\cdot{\nobreak\mathchar"2201\nobreak}`. A perjel körül alpból nem választ el a \LaTeX , de ha `\slash`-ként visszük be a perjelet, akkor helyette `\nobreak\slash\nobreak`-et kell írni, de egyszerűbb simán `/`-t gépelni.

Ha a `magyar.ldf` a `mathbrk=fix` opcióval van betöltve, akkor a szorzásjel és a perjel mellett nem engedélyezett a sortörés, és az összes [4]-ben felsorolt bináris reláció és bináris operátor a `\MathBrk` kirakása nélkül is a második sorban ismételve választódik el, feltéve, hogy a magyar nyelv aktív. Ez az opció a minden matematikai képlet elején újradefiniálja az egykarakteres operátorok jelentését, ami kicsit lassítja a \LaTeX -et, és kompatibilitási problémákat okozhat. A `nath.sty` és `amsmath.sty` csomagokkal az együttműködés tökéletes.

`\factorial` \in `magyar.ldf`, `mathfactorial=`

Ha a `magyar.ldf`-et a megfelelő `mathfactorial=yes` opcióval töltöttük be, akkor a képleteken felkiáltójel körüli térköz helyes lesz, tehát a faktoriális igen egyszerűen bevihető, például $\$x=a!+n!k!\$ \rightarrow x = a! + n! + k!$. (Az előző példában figyeljük meg a szorzótényezők közötti szóközt, ami `mathfactorial=yes` nélkül hiányozna.) Ha csak az alapértelmezés szerinti `mathfactorial=define` opció aktív, akkor `!` helyett `\factorial`-t érdemes gépelni, hogy a térközök helyesek legyenek. A `nath.sty` (CTAN-ról letölthető) csomag a `mathfactorial=` opció használata nélkül is a helyes térközzel szűr be.

`\MathReal{<képlet tizedestörtekkel>}` \in `magyar.ldf`, `mathreal=`

A magyar tipográfia a tizedestörtekben az egészrészt és a törtrészt tizedespont helyett tizedesvesszővel választja el. Magyar tizedestörtek nem gépelhetők közvetlenül `\$1,2\$`-ként, mert a vessző körül kihagyott térköz hibás lenne (ugyanis a vessző operátorosztálya *Punct*, a jó osztály, ahová a pont is tartozik, pedig *Ord*). A jó térközt szolgáltatató `\mathchar"013B_2` begépelése fáradságos. Helyette ajánlott a `\MathReal` parancs, amely a paraméterül kapott matematikai képletet úgy jeleníti meg, hogy minden pontot tizedesvesszőre cserél, de csak akkor, ha a magyar nyelv aktív. A paraméterben a pontokat tilos makróba rejteni. A `\MathReal` képleten belül és azon kívül is működik. Például:

Több szám: <code>\MathReal{1.25+2+3.75=7};\ \</code>	Több szám: 1,25 + 2 + 3,75 = 7;
<code>{\selectlanguage{english}\null\ \</code>	
Valós számok: <code>\MathReal{45}; \ \\$\MathReal{-56.07}\\$;\ \</code>	
Real numbers: <code>\\$MathReal{45}\\$; \ \\$\MathReal{+56.07}\\$;\ \</code>	Valós számok: 45; -56.07; Real numbers: 45; +56.07;

Újabbán a magyar természettudományos irodalomban angol mintára az a gyakorlat terjedt el, hogy tizedesvessző helyett tizedespontot használnak. Ennek a gyakorlatnak részben az az oka, hogy a tizedespontot gyorsabb begépelni, részben pedig az, hogy a szoftverek, melyek a numerikus eredményeket szolgáltatják, vagy megjelenítik, többségében tizedespontot generálnak, továbbá a programozási nyelvekben is csak a pont az elfogadott. Épp ezért manapság már kicsit zavaróan is hat, ha valaki a tudományos cikkében a tizedesvesszőt használ.

2.11. A határozott névelő (a, az)

<code>\az*{<szó>}</code>	a vers \in <code>magyar.ldf</code> , <code>az=</code>
<code>\Az*{<szó>}</code>	Az ősz kaján
<code>\az+\parancs{<argumentum>}</code>	az 5. táblázat
<code>\Az+\parancs{<argumentum>}</code>	Az 5. táblázat
<code>\aref*{<címke>}</code>	a 2.11. alfejezet
<code>\Aref*{<címke>}</code>	A 2.11. alfejezet
<code>\aref({<címke>})</code>	az (1) egyenlet
<code>\Aref({<címke>})</code>	a (2) egyenlet
<code>\apageref*{<címke>}</code>	a 15. oldalon
<code>\Apageref*{<címke>}</code>	A 15. oldalon
<code>\acite*{<irodalom-címkék>}</code>	az [1, 2] művek rövidek
<code>\Acite*{<irodalom-címkék>}</code>	A [3, 4] művek szépek

A `magyar.ldf` számos parancsot tartalmaz, melyek automatikusan ki tudják tenni a határozott *a/az* névelőt a szavak elé. A fenti parancsok csillagos változata csak a névelőt (pl. *a*), a csillag nélküli változat pedig a névelővel ellátott szót (pl. *a-vers*) adja. A fenti parancsok csak akkor elérhetők, ha a `magyar.ldf` a megfelelő `az=` opcióval lett betöltve.

A `magyar.ldf` az arab és római számok és a betűk elé is a megfelelő névelőt rakja (pl. `\az{xilofon}` \rightarrow a xilofon, `\az{x betű}` \rightarrow a x betű, `\az{\romannumeral10}.c. osztály` \rightarrow a x.c. osztály²). A római számokat akkor is helyesen kezeli, ha azok pl. egy `\aref` által hivatkozott fejezetszám elején található. Ehhez segítségként

² A gimnáziumi osztályok neveit általában nagybetűvel írjuk, de ezt `\makeatletter` nélkül viszonylag nehézkes előállítani: `\az{\csname @Roman\endcsname{10}}.C. osztály` \rightarrow a X.C. osztály.

az `.aux` file-ba minden `\newlabel` parancs után a `magyar.ldf` elhelyez egy `\hunnewlabel` parancsot, amely már mindenképpen arab számokat tartalmaz. (Ez csak akkor működik, ha a `magyar.ldf`-et a `hunnewlabel=yes` opcióval töltöttük be.) A `<szó>` elején levő írásjeleket nem veszi figyelembe, például `\az{${}^+52}` és `\az{52}` ugyanazt eredményezi. Megzavarhatja a működést, ha a `\szó`-ban a makrók kifejtése után kapcsos zárójelek vagy kifejthető token-ek (*expandable token*) maradnak.

Kompatibilitási okokból a fenti parancsok helyett a `\azr`, `\Azr`, `\azp`, `\Azp`, `\azc`, `\Azc` rövidítések is használhatók, ha a `magyar.ldf`-et a `shortrefs=yes` opcióval töltöttük be.

A `magyar.ldf` 1.5-ös verziójától kezdődően az `\az` és `\Az` parancsok `<szó>` paramétere tartalmazhatja a `\ref`, `\pageref` és `\cite` parancsokat. Továbbá a `+` kiválthat egy kapcsos zárójelt. Így például az alábbiak ugyanazt szúrják be: `\Az{\pageref{nagy}}`, `\Apageref{nagy}`, `\Az+\pageref{nagy}`. Egyenletre az alábbi módokon hivatkozhatunk:

<code>\begin{equation}1<2\label{egy}</code>		
<code>\end{equation}</code>		
<code>(\ref{egy})</code>		(5)
<code>\eqref{egy}</code>		(5) csak amstex.sty vagy amsmath.sty
<code>\az{\eqref{egy}}</code>		az (5) csak magyar.ldf és ams... .sty
<code>\az{(\ref{egy})}</code>		az (5) csak magyar.ldf
<code>\aref{egy}</code>		az (5) csak magyar.ldf

2.12. Betűvel írt számjegyek

<code>\@huordinal{<szám>}</code>	(ezerkilencszáznolcvannegyedik)	∈ magyar.ldf, hunumbers=yes
<code>\@Huordinal{<szám>}</code>	(Ezerkilencszáznolcvannegyedik)	
<code>\@hunumeral{<szám>}</code>	(ezerkilencszáznolcvannégy)	
<code>\@Hunumeral{<szám>}</code>	(Ezerkilencszáznolcvannégy)	

A fenti parancsok `–9999...9999` közötti egész számokat tudnak kibetűzni. Mivel a parancsok neve kukacot is tartalmaz, ezért csak `\makeatletter` kiadása után használhatjuk őket közvetlenül (azaz pl. `\huordinal1984` formában). Átválthatunk betűzött oldalszámozásra pl. a `\pagenumbering{huordinal}` kiadásával. A fenti parancsok csak akkor érhetőek el, ha a `magyar.ldf` a `hunumbers=yes` opcióval lett betöltve.

Lehetséges egy könyv részeit (`\part`) vagy fejezeteit (`\chapter`) kibetűzve számozni. Ezt megtehetjük kézzel úgy, hogy a preambulumban (`\makeatletter` után) kiadjuk pl. a `\def\thepart{\@Huordinal\c@part}` parancsot, vagy a `magyar.ldf`-et a `partnumber=Huordinal` opcióval töltjük be (`\part` helyett `\chapter`-re ugyanígy).

2.13. Számok todalékolása

<code>\told <szám>+<toldalék>{}</code>
<code>\told <szám>+<toldalék₁₂</code>
<code>\atold <szám>+<toldalékok>{}</code>
<code>\Atold <szám>+<toldalékok>{}</code>

Különösen a képletekre való hivatkozásnál lehet hasznos, hogy a `magyar.ldf` a számneveket ragozni tudja (kivéve, ha ezt a `magyar.ldf` betöltésekor a `told=` opcióban le van tiltva). Így például a „(3)-at hozzáadva (4)-hez, majd azt kivonva (6)-ból a (7)-et kapjuk” alakú kifejezéseket akkor is írhatunk, ha a pontos számokat a \LaTeX menet közben generálja. A fenti példa forrása így kezdődik: `\told\aref{eq3}+at{}`... Ha a dokumentumot átrendezzük, és a `eq3` címkéjű képlet sorszáma megváltozik, a `\told` akkor is a helyes todalékot fogja hozzátenni.

A `<szám>` argumentumot az egyértelműség kedvéért kapcsos zárójelbe szabad tenni. A `\told` a parancs végén található `{}` jelből tudja, hol van vége a todaléknak: például a `\told1+be{}` és a `\told1+ben{}` között csak így lehet különbséget tenni. Bizonyos feltételek mellett (például akkor, ha a parancsot pont vagy vessző követi) a `{}` elhagyható, de érdemes a biztonságra törekedve mindig kitenni. A `<szám>` állhat matematikai módban, és lehet negatív is (pl. `\told{${-2$}+edik{}` → `–2-edik`, olvasd: „mínusz kettedik”), de vigyázzunk, mert két mínusz jel kiejti egymást (pl. `\told{--2}+edik{}` → `–2-odik`, de kevesen olvasnák „második”-nak).

A `<szám>` tartalmazhat `\ref`, `\aref`, `\Aref`, `\pageref`, `\apageref`, `\Apageref`, `\cite`, `\acite` és `\Acite` parancsokat, de csak csillag és zárójel nélkül, például `\told\ref{told=}+as{}` `alfejezet` → `2.13-as alfejezet`. Az `\atold` illetve `\Atold` parancsok egyenértékűek egy `\az`-ba illetve `\Az`-ba ágyazott `\told`-dal. A `\told` mindig a legutolsó számot ragozza, tehát például ha az `ott` címkéjű alszakasz sorszáma `1.2.3`, akkor `\told\ref{ott}+ben{}` így jelenik meg: „1.2.3-ban”. A `<szám>` technikai okokból nem tartalmazhat `\refstruc` parancsot, mivel az nem expand-olható. A megoldás: `\told{\refstruc{fe}}+ben` helyett `\refstruc{fe+ben}`-t kell írni.

A számok végére illeszthető todalékok (`<toldalék2) a következők:`

<code>\told42+a{}</code>	42-e
--------------------------	------

<code>\told42+as{}</code>	42-es
<code>\told42+ad{}</code>	42-ed
<code>\told42+adik{}</code>	42-edik
<code>\told42+an{}</code>	42-en
<code>\told42+at{}</code>	42-t
<code>\told42+on{}</code>	42-en
<code>\told42+a{}</code>	42-e
<code>\told42+nal{}</code>	42-nél
<code>\told42+ul{}</code>	42-ül
<code>\told42+val{}</code>	42-vel
<code>\told42+hoz{}</code>	42-höz
<code>\told42+ban{}</code>	42-ben
<code>\told42+nak{}</code>	42-nek
<code>\told42+ba{}</code>	42-be
<code>\told42+ra{}</code>	42-re
<code>\told42+tól{}</code>	42-től
<code>\told42+ról{}</code>	42-ről
<code>\told42+szor{}</code>	42-ször

A folytatható toldalékok ($\langle toldalék_1 \rangle$) a következők:

<code>\told42+as+at{}</code>	42-eset
<code>\told42+ad+szor{}</code>	42-edszer
<code>\told42+adik+nal{}</code>	42-ediknél
<code>\told42+a+an{}</code>	42-én

Természetesen minden $\langle toldalék_1 \rangle$ minden $\langle toldalék_2 \rangle$ -vel kombinálható, de nincs értelme az összes kombinációnak.

2.14. Hivatkozás a dokumentum strukturális egységeire

`\refstruc{<címke>[+<toldalékok>]}`

A `\refstruc` parancs abban tér el a `\ref` parancstól, hogy nem csak a strukturális egység számszerű megjelölését, hanem a nevét is megjeleníti. A `\refstruc` csak akkor érhető el, ha a magyar.ldf betöltésekor a `refstruc=` opcióban nem tiltottuk le. Ha a magyar nyelv aktív, akkor „1.2. alszakasz” formában jeleníti meg a címet, egyébként „subsection 1.2” formában; az angol elnevezés az egységet nyitó L^AT_EX parancsszóval egyezik meg.

Ha a magyar nyelv aktív, akkor a `\refstruc` elfogad toldalékokat is, a 2.13. alszakaszban leírt módon. A -val/-vel ragot nem tudja helyesen hozzákapcsolni a nevekhez. Ha bekezdésekre (`\paragraph`) is hivatkozni kívánunk, akkor ne felejtsük el a `secnumdepth` számlálót megnövelni (lásd lent). Ne feledjük, hogy – csakúgy, mint `\ref` esetén – csak számozott egységre hivatkozhatunk, tehát például `\section*-ra` nem. Ha `\az+t` írunk a `\refstruc` elé, akkor a megfelelő a/az határozott névelőt kapjuk. Sose használjuk a `\refstruc`-ot ábrákra, táblázatokra, egyenletekre vagy tételszerű környezetekre való hivatkozásra; ezek ugyanis a dokumentumnak nem strukturális egységei, így például ha egy könyvben egy ábra sorszáma „3.4”, akkor is „3.4. szakasz” generálódna. Ennek a fogyatéknak az az oka, hogy a hivatkozott egység típusa nem áll rendelkezésre az `.aux` file-ból a `\refstruc` hívásakor.

Példa a `\refstruc` használatára:

```
\setcounter{secnumdepth}{9}% \paragraph
\chapter{Generált részek}\label{1}
\section{Fejezetszerű részek}\label{2}
\subsection{Tárgymutató}\label{3}
\subsubsection{Segédprogramok}\label{4}
\paragraph{\nproduct{husort.pl}}\label{5}
Itt látható \az+\refstruc{1},
amely kapcsolódik \az+\refstruc{2+hez},
és kiegészíti \az+\refstruc{3+t},
de hallgat \az+\refstruc{4+ról},
amit már leírtunk \az+\refstruc{5+ben}.
\par\selectlanguage{english}\refstruc{3}
```

I. Generált részek

1. Fejezetszerű részek

1.1. Tárgymutató

1.1.1. Segédprogramok

1.1.1.1. husort.pl ♦ Itt látható az I. fejezet, amely kapcsolódik az 1. szakaszhoz, és kiegészíti az 1.1. alszakaszt, de hallgat az 1.1.1. alszakaszról, amit már leírtunk az 1.1.1.1. bekezdésben.

subsection 1.1

csak a szerzőnek segítenek. Az ebben a bekezdésben leírt parancsokat a `magyar.ldf` `hutoday=` opciójával tehetjük elérhetővé, míg a `\emitdate`-nek saját opciója van.

<code>\emitdate[⟨toldalék⟩]{⟨formátum⟩}{⟨ dátum⟩} ∈ magyar.ldf, emitdate=</code>
--

Felismeri a `⟨datum⟩`-ot, és megjeleníti az előírt `⟨formátum⟩`-ban, esetleg `⟨toldalék⟩`-kal látva el. `⟨toldalék⟩` a `\told` parancs `+a` után megengedett tetszőleges toldalék lehet, tipikus értékei: `e` és `a+an`. Az `edik+e` (1848. március 15-ödike) használata nem ajánlott, helyette `e` javasolt.

A `\emitdate` az ilyen formák egyikében megadott `⟨datum⟩`-okat érti meg: `\today`, `É-H-N` (ISO dátumformátum), `É/H/N` (L^AT_EX csomagok keltezésének formátuma), `É/H/N:X:Y` (L^AT_EX alap `\today` parancs formátuma), `É.N.H` (keverék formátum), `É. hónap N` (magyar `\today`), `Hónap N`, `É` (amerikai `\today`, L^AT_EX osztályok `\today` formátuma). A `0...49` közötti évszámokhoz 2000-et, az `50...99` közöttiekhez pedig 1900-at hozzá. A dátum végén lehet egy extra pont, amit figyelmen kívül hagy. Szóközök számítanak. Ha a megadott dátum nem felel meg egyik formátumnak sem, a `\emitdate` nem minden esetben ad értelmes hibaüzenetet, és esetleg a dokumentum fordítása végleg elakadhat.

A `\emitdate` `⟨formátum⟩` argumentuma a kimeneti formátumot adja meg. A magyar helyesírásban minden `(a-h)` formátum helyes, de `h` esetén a `magyar.ldf` nem oldja meg a hónapnevek automatikus ragozását. A lehetséges formátumok:

<code>\emitdate{a}{1848-3-15};\</code>	1848-03-15;
<code>\emitdate{b}{1848-3-15};\</code>	1848. március 15.;
<code>\emitdate{c}{1848-3-15};\</code>	1848. márc. 15.;
<code>\emitdate{d}{1848-3-15};\</code>	1848. III. 15.;
<code>\emitdate{e}{1848-3-15};\</code>	1848. 03. 15.;
<code>\emitdate{f}{1848-3-15} közepé;\</code>	1848. március közepé;
<code>\emitdate[a+an]{g}{1848-3-15};\</code>	1848. március 15-én;
<code>\emitdate{h}{1848-3-15}-ában;%\</code>	1848 március-ában;

2.17. Ékezetes betűk hivatkozásokban

A hivatkozások (`\ref`, `\cite` stb.) neveiben az összes magyar betű megengedett, beleértve az ékezeteseket is, kivéve akkor, ha az `inputenc.sty` csomag a `tlenc.sty` csomag nélkül van betöltve. Ez utóbbi esetben a nem túl barátságos `! Missing \endcsname inserted. ... <to be read again> ... \unhbox` hibaüzenet jelenik meg. Mivel az ékezetes betűket tartalmazó szavak elválasztása miatt a `tlenc.sty` szinte mindig be van töltve, ezért a fenti probléma ritkán jelentkezik.

Ha az `inputenc.sty` nincs betöltve, akkor az `.aux` file-ba az ékezetes betűk hexa formában kerülnek, például az „ó” betű `^^f3`-ként. Ha viszont a csomag be van töltve, akkor parancsok kerülnek az `.aux` file-ba, például: `\'o`. Példa a hivatkozásokban levő ékezetes betűkre:

<code>\section{Fura címke}\label{áÉúŰúŰúŰí}</code>	1. Fura címke
<code>E fejezet sorszáma: \ref{áÉúŰúŰúŰí}</code>	E fejezet sorszáma: 1

2.18. Lábjegyzetek³

A L^AT_EX `\footnote` parancsával a szokásos módon hozhatjuk létre őket. Ha a `magyar.ldf` nem `footnote=unchanged` opcióval van betöltve (hanem pl. `footnote=huplain`), akkor a `\footnotestyle` paranccsal változtatunk a lábjegyzetek stílusán.

A lábjegyzetek magyar tipográfiai szabályainak ([1, 116–118. oldal]-at felhasználva) a `magyar.ldf` a szögletes zárójelben leírt módon próbál megfelelni:

1. Csillagok és arab számozás egyaránt használható a lábjegyzet jelölésére. [OK, a `\footnotestyle{mark=...}` paranccsal a kívánt számozás kiválasztható.]
2. A csillagos számozás minden oldalon újrakezdődik, az arab számozás pedig fejezetenként (*chapter*). [OK, a `\footnotestyle{starplain}` aktiválja a `\footnotestyle{mark=stars-max}`-ot.]
3. A lábjegyzetjel szóköz nélkül követi a szót, amire vonatkozik. [OK, a szerzőnek kell figyelni rá, nehogy a `\footnote` parancs új sorban kezdődjön. Például: `Barack\footnote{Értsd: barackszínű.}` ruha]
4. A lábjegyzet külön mondat, tehát nagybetűvel kell kezdeni, és általában ponttal kell befejezni. [OK, a szerzőre van bízva.]
5. Ha egy műben a szerkesztő és a szerző is helyez el lábjegyzeteket, a szerző lábjegyzeteit arab számokkal, a szerkesztőjét pedig csillagokkal érdemes jelölni. [OK, lásd `\editorfootnote` lent.]

³ Azaz lapolji jegyzetek, angolul: *footnote*.

6. A lábjegyzetek szövege és a főszöveg között elég egy üres sort hagyni, vonalra (`\footnoterule`) nincs szükség. [OK, `\footnotestyle{rule=none}` vagy `\footnotestyle{rule=one-line}`.]
7. A főszöveg és egy hosszú lábjegyzet előző oldalon félbemaradt részének folytatásával induló lábjegyzet-blokk között vonalat kell elhelyezni (melynek hossza a tükörszélesség (`\textwidth`) harmada vagy negyede legyen). [OK, egy legtöbbször jól működő megoldást valósít meg, `\footnotestyle{rule=choose}`.]
8. Rövid lábjegyzetek egy sorba tördelhetők. [OK, ehhez külső csomag szükséges: `\usepackage[para]{footmisc}` (ajánlott) vagy `\usepackage{fnpara}` (régebbi). A `footmisc.sty` a `minipage` környezetben belüli lábjegyzeteket nem teszi egy sorba, a `fnpara.sty` viszont igen.
9. Táblázat illetve ábra lábjegyzetei közvetlenül alatta, és ne az oldal alján jelenjenek meg. [OK, lásd `\makeFootnotable` lent.]
10. Ha nagy betűvel szedett szövegrészt látunk el lábjegyzettel, a lábjegyzetjel ne legyen nagyobb a normál betűmérethez tartozónál. [OK, lásd `\footnotestyle{marksizem=max-normal}`.]
11. A lábjegyzet kezdődhet beütéssel (azaz első sora beljebb kezdődik) vagy listaszerűen (a lábjegyzetjeltől eltekintve az első és a második sor egymás fölött kezdődik) [OK, lásd `\footnotestyle{indent=hulist}`.]
12. A főszövegen belüli lábjegyzetjel előtt és után nincs szóköz, az oldal alján található lábjegyzetjel után pedig egy rövid szóköz (`\,`) van. [OK, lásd `\footnotestyle{indent=article-sp}` és `\footnotestyle{indent=hulist}`.]
13. Ha túl sok csillagos megjegyzés kerülne egy oldalra, akkor más jelre kell áttérni, például: `* ** * ** * † † † † †`. [OK, lásd `\footnotestyle{mark=stars-max}`. A hetedik csillag helyett `*7`-tel számozás kezdődik.]

```
\footnotestyle{előírás}[, \dots] ∈ magyar.ldf
```

Változtat a lábjegyzetek megjelenítési stílusán. Hatása csak a csoporton (*group*) belül érvényesül. (Vigyázat! A lap alján megjelenő részek, például `\footnoterule`, általában az oldal vagy hasáb kiadásakor kerülnek szedésre, ami később történik, mint az oldal főszövegének szedése. Így például az 5. oldalon, egy csoporton belül kiadott `\footnotestyle` visszahathat a 4. oldal fejlécét és láblécét alkotó részekre.) A vesszővel elválasztott `(előírás)`-ok sorrendje lényeges.

A `magyar.ldf` betöltésekor a `footnote=` opció értékétől függően állítja be az alapstílust. `footnote=unchanged` esetén az egész lábjegyzetkezelő rész (benne a `\footnotestyle` paranccsal) be se töltődik. A `magyar.ldf` alapértelmezése `footnote=yes`, ekkor a parancsok betöltődnek, de a stílus nem módosul (tehát kezdetben csak a kernel (`latex.ltx`), az osztály és a betöltött csomagok alakítják). `footnote=huplain` esetén a `\footnotestyle{huplain}` kerül automatikusan kiadásra.

Az `(előírás)` az alábbi értékeket veheti fel:

reset=none. A dokumentumban folyamatosan számozza a lábjegyzeteket. Csak akkor kezdi újra a számozást, ha egyéb csomagok vagy az osztály ezt állítja be. Például `book.cls` esetén a lábjegyzetszámozás fejezetenként újakezdődik.

reset=section. Szakaszonként és fejezetenként újakezdi a lábjegyzetek számozását.

reset=chapter. Fejezetenként újakezdi a lábjegyzetek számozását.

reset=page. 1-től azonnal, és a továbbiakban minden oldalon újakezdi a lábjegyzetek számozását. Az oldalhatárok követéséhez az `.aux` file-t használja, és a konzolon figyelmeztet, ha a változások miatt a `latex`-et újra kell futtatni. Hatóköre végeztével az általa befejezett számozás folytatódik. Például:

<pre>\pagestyle{empty} \setcounter{footnote}{7} {\footnotestyle{reset=page} Egy\footnote{Egy.}\newpage Újra egy\footnote{Újra egy.}} Kettő\footnote{Kettő.}</pre>	<pre>Egy¹ ¹ Egy. ----- Újra egy¹ Kettő² ¹ Újra egy. ² Kettő.</pre>
---	--

A számláló nullázását ügyesen, a `\stepcounter` által meghívott `\c1@footnote` kampólistába (*hook list*) beépülve végzi, így igen jól együttműködik más, a lábjegyzetek megjelenítésébe beavatkozó csomagokkal.

reset=page-resume. 1-től azonnal, és a továbbiakban minden oldalon újakezdi a lábjegyzetek számozását. Hatóköre végeztével az őt megelőző számozás onnan folytatódik, ahol abbamaradt.

reset=page-cont. Minden oldalon újakezdi a lábjegyzetek számozását. Egy oldalon belül többször ki- és bekapcsolható, emiatt nem kezdi 1-től újra. A `page-cont` és a rajta kívüli lábjegyzetek számozása egymástól függetlenül folyamatos. A `\editorfootnote` is ezt használja.

resume. Elintézi, hogy a csoport végén visszaálljon az épp aktuális lábjegyzetszám. Kiegészítő funkció, önmagában nem használatos. Például a `\footnotestyle{resume,reset=page}` és a `\footnotestyle{reset=page-resume}` ugyanazt eredményezi.

indent=article-nosp. Az `article.cls`-sel egyező lábjegyzet-megjelenést állít be: a lábjegyzet első sora beljebb kezdődik a többinél, és a sor eleji lábjegyzetszám után nincs szóköz.

indent=article-sp. Az `article.cls`-hez hasonló, a magyar tipográfiának megfelelő lábjegyzet-megjelenést állít be: a lábjegyzet első sora beljebb kezdődik a többinél, és a sor eleji lábjegyzetszám után egy rövid szóköz (`\,`) van.

indent=hulist. A magyar tipográfiának megfelelő listaszerű lábjegyzet-megjelenést állít be: a lábjegyzet első bekezdésének minden sora a bal margótól az első szintű listával egyező mértékben kezdődik beljebb, a lábjegyzetszám ettől balra, egy rövid szóközzel (`\,`) elválasztva lóg ki.

rule=none. Nem helyez el vízszintes vonalat a főszöveg és a lábjegyzetek között. Lásd még a `\skip\footins`-et. Fontos, hogy a `rule=...` stílusbeállításokat minél korábban adjuk ki, mert értéküket a \TeX valamikor a következő oldal tördelése alatt veszi figyelembe.

rule=one-line. Nem helyez el vízszintes vonalat a főszöveg és a lábjegyzetek között, és e két téglalapot egymástól egy sor $\pm 10\%$ távolságra helyezi el. A sorméret a `\normalsize` beállítás `\baselineskip` értéke. Magyar tipográfiában ez a fix távolság ajánlott. A $\pm 10\%$ esélyt ad a \LaTeX -nek, hogy széthúzza az oldalt.

rule=fourth. Egy vízszintes vonalat helyez el a főszöveg és a lábjegyzetek között, melynek hossza a szedéstükör egynegyede (tipográfiai szempontból ez ajánlott). A két téglalap távolságát nem befolyásolja.

rule=choose. Három különbözőféle vonal közüli választást állít be a főszöveg és a lábjegyzetek között. A `\pagefootnoterule` makrót fogja hívni ha a legfelső lábjegyzet az aktuális (és nem egy korábbi) oldalon kezdődött. Alapértéke az eredeti `\footnoterule` vonalhúzómakró. Ha a legfelső lábjegyzet egy korábbi oldalon *kezdődött*, akkor az `\mpfootnoterule` makrót fogja hívni, melynek alapértéke a `\footnotestyle{rule=fourth}`-szal egyező. `minipage` környezetben viszont egységesen a `\mpfootnoterule` makró hívódik, melynek alapértéke az eredeti `\footnoterule` vonalhúzómakró.

A fenti három makrót a szerző a preambulumban átdefiniálhatja. Arra figyelni kell, hogy a makrókat a \LaTeX – ritka szerencsétlen esetben – egy oldalon többször is meghívhatja, tehát úgy kell megírni őket, hogy többszöri meghívásuk is ugyanúgy nézzen ki, mint az egyszeri. Ha figyelembe vesszük, hogy a vonalak alapszélessége 0.4pt , akkor például az alábbi definíciók helyesek:

```
\def \pagefootnoterule{\kern-3pt \hrule width 2in \kern 2.6pt}
\def \splitfootnoterule{\kern-3pt \hrule width .25\textwidth \kern 2.6pt}
\def \mpfootnoterule{}
```

A `\def\rossz{\unskip\vskip12pt}` az `\unskip` parancs miatt nem jó. Sajnos a `\unskip`-et a `\footnotestyle{rule=one-line,rule=choose}` is hívja, ezért ebben az esetben lábjegyzetek néha függőlegesen elcsúszhatnak. Szerencsére ez és más problémák is csak nagyon hosszú (több, mint kb. 10 soros) lábjegyzetek esetén következnek be.

A fenti három makró elnevezése és használata teljesen kompatibilis a `\usepackage[splitrule]{footmisc}` opcióval.

Hasznos tudni, hogy a lábjegyzet és a lábléc (oldalszám) közé néha beékelődhet egy úszó táblázat (`table` környezet) vagy ábra (`figure` környezet). Ez ellen úgy védekezhetünk, ha a \LaTeX -nek megtiltjuk a lefelé úsztatást, például így: `\begin{table}[th!t!hp]`. A szögletes zárójelek között `b-t` ne szerepeltessünk, mert az írja elő az oldal alját.

marksize=max-normal. A főszövegben megjelenő lábjegyzetjel méretét korlátozza: ha az aktuális betűméret nagyobb `\normalsize`-nál, akkor `\footnotesize`-ra korlátozza a jel betűméretét, egyébként nem változtat rajta.

mark=arabic. A lábjegyzetszámok arab megjelenítését állítja be.

mark=stars. A lábjegyzetszámokat egyre több csillaggal jeleníti meg. Bár a csillag már az alapfontban is felső indexben van (`*`), nem baj, hogy a `\@makefnmark` által hívott `\textsuperscript` hatására még feljebb kerül. Három csillag fölött már kényelmetlenül hosszú ez a számozás, helyette a `stars-max` ajánlott.

mark=stars-max. Az első hét lábjegyzetet így számozza: `* ** * * * † †† ††† *7`. Ezután az arab számozás folytatódik, csillaggal kezdve. Ajánlott visszatérni `mark=arabic`-ra, ha oldalanként átlagosan több, mint 1.5 lábjegyzet van.

mpmark=arabic, mpmark=stars és mpmark=stars-max. A `mark=...` megfelelői, de a `minipage` környezetben belüli lábjegyzetjel (`\mpfootnotemark`) megjelenését módosítják. Továbbá módosítják a `\makeFootnotable` parancs (lásd lent) által előkészített környezeteket is.

plain. Egyenértékű `\footnotestyle{reset=none,mark=arabic}`-kel, ami a \LaTeX alapbeállítása.

huplain. Egyenértékű `\footnotestyle{marksize=max-normal, reset=none, mark=arabic, rule=none, indent=hulist}`-tel, ami a \LaTeX alapbeállításához hasonló, de a magyar tipográfiát követő arabszámú stílus.

starplain. Egyenértékű `\footnotestyle{marksize=max-normal, reset=page, mark=stars-max, rule=none, indent=hulist}`-tel, ami a magyar tipográfiát követő csillagos stílus.

editor. Egyenértékű `\footnotestyle{marksize=max-normal, reset=none, mark=arabic, rule=none, indent=hulist}`-tel, ami a szerkesztő csillagos megjegyzéseihez használatos stílus. A szerző lábjegyzeteit ekkor `huplain` stílusban ajánlott elhelyezni. Lásd még a `\editorfootnote` parancsot lent.

`\makeFootnotable{<környezet-név>} ∈ magyar.ldf`

A megadott úsztatott környezetet felkészíti `\footnote` parancs befogadására. A `<környezet-név>` értéke általában `figure` vagy `table`. Ezek olyan úsztatott környezetek, amelyekben eredetileg a `\footnote` parancs hatására csak a lábjegyzetjel jelenik meg, maga a lábjegyzet hiányozni fog a dokumentumból. `\makeFootnotable{figure}` kiadása után viszont minden `figure` környezetben meg fognak jelenni a lábjegyzetek, mégpedig a cím (`\caption`) fölött. A működés a magyar tipográfiai gyakorlatnak megfelelő, ugyanis az ábra vagy táblázat lábjegyzetét nem a főoldalon, hanem magában az ábrában kell elhelyezni. A `makeFootnotable{}` csak az aktuális úsztatott környezet aktív példányára van hatással.

Alternatív megoldás, ha a szóban forgó környezet tartalmát egy `minipage` környezetben helyezük el. Ekkor azonban számolnunk kell a `minipage` egyéb velejáróival is.

Ha a parancsot a `\usepackage[para]{footmisc}` mellett használjuk, akkor a `footmisc.sty` csomagtól függetlenül, csakúgy, mint `minipage` környezetben, a lábjegyzetek egymás alá fognak kerülni. `\usepackage{fnpara}` esetén viszont e belső lábjegyzetek is egymás mellett jelennek majd meg.

A belső lábjegyzetek megjelenítési stílusát, csakúgy, mint `minipage` környezetben, a `\thempfootnote` makró szabályozza, ami a \LaTeX kernel alapértelmezésében dőlt kisbetűket állít be. Átdefiniálható például félkövér római számokra a `\def\thempfootnote {\bfseries\roman{mpfootnote}}` paranccsal, csillagokra pedig a `\footnotestyle{mpmark=stars}` paranccsal.

`\editorfootnote{<lábjegyzet-szöveg>} ∈ magyar.ldf`

A rendes lábjegyzetek közé egy `\footnotestyle{starplain}` stílusban, csillagokkal, oldalanként újrazekódó számozású lábjegyzetet helyez el. Hatása megegyezik egy csoporton belül `\footnotestyle{editor}` után kiadott `\footnote`-éval. Jól használható a szerkesztő megjegyzéseinek feltüntetésére, miközben a szerző jegyzetei sima `\footnote` paranccsal szerepelnek. Például:

<pre>\footnotestyle{huplain} α\footnote{A szerzőtől.} β $\footnote{Ez is.}$ γ\editorfootnote{Túlzás. -- \textit{A szerk.}} δ\footnote{A szerző is.} ω\footnotestyle{editor}% \footnote{A szerkesztőtől.}}</pre>	$\alpha^1 \beta^2 \gamma^* \delta^3 \omega^{**}$ ¹ A szerzőtől. ² Ez is. [*] Túlzás. – <i>A szerk.</i> ³ A szerző is. ^{**} A szerkesztőtől.
---	---

`\setlength{\skip\footins}{<nyúló távolság>} \bigskipamount`
`\setlength{\dimen\footins}{<maximális magasság>} 8 in`

A főszöveg és a lábjegyzettömb (nyúlni képes) távolságát, illetve a lábjegyzettömb maximális magasságát állítja be. A keret a \LaTeX kernel-beli alapértékeiket mutatja, de ezt az osztály és a betöltött csomagok felülírhatják.

`\headingfootnote[[· ·]] {<lábjegyzet-szöveg>} ∈ magyar.ldf`

Fejezetcímekben a lábjegyzetek használata nem csak tipográfiai szempontból kerülendő, hanem azért is, mert például a `\subsection{Fent\footnote{Lent.}}` le se fordul, mert a `\subsection` parancs argumentuma az `.aux` file-ba íródik, és mivel a `\footnote` parancs nem „*robust*”, ezért ő nem írható ki. Ha feltétlenül szükséges a egy címben lábjegyzetet használni, akkor a `\protect\footnote`-ot kell írni `\footnote` helyett. Ekkor azonban a tartalomjegyzékben (`\tableofcontents`) meg fognak ismétlődni a lábjegyzetek, és – ami még rosszabb – pl. `book` és `report` stílus esetén az oldalfejlécben levő fejezetcím miatt minden oldalon megismétlődik a lábjegyzet, ráadásul igen valószínű, hogy rossz stílusban, mivel amikor az oldalfejléc generálódik, a \TeX általában már a következő oldal közepén tart, tehát az ottani stílust fogja figyelembe venni.

A tartalomjegyzék és az oldalfejléc problémáit megoldja, ha fejezetcímekben a `\footnote` helyett a `\headingfootnote` parancsot használjuk, például `\section{Cím\titlefootnote{Lábjegyzet.}}`.

2.19. Ami a magyar.ldf-ből kimaradt

A `magyar.ldf` nem felel meg az alábbi magyar tipográfiai előírásoknak:

- A listák fölött és alatt nem szabad függőleges helyet kihagyni. Ez megvalósítható a `\partopsep=0pt \topsep=0pt \parsep=0pt` értékadásokkal. Nagyon kevesen alkalmazzák ezt a szabályt.
- A listaelemek között nem szabad függőleges helyet kihagyni. Ez megvalósítható az `\itemsep=0pt` értékadással. Az egy listelemezen belüli bekezdések közötti távolság `\parsep`, ezt is érdemes lenulláznia.
- A definíciós lista (`description`) környezet címkéje után pontot, gondolatjelet, vagy a `postpara=` opcióban felsorolt szimbólumok egyikét kell tenni, a címke semmiképp sem maradhat lezáratlanul. Ez megvalósítható a `\descriptionlabel` felüldefiniálásával, például `\def\descriptionlabel#1{\hskip\labelsep\lrcorner\normalfont\bfseries #1.}`. Itt a `\hskip\labelsep`-re azért van szükség, mert a `@list` belső LaTeX parancs a címke szedése előtt `\labelsep`-nyit balra megy, utána pedig `\labelsep`-nyit jobbra. A `description` esetén `\labelwidth=0pt`, tehát a címke jobbra belelóghat a szövegbe. A `\descriptionlabel` után még `\labelsep` kerül kihagyásra. Ez megelevezhető a `\descriptionlabel` definíciójának végére helyezett `\hskip-\labelsep`-pel.
- A számozatlan listák címkéi a gondolatjeltől eltekintve csökkentett méretűek és emelt pozícióban vannak. Például:
!!
- add their itemize symbols and enumerate styles
- Ha egy listakörnyezet elemei egysorosak, akkor a *címkék* és a bal margó távolsága $i \times \text{\parindent}$ legyen ($i = 1, 2, \dots, 5$), még ha az elemek többsoros bekezdések, akkor ugyanekkora legyen bal margó és az *bekezdés szövegének* távolsága. A listabekezdések további sorai általában az első sor alatt kezdődnek, de megengedett az is, hogy a bal margón kezdődjenek, bár ez a gyakorlat manapság nem elterjedt. A címkék és a szöveg távolsága (`\labelsep`) $1/3$ em legyen. Ezeket a távolságokat a szerző a `\leftmargini` és társai, továbbá a `\labelsep` kétszeri felüldefiniálásával beállíthatja magának.
- A fejezetcímek előtt és után kevesebb üres helyet kell kihagyni függőlegesen, mint a LaTeX alapértelmezés. Ehhez minden osztályban a `@part`, `@spart`, `@makechapterhead`, `@makeschapterhead`, `section`, `subsection` stb. parancsokat át kéne definiálni. Ezt általánosan nehéz megoldani, de minden szerző megteheti, hogy saját osztályt definiál magának az egyik alaposztályból kiindulva, és ott átírja ezeket a parancsokat.
- A fejezetcímek formázásakor nagy szabadsága van a szerzőnek. Lehet álló vagy dőlt, csupa nagybetűs vagy kiskapitális, de még a félkövér és a félkövér dőlt betű is megengedett, bár nem ajánlott. A `magyar.ldf` a szerzőre bízva ezeken formázásokat, melyeket legkényelmesebben az osztályban a fent felsorolt parancsok átírásával lehet megtenni. A `magyar.ldf` biztosítja, hogy a fejezetszám után legyen pont, de a szerzőnek kell arra figyelnie, hogy a fejezetcím végén ne legyen.
- A LaTeX félkövér (*bold*) betűje alaphoz széthúzott (*extended*). A `\bf` parancs a `\normalfont\bfseries`-t hívja, a `\bfseries` pedig a `\fontseries{\bfdefault}\selectfont`-ot, a `\bdefault` makró értéke pedig `bx`. Ha ezt átírjuk `b`-re (például a preambulumban kiadott `\def\bfdefault{b}` parancssal), akkor **félkövér széthúzott** betűk helyett **félkövér** betűk kerülnek a dokumentumba. Sajnos azonban nem minden fontcsaládban van félkövér nem széthúzott betű, és a LaTeX nem elég okos ahhoz, hogy figyelmeztetés nélkül `bx`-et alkalmazzon, ha a `b` nem elérhető. Épp ezért a fenti átdefiniálás a szerzőre van bízva, és neki kell ügyelnie arra, hogy csak létező fontokat használjon.
A magyar tipográfia se a főszövegen belül, se a fejezetcímekben nem engedi a félkövér széthúzott betű használatát. Az egyedüli megengedett helyek a címoldalok (könyv-, rész- vagy fejezetcím, ha főszöveg nincs az oldalon).
- Döntött (*slanted*) betűket nem szabad használni, helyettük a dőlt (*italic*) ajánlott. A `magyar.ldf` `theoremtitle=hu` opciója a tételszerű környezetek törzsét dőlt betűkkel írja, de minden másra a szerzőnek kell figyelnie.
- Soregyent (*baseline grid*) kell alkalmazni, vagyis – ha csak lehetséges – a szomszédos soroknak pontosan `\baselineskip`-nyi távolságra kell lenniük egymástól. Ez különösen kiemelt matematikai képleteknél és főszövegen belül nagyméretű törteknél (például $\frac{n^2}{8}$) nehezen megoldható. A LaTeX függőleges távolságainak nagy része nyújtható, és ha a soregyen miatt ezt a nyújthatóságot megszüntetjük, akkor a LaTeX gyakran fog `Underfull \vbox` hibüzenetet adni, mivel nem tud saját mércéi szerint izlésesen oldalt törni. Végláthatatlan munka lenne az összes csomagban definiált összes távolságot a soregyen miatt felkutatni és átírni. A szerzők itt magukra vannak utalva, mert egy konkrét dokumentumon belül kisebb-nagyobb fáradtsággal biztosítható.
- Az üres oldalakon oldalszámnak sem szabad szerepelnie. A `magyar.ldf` nem figyel erre, például `\newpage-\newpage article.cls` esetén egy üres oldalt hoz létre oldalszámmal.

- A `\baselineskip`-hez kapcsolódóan a `\parskip`-et nullára kell állítani a postani `0pt plus 1pt` helyett. Ezalól csak akkor szabad kivételt tenni, ha egy rövid, beütés nélküli (`\parindent=0pt`) dokumentumról, például prospektus, van szó.
- Egyoszlopos dokumentumnál a bekezdés első sorának beütését (`\parindent`) a tükörszélesség és a betűméret alapján kell megállapítani. Ha `\textwidth < 24cc` (más források szerint `20cc`), akkor `\parindent=1em`-et, egyébként `\parindent=2em`-et kell beállítani. A `\parindent=0pt` beállítás csak rövid dokumentumoknál megengedett. A `\parindent` egész számú többszöröseit kell alkalmazni a többszintű listák és a lábjegyzet bal margójának megállapításakor.
- Egy bekezdés utolsó sorának (*break line*) hossza `\parindent > 0pt` esetén `\parindent` és `\hspace - \parindent` közé kell eszen, vagy pontosan `\hspace` kell, hogy legyen, azaz az utolsó sort is ki kell vinni a margóig. `\parindent = 0pt` esetén csak az utolsó sor csak `\parskip > 0,pt` mellett érhet a jobb margóig. Ha az utolsó sor nem éri el a jobb margót, akkor hossza sorkizárt bekezdések esetén legfeljebb `\hspace - 2em` (más források szerint énnél még kevesebb) lehet, balra igazított bekezdések esetén pedig legfeljebb `3/4 \hspace`. Az utolsó sor legalább `1em` (más források szerint `2em`) széles legyen. Donald Arseneau az alábbi megoldást javasolta a `\parindent > 0pt` esetre, de figyelmeztett, hogy ez valószínűleg összeakad azzal, ahogy a \LaTeX önmaga átdefiniálja a `\par`-t, és nem világos, hogyan kéne használni a `\@setpar`-t és a `\@restorepar`-t, hogy az összeakadást elkerüljük.

```

\def\par{\ifhmode
  \unskip \strut \hskip-\parindent \vadjust{\nobreak \hskip2\parindent
  \vrule depth 54321sp width 0sp \endgraf
  \ifdim\prevdepth=54321sp
    \nobreak \vskip-2\baselineskip \hbox{\strut}%
  \fi \fi}
\parfillskip=\hspace minus\hspace
\advance\parfillskip -2\parindent minus-2\parindent

```

- !! gy ligatúra

2.20. A magyar.ldf szerzői

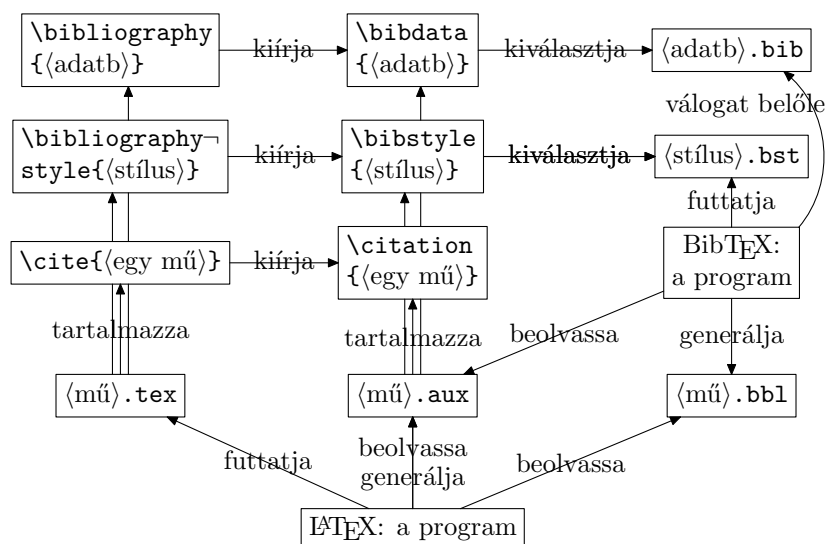
- Johannes L. Braams, \TeX niek: a Babel szerzője
- Bíró Árpád
- Bérces József: 1.4-es verzió
- Szabó Péter: 1.5-ös verzió, 2003 október eleje óta
- Wettl Ferenc, Bujdosó Gyöngyi: kutatás, hogy mi kerüljön bele az 1.5-ös verzióba
- Donald Arseneau: ötletek az 1.5-ös verzióhoz

3. Magyar irodalmi hivatkozások

Ha a szerző jól ismeri az irodalomjegyzék (bibliográfia) tördelésére vonatkozó magyar szabványokat, és a megírt irodalomjegyzéket nem kívánja más dokumentumokban felhasználni, akkor a `thebibliography` környezet kézi használata javasolt (leírását lásd bármelyik \LaTeX -ről szóló könyvben). Általában viszont érdemes igénybe venni valamely bibliográfiai adatbázis-kezelő segítségét, amely a következő előnyökkel jár: az adatbázis megosztása több dokumentum között; csak a ténylegesen használt irodalom kigyűjtése; a tételek (bejegyzések) automatikus ábécésorrendbe-rendezése; a tipográfiai szabályok automatikusan helyes alkalmazása.

A \LaTeX -hez kapcsolódó szokásos bibliográfiai adatbázis-kezelő, a `BibTeX` ún. stílusfile-okkal tetszőlegesen testre szabható. A magyar szabályokat alkalmazó stílusfile (`huplain.bst`) a `Magyar \LaTeX` része. A `BibTeX` igen régi: legfrissebb változatát (0.99c) 1988-ban írták, és azóta számos öt kiváltó megoldás született, melyek a többnyelvű bibliográfiák generálását is támogatják (pl. `MLBibTeX`). Mi mégis az őskövületnek számító a `BibTeX`-hel foglalkozunk, mert jól dokumentált, része minden \TeX disztribúciónak, és a legtöbb szerző igényeit kielégíti. Legfontosabb hátránya, hogy a többnyelvű irodalomjegyzékek kezelése nehézkes, különösen akkor, ha a dokumentum nyelve nem egyezik meg a hivatkozott mű nyelvvel. A `BibTeX` és a \LaTeX adatcseréjét a 3. ábra foglalja össze.

A `huplain.bst` a szabványos `plain.bst` magyarított változata. A magyarítás során fontos szempont volt, hogy ugyanazt az adatbázist lehessen használni a `huplain.bst`-vel magyar nyelvű, `plain.bst`-vel pedig angol nyelvű dokumentumokban. Ez csak úgy valósulhat meg, ha a szerző ennek tudatában készíti el a bibliográfiai adatbázist. Az egyszerűség kedvéért a `huplain.bst` minden bejegyzést a dokumentum nyelvéhez igazít, függetlenül a bejegyzés nyelvével, tehát egy könyv szerkesztőjét mindig „(szerk.)” jelöli, és sosem pl. „, editor”, még akkor sem, ha



3. ábra. A BibTeX és a L^AT_EX adatcseréje

a szerkesztő és a könyv is angol. A másik megoldást, nevezetesen, hogy minden tétel az ő saját nyelvén kerül szedésre, sokkal fáradtságosabb lett volna megvalósítani a BibTeX eszköztárával. A `huplain.bst` fontos tulajdonsága, hogy az ábécésorrendbe rakást az ékezetek és kettős mássalhangzók figyelembe vétele nélkül végzi. Ennél korrektebb rendezést fáradtságos lett volna leprogramozni.

3.1. Példa a BibTeX és a `huplain.bst` használatára

A BibTeX adatbázis egy `.bib` kiterjesztésű szövegfájl, amely bibliográfiai tételekből (bejegyzésekből) áll. Bejegyzésekre számos példát láthatunk a MagyarL^AT_EX részét képező `magyar-minta.bib` fájlban. Minden tétel mezők (*field*) listája. Minden mezőnek van egy (szabványos) neve, és egy értéke. A jelen dokumentumot leíró tétel például így fest:

```

1 @MANUAL{MagyarLaTeX,
2   inputenc = {latin2},
3   title = {Magyar nyelvű szöveg szedése {\sf Magyar\LaTeX}-hel},
4   author = {Szabó, Péter}, huname=1,
5   year = 2003, month = nov,
6   address = {Budaörs},
7   url = {http://www.math.bme.hu/latex/magyarlatex-doc.pdf}
8 }
```

Ha a fenti tételt elmentjük egy `adatb.bib` kiterjesztésű fájlba, akkor például így használhatjuk:

```

1 % bibproba.tex
2 \documentclass{article}
3 %\let\bibEarlyBegin\hfuzz \InputIfFileExists{\jobname.bbl}{-}{-}
4 \begin{document}
5 ...
6 Olvasd el \cite{MagyarLaTeX,MagyarLaTeX}-t!
7 ...
8 %\nocite{fmts}
9 %\def\bibOverride{\def\bibEtAl##1{ et al.}}
10 \bibliographystyle{huplain}
11 \bibliography{adatb}
12 \end{document}
```

A fenti L^AT_EX dokumentumot először a `latex bibproba` paranccsal fordítsuk le, majd futtassuk a `bibtex bibproba` parancsot, ami létrehozza a `bibproba.bbl` fájlt, melyben a dokumentum rendezve kiszedett irodalomjegyzéke van. Ezután a `latex bibproba` parancsot kétszer kell futtatnunk, hogy eltűnjenek a L^AT_EX figyelmeztetések. A két futtatás helyett elég egy is, ha a `huplain.bst`-t használjuk, és a dokumentumból eltávolítjuk a százalékjelű `\bibEarlyBegin`-t tartalmazó sor elejétől.

3.2. Tételtípusok és mezők

A `huplain.bst` a magyar tipográfia szerint szedi ki az elődje által támogatott összes `plain.bst` által definiált tételtípust (*entry type*), nevezetesen:

- article.** Egy folyóiratban (journal) vagy magazinban megjelent cikk. Szükséges mezők: `author`, `title`, `journal`, `year`. Kiegészítő mezők: `volume`, `number`, `pages`, `month`, `note`, `numpages`, `issn`, `url`, `tetex-url`, `ctan-url`.
- book.** Egy kiadó (publisher) által kiadott könyv. Szükséges mezők: `author` vagy `editor`, `title`, `publisher`, `year`. Kiegészítő mezők: `author2`, `volume` vagy `number`, `numvolumes`, `series`, `address` (városnév), `edition`, `month`, `note`, `numpages`, `isbn`, `url`, `tetex-url`, `ctan-url`.
- inbook.** Egy kiadó által kiadott könyv egy fejezete vagy egyéb része. Szükséges mezők: `author` vagy `editor`, `title`, `chapter` és/vagy `pages`, `publisher`, `year`. Kiegészítő mezők: `type`, `author2`, `volume` vagy `number`, `numvolumes`, `series`, `address` (városnév), `edition`, `month`, `note`, `numpages`, `isbn`, `url`, `tetex-url`, `ctan-url`. A rész típusát a `type` mező adja meg, például `type={fejezet}`. A rész címét pedig a `chapter` mezőben kell feltüntetni. E mezőkről további részleteket lásd később.
- incollection.** Egy kiadó által kiadott gyűjteményes mű egy része. A gyűjteményes művek jellemzője, hogy van szerkesztőjük (aki az egész műért felel), és minden résznek megvan a maga szerzője. Szükséges mezők: `author`, `title`, `booktitle`, `publisher`, `year`. Kiegészítő mezők: `editor`, `author2`, `volume` vagy `number`, `numvolumes`, `series`, `type`, `chapter`, `address` (városnév), `edition`, `month`, `note`, `numpages`, `isbn`, `url`, `tetex-url`, `ctan-url`. A `type` és `chapter` mezők értelmezését lásd az `inbook` tételtípusnál. Ha a rész szerzője és a mű szerkesztője megegyezik, ez `editor={self}` megadásával jelölendő (csak `huplain.sty`-nal).
- booklet.** Nyomtatott és kötött kiadvány; kiadó és támogató intézmény nélkül. Szükséges mező: `title`. Kiegészítő mezők: `author2`, `volume` vagy `number`, `numvolumes`, `series`, `address` (városnév), `edition`, `year`, `month`, `note`, `numpages`, `isbn`, `url`, `tetex-url`, `ctan-url`.
- manual.** Technikai dokumentáció vagy szabvány, melynek nincs kiadója, és általában szerző sincs feltüntetve. Szükséges mező: `title`. Kiegészítő mezők: `author2`, `author`, `organization` (pl. a szabvány kódneve), `address`, `edition`, `month`, `year`, `note`, `numpages`, `url`, `tetex-url`, `ctan-url`.
- unpublished.** Egy szerzővel és címmel bíró dokumentum, melyet még formálisan nem publikáltak. Szükséges mezők: `author`, `title`, `note`, `numpages`. Kiegészítő mezők: `month`, `year`, `url`, `tetex-url`, `ctan-url`.
- mastersthesis.** Doktori vagy egyéb tudományos cím megszerzéséhez kapcsolódó értekezés, kiemelés (`\emph`) nélküli címmel. Szükséges mezők: `author`, `title`, `school`, `year`. Kiegészítő mezők: `author2`, `type` (alapértelmezett értéke „Doktori értekezés”, felülbírállható: pl. `type={Kandid{\a}tusi {\e}rtekez{\e}s}`), `address` (lehetőleg városnév legyen, és ne a `school`-on belüli intézménynév), `note`, `numpages`, `url`, `tetex-url`, `ctan-url`.
- phdthesis.** PhD vagy egyéb tudományos cím megszerzéséhez kapcsolódó értekezés, kiemelt (`\emph`) címmel. A `type` mező alapértéke „PhD értekezés”. Minden egyébben megegyezik a `mastersthesis` tételtípussal.
- techreport.** Egy oktatási vagy egyéb intézmény által kiadott jelentés, általában sorszámozott. Szükséges mezők: `author`, `title`, `institution`, `year`. Kiegészítő mezők: `author2`, `type`, `number`, `address`, `month`, `note`, `numpages`, `url`, `tetex-url`, `ctan-url`.
- proceedings.** Egy konferencia teljes írásos anyaga, az előadásokhoz tartozó cikkek gyűjteménye. Szükséges mezők: `title`, `year`. Kiegészítő mezők: `editor`, `volume` vagy `number`, `series`, `address`, `month`, `organization`, `publisher`, `isbn`, `issn`, `url`, `tetex-url`, `ctan-url`, `numpages`, `note`.
- inproceedings (= conference).** Egy konferencián elhangzott előadáshoz tartozó cikk. Szükséges mezők: `author`, `booktitle`, `title`, `year`. Kiegészítő mezők: `pages`, `editor`, `volume` vagy `number`, `series`, `address`, `month`, `organization`, `publisher`, `isbn`, `issn`, `numpages`, `note`, `url`, `tetex-url`, `ctan-url`.
A *conference* csak kompatibilitási okokból maradt meg, helyette az *inproceedings* használata javasolt.
- misc.** Ha a többi tételtípus egyike sem illik a dokumentumra. Nincs szükséges mező. Kiegészítő mezők: `author`, `title`, `howpublished`, `month`, `year`, `note`, `numpages`, `isbn`, `issn`, `url`, `tetex-url`, `ctan-url`.

A BibTeX sajnos szó nélkül figyelmen kívül hagyja az adatbázisban szereplő, ismeretlen nevű mezőket. Tehát például ha egy könyv kiadási évének mezőnevét elgépeljük (pl. `yaer=2003`), akkor az évszám nem fog megjelenni a dokumentumban, és a korrektornak igencsak résen kell lennie, hogy észrevegye a hiányt.

Az alábbi kiegészítő mezőket, melyek nem jelennek meg, de befolyással vannak egyéb mezőkre, minden típusban használhatjuk:

inputenc. Elképzelhető, hogy bizonyos mezőértékek ékezetes betűket tartalmaznak. Az `inputenc` mező használata mellett az ékezetes betűk egyetlen karakterrel bevihetők. A `inputenc` csak a `huplain.sty`-ban működik, ezért az őt használó adatbázis nem hordozható. Az ékezetes betűk begépelésére a BibTeX az alábbi szabványos formátumot ajánlja: `{\’e}` és `{\H o}`. A kapcsos zárójeleket a BibTeX felismeri, így helyesen

ì {\`i}	í {\'i}	î {\^i}	ĩ {\~i}
ī {\=ai}	ï {\.i}	ï {"i}	ı {\c i}
ÿ {\u\i}	ÿ {\v\i}	ı̇ {\H\i}	ı̈ {\d i}
ı̄ {\b i}	ı̇ {\t\i\i}	oo {\t oo}	ı̈ {\r\i}
œ {\oe}	Œ {\OE}	æ {\ae}	Æ {\AE}
å {\aa}	Å {\AA}		
ø {\o}	Ø {\O}	ı̄ {\l}	Ł {\L}
ı̇ {\i}	ı̇ {\j}	ı̇ !'	ı̇ ?'
þ {\BN th}	Þ {\BN TH}	ı̇ {\k i}	
đ {\BN dh}	Đ {\BN DH}	đ {\BN dj}	Đ {\BN DJ}
ı̇ {\BN ng}	Đ {\BN NG}	ß {\ss}	SS {\BN SS}

2. táblázat. A BibTeX ékezetei és különleges betűi

rendez és kisbetűsít ékezetes betűket is. A formátum hátránya, hogy megtöri a T_EX ligatúrákat, épp ezért a `huplain.bst` a kellő időben eltávolítja az ékezetes betűket körülvevő kapcsos zárójeleket.

Az `inputenc` a bejegyzés bemeneti karakterkészletét adja meg, az `inputenc.sty`-nek adandó paraméter formájában. Csak akkor szerepeltessük, ha az ékezetes betűket egyetlen karakterként visszük be a tételben. Tipikus értéke magyar bejegyzéseknél `latin2`, nyugat-európai bejegyzéseknél `latin1`. A bejegyzésenkénti karakterkészlet-váltás érezhetően lassítja a L^AT_EX-et, ezért nem érdemes a preambulum `\usepackage~[...]{inputenc}` sorában megadott készlettel eltérőt megadni. Ha az ékezetes betűket több karakterrel, a 2. táblázatban megadott módon visszük be, akkor nincs szükség az `inputenc` mezőre. Az egyetlen karakterből álló ékezetes betűkhöz pedig kapcsos zárójelre nincs szükség.

Érdekes összevetnünk a fenti táblázatot a L^AT_EX ékezetes betűit leíró 1 táblázattal. A legfontosabb különbség, hogy a kapcsos zárójeleket máshová kell helyezni. Ennek az az oka, hogy a BibTeX a `{\langleparancsnév\rangle...}` konstrukciót egyetlen betűnek kezeli. Ennek hatása cíkcímek kisbetűsítésekor ("`t`" `change.case$`), keresztnév kezdőbetűjének kinyerésekor ("`ff.`" `format.name$`) és az ékezetek sorbarendezés előtti eltörlésekor (`purify$`) is megmutatkozik. a 2 táblázatban szereplő `\BN` parancs egy ravasz megoldás arra, hogy – ellentétben például a `\ss` betűvel – a BibTeX a `\dj` parancsot nem ismeri. Ha a `\BN`-t nem a `huplain.bst`-vel használjuk, el kell helyeznünk az alábbi sorokat az adatbázisban:

```
@preamble{ "\def\BN{\expandafter\BNc\expandafter{\iffalse}\fi} "
# "\def\BNc#1{\iffalse{\fi\curname#1\endcurname}} " }
```

Az `inputenc` a tételek rendezési sorrendjére is hatással van. Ha nem adunk meg `inputenc`-et, és egykarakteres ékezetes betűket használunk, a BibTeX ki fogja őket hagyni a rendezésből. Ha viszont megadtunk `inputenc`-et (és az `latin1` vagy `latin2`), vagy az ékezetes betűket a fenti formában, kapcsos zárójelek között visszük be, akkor a `huplain.bst` ékezetek nélkül, de a betűk megtartásával fogja besorolni a tételt. A fenti kettőn kívül más kódolás használata ellenjavallt, mert csak e kettő betűről tudja a `huplain.bst` levenni az ékezetet a rendezéshez; más kódolás esetén olyan, mintha meg se adtunk volna `inputenc`-et.

nocheck. Ha értéke nem üres (ajánlott: 1), akkor kevesebb figyelmeztető üzenetet ír ki a BibTeX az adott tételről, például nem figyelmeztet arra, hogy *article* esetén nem adtuk meg `number`-t.

preseries. Ha értéke nem üres (ajánlott: 1), akkor a könyv sorozatcíme (`series`) a cím (`title`) elé kerül.

huname. A szerzők és a szerkesztők neveinek megjelenítését szabályozza. Ha nem adtuk meg, vagy értéke 0, akkor az angol névsorrendet követi (pl. „Jules Verne”); ha 1, akkor a magyart (pl. „Verne Gyula”); ha 2, akkor pedig a fordított angolt, vesszővel (pl. „Verne, Jules”); ha pedig 3, akkor a magyart, de rövidített keresztnévvel (pl. „Verne Gy.” – az adatbázisban a kettős mássalhangzó miatt így szerepel: `author={Verne, {\empty Gy}ula}`). Ajánlott az 1 illetve 2 érték megadása, attól függően, hogy magyar vagy külföldi az illető. E mező értékétől függetlenül az adatbázisban a szerző nevét angol sorrendben kell megadnunk, mint ahogy a példabejegyzésben is szerepelt.

Ha a rövidített keresztnév kettős mássalhangzóval kezdődik, akkor a kettős mássalhangzót így kell körülvenni: `{\empty<...>}`. A BibTeX a kapcsos zárójelekről tudja, hogy meddig tart a név első betűje. A `huplain.sty` átdefiniálja L^AT_EX `\empty` makróját, így kapcsos zárójelek nem törnek majd meg ligatúrákat.

key. A rendezéshez használatos kulcsot adja meg. Csak a `huplain.bst` sajátossága, hogy ha `key={.}`-ot írunk elő, akkor a dokumentum a címe (`title`) alapján lesz besorolva a szerzővel rendelkező dokumentumok elé, továbbá nem kapunk figyelmeztetést, ha nem adtuk meg szerzőt.

A szerző és szerkesztő nélküli művek `key={.}` nélkül címük (`title`) szerint kerülnek besorolásra (eltérően a `plain.bst`-től, ahol a lista elejére kerülnek, ömlesztett sorrendben), a szerzővel rendelkező művek közé. Ha nem szeretnénk figyelmeztetést kapni a szerző és szerkesztő hiányáról, szerepeltessük a `nocheck=1` bejegyzést.

crossref. A BibTeX mezőörökítő funkcióját valósítja meg: ha egy tétel tartalmazza a `crossref = {foo}` mezőt, akkor az összes hiányzó mezőjét a `foo` nevű bejegyzésben is megnézi a BibTeX. A `crossref` használata kerülendő, mert az alapfunkción kívül két mellékhatása is van: ha a szülő műre többször hivatkoznak, akkor ő maga is belekerül az irodalomjegyzékbe (lásd még a `bibtex -min-crossrefs=` parancssori opciót), továbbá a `huplain.bst` nem a magyar tipográfiának megfelelően reagál a `crossref-re`.

A fontosabb szabványos mezők jelentése:

author, editor. A szerzők neveit az `author`, és/vagy a szerkesztőkéit pedig az `editor` mezőben adjuk meg. Ha több nevet sorolunk fel a mezőben, akkor `and`-del válasszuk el őket. Ha nem soroljuk fel az összes nevet, akkor `and_others`-szel zárjuk a felsorolást. A neveket az adatbázisban mindig angol sorrendben, azaz `author={Jules Verne}` vagy `author={Verne, Gyula}` formában adjuk meg. Ezáltal az adatbázis magyar és angol nyelvű dokumentumokban változtatás nélkül felhasználható lesz. Egy minta bejegyzés: `authors={dr. Köpeczi, Béla and Jules Verne and Kossuth, Lajos and gróf Széchenyi, István and John von Neumann and Donald E. Knuth and Strunk, Jr., William and Knézy, ifj., Jenő and L. R. McColvin and Tinódi Lantos, Sebestyén and Molnár-Sáska, Balázs and Claude Lévi-Strauss and Kis, Péter Pál and Mary-Claire van Leunen and Paul Gerhard Hoel and G. Bernard Godfrey and II., János-Pál and {Szerencsejáték Rt.} and {Earl of} Traquair and Horace [pseud.] Hunt} and Verne, Gyula and others.` A BibTeX szükség esetén a névben levő szóközt nem törhető szóközre (~) cseréli, ezzel nem kell foglalkoznunk.

Lehetőség van a nevek megjelenítésének szabályozására. Ehhez a L^AT_EX dokumentumban egy `\nocite{f1}` hivatkozást kell szerepeltetni, az adatbázisban pedig egy ilyen bejegyzést elhelyezni: `@FMTS{f1, firstname.fmt={f.}}`. Ekkor a keresztnevek rövidítve jelennek majd meg. A vezetéknevek megjelenítése is szabályozható a `lastname.fmt` mező megadásával, de ezt nem szokás rövidíteni.

A fenti nevek rövidített alakjai `huname=1` mellett: dr. Köpeczi B., Verne J., Kossuth L., gróf Széchenyi I., von Neumann J., Knuth D. E., Jr. Strunk W., ifj. Knézy J., McColvin L. R., Tinódi Lantos S., Molnár-Sáska B., Lévi-Strauss C., Kis P. P., van Leunen M.-C., Hoel P. G., Godfrey G. B., II. J.-P., Szerencsejáték Rt., Traquair Earl of., [pseud.] Hunt H., Verne Gy.

Ugyanezen nevek rövidített alakjai `huname=0` mellett: B. dr. Köpeczi, J. Verne, L. Kossuth, I. gróf Széchenyi, J. von Neumann, D. E. Knuth, W. Strunk, Jr., J. Knézy, ifj., L. R. McColvin, S. Tinódi Lantos, B. Molnár-Sáska, C. Lévi-Strauss, P. P. Kis, M.-C. van Leunen, P. G. Hoel, G. B. Godfrey, J.-P. II., Szerencsejáték Rt., Earl of. Traquair, H. [pseud.] Hunt, Gy. Verne.

title. A cikkcímek (`@ARTICLE, title`) nagy kezdőbetűit (az elsőt kivéve) a `huplain.bst` kisbetűsíti. Ez ellen kapcsos zárójelekkel védekezhetünk, például `title = {Fejtő {Ferenc} és a szociáldemokrácia-},,` vagy `title = {The Not So Short Introduction to {\LaTeXe}}`. Elég, ha a nagy kezdőbetűt zárójelezzük, de a ligatúrák megtartása miatt érdemes az egész szót zárójelbe tenni. A kisbetűsítés helyesen működik ékezetes betűvel kezdődő címekre is.

year, month. A dokumentum megjelenésének dátumát a `year` és `month` mezőben adhatjuk meg. A hónapot a 12 konstansból válasszuk (`jan, feb, mar, apr, may, jun, jul, aug, sep, oct, nov` vagy `dec`). Ha a hónap napja is ismert, akkor azt angol tipográfiával, a hónap neve után fűzzük, például `month = "15~" # mar`. Ne tegyünk e mezők végére pontot. Az évszámot és a többi számértékű mezőt nem kötelező idézőjelbe vagy kapcsos zárójelek közé tenni.

volume, number, pages. A kötetszám (`volume`), sorozaton belüli sorszám (`number`), hivatkozott oldalszámok (`pages`) mezők egy számot vagy egy intervallumot (pl. 5-7) tartalmaznak. Az intervallum határait egy vagy két kötőjellel is elválaszthatjuk, a BibTeX mindenképpen gondolatjelet (-) csinál belőle. Ne tegyünk e mezők végére pontot.

edition. A kiadás sorszáma. Pontosan úgy írjuk, ahogy a művön szerepel: figyeljünk az arab és római számok, továbbá a kis- és nagybetűk közötti különbségre – de ha betűvel van kírva (pl. „tizedik kiadás”), akkor arab számmal írjuk (pl. `edition=10`). Ne tegyünk a végére pontot.

series. Itt adjuk meg a sorozat címét, ha a könyv egy több kötetből álló vagy számozott sorozatból van, de mi ebből csak egyetlen kötetre hivatkozunk. Ne vegyük fel a „sorozat” szót (pl. `series = {A kultúra világa}`), ezt a `huplain.bst` magától hozzáfűzi. Ekkor a `series` mező tartalma a `title` mögé kerül. Ha ehelyett elé szeretnénk írni, akkor vegyük fel a `preseries=1` értéket (csak `huplain.sty`-ban). Ajánlott a `series` mellett a `volume` vagy `number` mezők egyikét szerepeltetni, mellyel megadjuk, hogy a könyv a sorozat hányadik tagja.

Ha a teljes sorozatra hivatkozunk, és nem csak egy tagjára, akkor a `series, volume` és `number` mezőket ne szerepeltessük, helyettük a sorozatcímét a `title` mezőben, a sorozat kötetének számát pedig a `numvolumes` mezőben adjuk meg.

booktitle. *incollection* esetén a szerkesztett gyűjtemény címét adja meg, *inproceedings* esetén pedig vagy a konferenciaanyag címét, vagy a konferencia nevét, ha a mű elhangzott ugyan, de a konferenciaanyagban

nem szerepel. Ez utóbbi esetben szerepeltessük a `talked=1` bejegyzést, hogy a `huplain.bst` így jelenítse meg a tételt: „Elhangzott a ... konferencián”.

type. *mastersthesis* és *phdthesis* esetén az értekezés típusa, például `type = {kandid{\`a}tusi {\`e}rtekez-{\`e}s}`.

inbook és *incollecion* esetén a rész típusát adja meg. A `type` mező alapértéke „fejezet” (*chapter*). Az ettől eltérő értékeket angolul érdemes megadni: (*part* → rész, *volume* → kötet, *chapter* → fejezet, *section* → szakasz, *subsection* → alszakasz, *subsubsection* → alalszakasz, *paragraph* → bekezdés, *subparagraph* → albekezdés, *figure* → ábra, *table* → táblázat, hogy ugyanaz az adatbázis angol nyelvű stílusokkal is használható legyen. A `huplain.bst` a fent felsorolt angol kulcsszavakat magyarra fordítja, az ettől eltérő string-eket pedig változatlanul hagyja.

chapter. *inbook* és *incollecion* esetén a rész címét adja meg. Ha a rész többszintű, e sorok írója az mélyülő sorrendet javasolja, vesszővel elválasztva, például `chapter = {Veszélyforrások, Vírusok, Makróvírusok}`.

Tételtípustól függően használhatók az alábbi, a `plain.bst`-ből még hiányzó mezők:

ISBN. Könyv ISBN száma, esetleg végződhet X-re. Például: `ISBN = {964 379 227 4}`.

ISSN. Folyóirat vagy könyvsorozat ISSN száma.

URL. A tételhez tartozó webcím. Ajánlott a dokumentum preambulumban az `url.sty` betöltése, mert ellenkező esetben a `huplain.bst` a `\verb` parancsot használja, ami nem tudja eltörni a sort, ezért a hosszú URL-t a következő sorba rakja (és az előző sort balra igazítja). Például: `url = {http://www.dante.de/}`.

teTeX-URL. A tételhez tartozó cím a `teTeX` `TeX`-disztribúció fáájában. Általában egy DVI, esetleg egy PDF file neve, például `tetex-url = {doc/makeindex/makeindex.dvi}`.

CTAN-URL. A tételhez tartozó cím a CTAN `TeX`-archív (`http://www.ctan.org/`) fáájában. Általában egy DVI, esetleg egy PDF file neve, például `ctan-url = {indexing/makeindex/doc/makeindex.dvi}`.

numpages. A dokumentum terjedelme, utolsó számozott oldalának (vagy az öt követő, még a fejezet részét képező utolsó oldalnak) a sorszáma. Nem tévesztendő össze a `pages` mezővel, amely kijelöli, hogy a `numpages` hosszúságú műből mely oldalakra hivatkozunk (pl. `pages=23` vagy `pages={23--45}`).

numvolumes. Egy könyv köteteinek száma. Nem tévesztendő össze a `volume` mezővel, amely kijelöli, hogy a `numvolumes` db kötetből álló könyv mely kötetére hivatkozunk (pl. `volume=2`).

talked. Csak *inproceedings* esetén használható. A bejegyzésben „In” helyett „Elhangzott a” fog szerepelni.

author2. A mű elkészítésében közreműködő egyéb személyek, például `author2 = {Illusztrálta Kis Géza. A borítót tervezte Gipsz Jakab}`.

3.3. A kinézet módosítása

A tételek kinézetén (csak `huplain.bst` esetében) változtathatunk a `\bibOVERRIDE` parancs definiálásával, melyben egyéb `\bib...` makrókat módosíthatunk. A `\bibOVERRIDE` használatára láttunk példát a szakasz elején (ne felejtjük el eltávolítani a százalékjelet a sor elejéről). A `\bib...` makrók a következők: `\bibAnd`, `\bibEtAl`, `\bibEd`, `\bibNewBlock`, `\bibVolume`, `\bibTechRep`, `\bibInSelf`. A makrók eredeti értékét megnézhetjük a `BibTeX` által generált `.bbl` file-ban.

A `@preamble` speciális tételtípusban extra `LATeX`-definíciók adhatók meg, például:

```
@preamble{ "\def\xindy{\textsf{X}\r\i ndy}} "
          # "\def" # "\noopsort#1{}" }
@PREambLE{ "\def\printfirst#1#2{#1}" }
```

A fenti példa azt is mutatja, hogy több `@preamble` tétel is szerepelhet, továbbá hogy egy tételen belül a `#` operátor string-ek összefűzésére használható. (Az utóbbira már láttunk példát a `month` mező megadásakor: `month = "15~" # mar`). Az így kapott string-eket a `BibTeX` közvetlenül a `\begin{thebibliography}` után szűri be a `.bbl` file-ba. A `huplain.bst` kinézetet befolyásoló makrói csak ezután következnek, így ezek a `\bibOVERRIDE` átírásával definiálhatók felül, például: `@preamble { "\def\bibOVERRIDE{\def\bibEtAl#1{ et~al.}} " }` hatására „et al.” fog megjelenni „és mások” helyett, ha az `author` mező `\and\others`-re végződik. Ajánlott a `@preamble` string-jében minél gyakrabban szóközt hagyni, mert ha nincs szóköz, akkor kb. a 76. karakternél a `BibTeX` mindenképpen eltöri a sort, még akkor is, ha a sortörés egy parancsszót vág ketté(!).

4. Magyar tárgymutató készítése

A tárgymutatóba (*index*) kerülő kifejezéseket az összes előfordulási helyükön `\index{<kifejezés>}` paranccsal kell bevinni. A `\index` parancs nem jelenít meg szöveget, csak az `.idx` kiterjesztésű file-ba ír. Ebből a file-ból egy külső program fogja generálni az `.ind` file-t, ami a tárgymutató bejegyzéseit ábécé sorrendben, oldalszám szerint összesítve tartalmazza. A \LaTeX -et újra kell futtatni, hogy a tárgymutató végső formája, a `.ind` file tartalma belekerüljön a dokumentumba.

A használatos tárgymutató-rendező segédprogramok (*index processor*):

makeindex. Egy viszonylag egyszerű C program, 1987-es fejlesztés. Legfrissebb változata 1997-ben készült. Szerény a tudásszintje, viszont gyors és kevés erőforrást használ. A magyar ábécésorrendet nem ismeri, és nem is tanítható meg rá. A UNIX-os \TeX disztribúció része. Dokumentációja [3]-ban olvasható.

Xindy. 1997-es fejlesztés. Common LISP-ben írták, ezért számos operációs rendszeren nem elérhető (Win32, Linux i386, Solaris és számos egyéb változat van). Nem része a \TeX disztribúciónak. Linux alatt problémák lehetnek a letölthető binárisal, az újrafordítás pedig egyszerű UNIX felhasználónak szinte reménytelen. A Magyar \LaTeX honlapjáról letölthető egy minden Linux-on jól működő változat. A magyar ábécésorrendet és tipográfiai szabályokat nem ismeri, viszont programozható, így bármely nyelvre megtanítható. A Magyar \LaTeX részét képező a `magyar.xdy` (lásd még a 4.3. szakaszban) jó közelítéssel igazítja a magyar szabályokhoz. A Xindy óriási előnye a hajlékonyság, az extrém testreszabhatóság.

husort.pl. E sorok írója fejlesztette ki Perl nyelven, 2002-ben. Pontosan úgy kell használni, mint a `makeindex`-et. Sajnos egyáltalán nem konfigurálható, a magyar nyelv ismerete van beledrótozva. Fő funkciója, hogy magyar nyelvű cikkek, könyvek tárgymutatóját a magyar helyesírás és tipográfia szabályai szerint, automatikusan generálja. A Perl számos platformon elérhető (köztük Win32, Linux, Macintosh és szinte az összes UNIX), és ezáltal a `husort.pl` is futtatható ezeken a platformokon.

4.1. A husort.pl és a makeindex használata

A készülő magyar nyelvű \LaTeX dokumentum az alábbi struktúrájú legyen:

```
1 \documentclass...{...}
2 \usepackage{t1enc}% automatikus szöelválasztás
3 \usepackage[magyar]{babel}% magyar feliratok stb.
4 \usepackage[latin2]{inputenc}% ékezetes betű lehessen a .tex-ben
5 \usepackage{makeidx}\makeindex% makeindex mód
6 %{\makeatletter\newif\ifscan@allowed
7 % \gdef\pfill{, }\gdef\efill{\hfill\nopagebreak}}
8 % ^^^ 'makeindex -s gind' nek kéne
9
10 \begin{document}
11
12 ... szöveg, benne \index{...} bejegyzések...
13 \printindex% a \section*{Tárgymutató} is megjeleníti
14
15 \end{document}
```

Ha a forrásfile neve `mű.tex`, akkor a `latex` mű; `husort.pl` mű; `latex` mű parancsok egymás utáni kiadásával kell lefordítani minden változtatás után. Angol nyelvű szöveg szedésekor a `husort.pl` helyett a `makeindex` parancsot futtassuk.

A `husort.pl` telepítése abból áll, hogy egy olyan könyvtárba másoljuk, ahonnan kényelmesen el tudjuk indítani. Ez lehet az aktuális könyvtár, vagy a `makeindex` parancsot tartalmazó könyvtár. Az aktuális könyvtár esetén UNIX alatt ne feledkezzünk meg arról, hogy `./husort.pl` néven kell indítani.

Mind a két segédprogram elfogadja az alábbi parancssori kapcsolókat:

- `s_gind`. A kezdőbetűket alcímként kiemeli. A `makeindex` ilyenkor az `gind.ist` stílusfile-ban előírt módon generálja a tárgymutatót. A stílusfile-ban némi testreszabásra van lehetőség. Ha nem töltjük be a `doc.sty`-t (és nem a `husort.pl` `|_run` módjában dolgozunk), akkor a `\efill` parancsot és társait a fenti példában leírt módon kell definiálnunk a `\printindex` használata előtt. A `husort.pl` nem használ stílusfile-t, ő csak a `gind` (és `bbind`) opcióra reagál.
- `i`. A szabványos bemenetről olvas és a szabványos kimenetre ír.
- `q`. Csendesesen dolgozik, csak a hibaüzeneteket írja a konzolra (terminálra).
- `a`. Csak a `husort.pl`-ben használható. Az ékezetes betűket a lexikonoknak megfelelő sorrendbe rendezi (lásd az S4b. pontot lent), például `sas < sár`. Az alapértelmezés a `sár < sas` sorrendet írja elő.

- r. Ha egy bejegyzés több (≥ 3) egymás utáni oldalon is szerepel, akkor -r kapcsoló nélkül ezeket egy intervallumba összevonja (pl. „7–9”), a kapcsolóval viszont egyenként felsorolja („7, 8, 9”).
- o_□{filenév}. Ha a kimenetet nem a {<mű>}.ind-be szeretnénk megkapni, így adhatjuk meg a kimeneti file nevét.

Az \index bejegyzéseket az alábbi módon ajánlott elhelyezni:

<p>Az ifjú Gauss-nak\index{Gauss} öt percre se volt szüksége, hogy kiszámolja az $1+2+\cdots+100=5050$ összegét.</p>	<p> Az ifjú Gauss-nak öt percre se volt szüksége, hogy kiszámolja az $1 + 2 + \cdots + 100 = 5050$ összeget.</p>
---	---

Lehetőség van a husort.pl parancssori kapcsolóit a .tex file-ban megadni. Ezt az első közönséges \index parancs előtt kiadott \index{|run_husort.pl_□{kapcsolók}} hívással tehetjük meg. A kapcsolókat szóközzel kell egymástól elválasztani. A husort.pl ugyanígy arra is rávehető, hogy a teljes munkát egy másik programmal végeztesse el, például \index{|run_makeindex_□-r_□-s_□gind}. Ennek az indirekt hívásnak az az előnye, hogy általa a másik program parancssori kapcsolói, sőt az tárgymutató feldolgozását végző program neve is a .tex file-ban megadhatók.

A továbbiakhoz érdemes átolvasni az \index parancs paraméterének a makeindex által elvárt formátumát.

Általában nem lesz szükség a megjelenítés és a sorbarendezés különválasztására (\index{<sortend>@<kinézet>}), mert a husort.pl a legtöbb előforduló tárgymutató-bejegyzést a magyar szabályok szerinti jó sorrendbe rakja. A husort.pl a T1 fontkódolás összes betűjét ismeri, és a benne található ékezetes betűket az ékezet nélküli betűvel egy helyre sorolja, kivéve, ha két szó között csak így lehetne különbséget tenni.

Néhány, csak a husort.pl-re jellemző szekvencia, ami az \index{· · ·}-ben hasznos lehet:

- \~{} vagy \lowtilde{}. alacsony (az alapvonalhoz közeli) hullámvonalat szúr be, melynek ábécésorrendbeli értéke a felsőbb szintről jön. Pl. \index{idő!tér és \~{}} az idő-n belül tér és idő-ként sorolódik be, de a tárgymutatóban „tér és ~” jelenik meg.
- . fontos szóközt szúr be, amely az összehasonlításakor nem nyelődik el, hanem minden betűt megelőz.
- . fontos pont karaktert szúr be, amely az összehasonlításakor nem nyelődik el, hanem minden betűt megelőz, de a fontos szóközt nem.

A husort.pl oldalszámként az arab és római számokat ismeri. Egy bejegyzésen belül először a római, majd az arab, végül az egyéb oldalszámok következnek. Ez utóbbiak ugyanazzal a módszerrel kerülnek besorolásra, mint a bejegyzések (tehát pl. a kis- és nagybetűs „oldalszámok” keveredni fognak). L^AT_EX-ben az oldalszámozás stílusát (akár egy dokumentumban többször is) a \pagenumbering parancs segítségével állíthatjuk be. A megengedett paraméterek: arabic, roman, Roman, alph, Alph, és – ha a magyar.ldf-et a hunumbers= opcióval töltöttük be, akkor huordinal, Huordinal, hunumeral és Hunumeral.

Példa a különböző oldalszám-formátumok használatára:

```
\makeatletter\def\@alphb#1{ \@alph{#1}}      |szikla iii, d, 1, E, i
1\index{szikla}\newpage
\pagenumbering{roman}\setcounter{page}{3}
3\index{szikla}\newpage
\pagenumbering{alph} \setcounter{page}{4}
4\index{szikla}\newpage
\pagenumbering{Alph} \setcounter{page}{5}
5\index{szikla}\newpage
\pagenumbering{alphb}\setcounter{page}{9}
9\index{szikla}\newpage
```

A fenti példában a 4. oldal száma „d”, melyet a husort.pl – tévesen – római számnak érzékel, és emiatt az arab számok elé sorol. Az ez elleni védekezést mutatja a 9. oldal, melynek a száma („i”) szóközzel kezdődik az \@alphb sorszámozási stílus miatt, ezért a husort.pl nem ismeri fel római számként. Sajnos a makeindex az oldalszámokban figyelmen kívül hagyja a szóközöket, ezért ott az i megelőzi az 1-est.

4.2. A husort.pl és a makeindex közti főbb különbségek

- Csak a husort.pl támogatja a |_run módot.
- A makeindex nem veszi figyelembe az oldalszám elején a szóközt.
- A makeindex nem veszi figyelembe a -q kapcsolót, ha -i is adott.
- A makeindex kimenete speciális parancsok definiálását igényli, ha -s_□gind kapcsolóval hívtuk (és nem a husort.pl |_run módján keresztül).

- Ha a `makeindex`-et paraméter nélkül hívjuk, akkor azt az `-i` kapcsolónak veszi. A `husort.pl` a program sűgőt ír ki önmagáról a konzolra.
- A `\index{a}(c)`-t `husort.pl`-ben csak `\index{a|}c`-vel szabad lezárni; a `makeindex` megengedi a `\index{a|}` zárást is. Kompatibilitási okokból csak az előbbi használata ajánlott.
- A `makeindex` nem fogadja el ugyanannak a címszónak különböző stílusú nyitásait, például: `\index{a|(b)}` `\index{a|(c)}`.
- A `husort.pl` csak a `"", "@, "!` és `"|` szekvenciákból vágja le az idézőjelet, míg a `makeindex` az összes idézőjellel kezdődőből.
- A `husort.pl` általában a szabványos kimenetre naplóz, és csak `-i` esetén tér át a szabványos hibakimenetre, a `makeindex` viszont mindent a szabványos hibakimenetre ír.
- A `husort.pl` nem engedi meg a parancssori kapcsolók összevonását, tehát csak a `husort.pl -q -i` helyes, míg `makeindex -qi` is helyes.
- A `husort.pl` automatikusan hozzáteszi az `.idx` kiterjesztést a parancssorban megadott filenévhez. A `makeindex` pedig először a kiterjesztés nélküli file-t próbálja megnyitni, csak utána teszi hozzá az `.idx`-et.
- A `husort.pl` elfogadja a `-sgind` kapcsolót is, de csak a `makeindex -s gind` a helyes.
- Csak a `makeindex`-nek adható meg több bemeneti `.idx` file a parancssorban.
- Csak a `makeindex` készít munkájáról naplófile-t (`.ilg` kiterjesztéssel).
- Csak a `husort.pl` fogadja el a `\index{a|b|textbf}` bejegyzést (és írja ki a `\textbf{42}`-t sorszámnak). A `makeindex` visszautasítja a bejegyzést.
- A `husort.pl` nem ír ki figyelmeztetést („Conflicting entries: multiple encaps. .”), ha ugyanazon az oldalon `\index{a|textbf}` és `\index{a|textit}` is szerepel.
- Csak a `makeindex` támogatja a `-l, -c, -g, -t` és `-p` parancssori kapcsolókat.

4.3. Xindy

A stílusfájl szerkezete ♦ Az Xindy stílusfájl szerkezetét, a jelen könyvhöz is használt `magyar.xdy` file részletes magyarázatával mutatjuk meg. A stílusfájl lényegében két részre osztható. Az elsőben az index formátumát leíró utasításokat adjuk meg, a második részben pedig a nyelvre vonatkozó egyéb szabályokat, ABC-t, rendezést definiáljuk.

A következő parancsokkal a dokumentumban használt hivatkozási hely típusokat (különböző oldalszámok, szekciók) és ezek közti sorrendet adjuk meg.

```
(define-location-class "arabic-page-numbers" ("arabic-numbers"))
(define-location-class "roman-page-numbers" ("roman-numbers-lowercase"))
(define-location-class "Roman-page-numbers" ("roman-numbers-uppercase"))
(define-location-class "alpha-page-numbers" ("alpha"))
(define-location-class "Alpha-page-numbers" ("ALPHA"))

(define-location-class "sections" :var
  ("arabic-numbers" :sep "."
   "arabic-numbers" :sep "."
   "arabic-numbers"))

(define-location-class-order ("roman-page-numbers" "arabic-page-numbers"
  "alpha-page-numbers" "Roman-page-numbers" "Alpha-page-numbers" "sections" "see"))
```

Ezután definiálhatjuk a hivatkozások attribútumait és a hozzájuk tartozó megjelenítést, melyeket a \LaTeX dokumentumunkban a `\index{szó|attribútum}` formában adhatunk meg (például `\index{környezet|usage}`). Az attribútum nélküli elemek a default csoportba tartoznak.

```
(define-attributes (("default") ("usage")))
(markup-locoref :open "\textbf{" :close "}" :attr "usage" )
```

A \LaTeX fájlban az attribútumhoz hasonlóan adjuk meg az indexen belüli utalást vagy kereszthivatkozást és a hozzá tartozó megjelenítést.

```
(define-crossref-class "see")
(markup-crossref-list :open "\see{" :close "}" :class "see")
```

Az index részt megadó \LaTeX parancsok. A `n` a soremelés jele.

```
(markup-index :open "\begin{theindex}~n" :close "~n~n\end{theindex}" :tree)
```


Az indexcsoportokat elválasztó parancs.

```
(markup-letter-group-list :sep "~n~n \indexspace-n")
```

Az indexelemen belüli szintek definiálása.

```
(markup-indexentry :open "~n \item " :depth 0)
```

```
(markup-indexentry :open "~n \subitem " :depth 1)
```

```
(markup-indexentry :open "~n \subsubitem " :depth 2)
```

Oldalszámok listájának, intervallumának, illetve az indexen belüli szintek elválasztójelei.

```
(markup-locref-list :sep ", ")
```

```
(markup-range :sep "--")
```

```
(markup-locclass-list :open ", " :sep ", ")
```

Az indexcsoportokat jelölő kezdőbetűk formátuma, illetve a kezdőbetűvel nem rendelkező elemek (speciális karakterek) előtti *Default* szó kommentezése.

```
(markup-letter-group :open-head "~n \textbf{" :close-head "}" :capitalize)
```

```
(markup-letter-group :open-head "~n% " :close-head "" :group "default")
```

A stílusfájl második részében az ún. betűcsoportokat (*letter group*) definiáljuk, mely lényegében a nyelv ABC-jének megadását jelenti. Az első típusú megadásnál egyszerűen felsoroljuk a betűket. A felsorolás sorrendje megadja a csoportok közti sorrendet is.

```
(define-letter-groups
```

```
  ("b" "c" "d" "f" "g" "h" "j" "k" "l" "m"
```

```
   "n" "p" "q" "r" "s" "t" "v" "w" "x" "y" "z" ))
```

A második típusnál, a csoportokat egyesével adjuk meg. A `define-letter-group` utáni string jelenik meg az indexben, mint csoport azonosító, a `:before` és `:after` részekkel a csoport rendezését adjuk meg, a már korábban definiált csoportokhoz képest. A `:prefixes` adja meg a szavak besorolási feltételét, azaz az itt megadott szöveggel kezdődő szavak kerülnek az adott csoportba. A `e` olyan karaktert definiál, amely minden más karakter után következik.

```
(define-letter-group "cs" :after "c" :before "d" :prefixes ("c~e" ))
```

```
(define-letter-group "dz" :after "d" :before "f" :prefixes ("d~e" ))
```

```
(define-letter-group "dzs" :after "dz" :before "f" :prefixes ("d~e~e" ))
```

```
(define-letter-group "gy" :after "g" :before "h" :prefixes ("g~e" ))
```

```
(define-letter-group "ly" :after "l" :before "m" :prefixes ("l~e" ))
```

```
(define-letter-group "ny" :after "n" :before "p" :prefixes ("n~e" ))
```

```
(define-letter-group "sz" :after "s" :before "t" :prefixes ("s~e" ))
```

```
(define-letter-group "ty" :before "v" :after "t" :prefixes ("t~e" ))
```

```
(define-letter-group "zs" :after "z" :prefixes ("z~e" ))
```

```
(define-letter-group "a, á" :before "b" :prefixes ("a" ))
```

```
(define-letter-group "e, é" :before "f" :after "d" :prefixes ("e" ))
```

```
(define-letter-group "i, í" :before "j" :after "h" :prefixes ("i" ))
```

```
(define-letter-group "o, ó" :before "p" :after "ny" :prefixes ("o" ))
```

```
(define-letter-group "ö, ő" :before "p" :after "o, ó" :prefixes ("o~e" ))
```

```
(define-letter-group "u, ú" :before "v" :after "ty" :prefixes ("u" ))
```

```
(define-letter-group "ü, ű" :before "v" :after "u, ú" :prefixes ("u~e" ))
```

A következőkben az ún. *merge rule*-kat definiáljuk, melyek reguláris kifejezés alapú konverziós szabályok. Az Xindy a szabályokat minden egyes indexelemre alkalmazza, majd az előállt készlet alapján dől el, hogy mely elemek azonosak. Mi itt a T_EX-es ékezetes karaktereket konvertáljuk latin2 kódolásúvá. Az alábbiakban csak néhány elemet sorolunk fel, a többi hasonlóan elkészíthető, kivéve az *í* betűhöz tartozó szabály. A megadott formátumú *í* betű az `inputenc` csomag hatására íródik ki az `.idx` fájlba.

```
(merge-rule "\\\'[[:blank:]]*e" "é" )
```

```
(merge-rule "\\\'[[:blank:]]*{[[:blank:]]*e[[:blank:]]*}" "é" )
```

```
(merge-rule "\\IeC[[:blank:]]*{[[:blank:]]*\\\'[[:blank:]]*\\i[[:blank:]]*}" "í" )
```

```
(merge-rule "\\IeC[[:blank:]]*{\\\'{[[:blank:]]*\\i[[:blank:]]*}[[:blank:]]*}" "í" )
```

Mivel a kis- és nagybetűs szavakat nem különböztetjük meg, ezért ezeket a konverziókat is itt adjuk meg.

```
(merge-rule "A" "a")
```

```
(merge-rule "Á" "á")
```

```
...
(merge-rule "Z" "z")
```

A stílusfájl végén adjuk meg az ún. *sort rule*-okat, melyek a *merge rule*-okhoz hasonlóan reguláris kifejezés alapú konverziók, melyek az indexelemek rendezését és betűcsoportokba való besorolását határozza meg. Először bizonyos kivételszavakat adunk meg, melyekre nem szeretnénk alkalmazni a későbbi általános szabályokat. Ilyenek például bizonyos angol szavak, illetve az olyan összetett szavak, melyeknél az összetételnél kettős betű keletkezik, de azt külön ejtjük.

```
;; Enélkül a type a ty betűhöz kerülne
(sort-rule "type" "type")
(sort-rule "ttype" "ttype")
(sort-rule "family" "family")
```

```
(sort-rule "élcsoorozat" "élcsoorozat")
```

Ezután jönnek az általános szabályok. A kettőzött dupla mássalhangzókat úgy rendezzük, mintha a dupla mássalhangzót megkettőztük volna.

```
(sort-rule "ccs" "cscs" )
(sort-rule "ggy" "gygy" )
(sort-rule "lly" "lyly" )
(sort-rule "nny" "nyny" )
(sort-rule "ssz" "szsz" )
(sort-rule "tty" "tyty" )
(sort-rule "zsz" "zszs" )
```

A rövid és a hosszú magánhangzók között nincs rendezésszerű különbség.

```
(sort-rule "á" "a" )
(sort-rule "é" "e" )
(sort-rule "í" "i" )
(sort-rule "ó" "o" )
(sort-rule "ö" "o~e" )
(sort-rule "ő" "o~e" )
(sort-rule "ú" "u" )
(sort-rule "ü" "u~e" )
(sort-rule "ű" "u~e" )
```

Dupla mássalhangzók.

```
(sort-rule "cs" "c~e" )
(sort-rule "dz" "d~e" )
(sort-rule "dzs" "d~e~e" )
(sort-rule "gy" "g~e" )
(sort-rule "ly" "l~e" )
(sort-rule "ny" "n~e" )
(sort-rule "sz" "s~e" )
(sort-rule "ty" "t~e" )
(sort-rule "zs" "z~e" )
```

4.4. A Perl lecsupaszítása Windows-on

Windows alatt csak a `perl.exe` és a `perl5*.dll` szükséges a `husort.pl` futtatásához. Ezekhez úgy juthatunk hozzá, hogy letöltünk (pl. a <http://www.perl.org/>-ról) és telepítünk egy tetszőleges Windows-os Perl disztribúciót, kimásoljuk belőle ezt a két file-t, majd eltávolítjuk (*uninstall*) a disztribúciót a gépről. A `husort.pl`-t Windows-on így kell indítani: `perl -x husort.pl`. Ha olyan hibaüzenet érkezik, hogy a Windows nem találja az adott programot, akkor az `.exe`-t és a `.dll`-t másoljuk be a Windows SYSTEM könyvtárába (vagy vegyük fel az ő könyvtárukba a PATH környezeti változóba, vagy másoljuk őket az aktuális könyvtárba). Ha a program egyéb okokból nem indulna el, töröljük ki a `husort.pl` file-ból a `#!perl`-lel kezdődő sort megelőző sorokat.

4.5. A tárgymutatókészítés szabályai a magyar tipográfiában

A tárgymutató-tételek betűrendbe (ábécé-rendbe) sorolási szabályainak felállításakor a `husort.pl` szerzője [2]-re támaszkodott. Az alábbi szabálylistában „[OK]” jelzi a `husort.pl` által követett szabályokat, a szabályoktól való eltérést pedig [...] fejt ki.

- S0a. „szó”-nak nevezünk egy előre rögzített kódolású (ISO-8859-2 vagy T_EX-T1), tetszőleges (0..255) karaktereket tartalmazó, tetszőleges hosszú karaktersorozatot.
- S0b. Két szó azonos, ha egyenlő hosszúak, és azonos pozícióban azonos karakterek állnak. Jelölés: egyik = egyik. [OK]
- S0c. Ha két szó nem azonos, akkor egyértelműen szabályozva van, hogy melyik kerül előrébb az ábécében. Jelölés: egyik < másik. [OK]
- S1a. A magyar nagyábécé: a á b c cs d dz dzs e é f g gy h i í j k l ly m n ny o ó ö ő p q r s sz t ty u ú ü ú v w x y z zs [OK]
- S2a. A kis- és nagybetűk besorolási értéke azonos. Pl. kín < Kína < kínál < Kovács < kovács. [OK]
- S2b. Ha a két szó csak a kis- és nagybetűkben különbözik, akkor a nagybetűk előrébb kerülnek. Pl. Kovács < kovács. [OK]
- S3a. Ha a szavak azonos betűvel kezdődnek, a sorrendet az első nem azonos betű határozza meg. Pl. ha < habár < habitáció < habitus [OK]
- S3b. Ha a másik szó az egyikkel kezdődik, akkor az egyik kerül előre, pl. ha < habár [OK].
- S4a. A magánhangzók rövid és hosszú változatai azonos értékűek (de pl. u < ü): a = á = ä < e = é < i = í < o = ó = ô < ö = õ < u = ú < ü = ű. Azaz például á-val kezdődő szó megelőzhet a-val kezdődőt. Pl. ács < acsarkodik. [OK, mindig azonos értékűnek veszi, tehát nem a nyelvészeti munkák és ... kivételes esete valósul meg]
- S4b. A nyelvészeti munkákban, valamint a hosszabb terjedelmű lexikonokban a magánhangzók rövid és hosszú változatai különböző értékűek, pl. u < ú < ü < ű. [nem támogatja, helyette ld. S4a, ami gyakoribb igény]
- S4c. Ugyanez vonatkozik az egyéb ékezetekre (mellékjelekre) is (ha például idegen szavakat veszünk fel a mutatóba). Pl. áhitat < ählich < ahogy. [OK. T_EX-T1 esetén (ez az alapértelmezés) csak a kódolásban megtalálható betűkre, melyek az a..z betűk ékezetes változatai. ISO-8859-2 esetén csak a magyar ékezetes betűk és ‘ä’.]
- S5a. Az egyjegyű és kétjegyű (illetve háromjegyű) mássalhangzókat (c–cs, d–dz–dzs, g–gy, l–ly, n–ny, s–sz, t–ty, z–zs) nem két betűnek, hanem egyetlen, külön betűnek tekintjük. Pl. C < Curie < Czuczor < Cs < Csajkovszkij < Csehov. [OK, az idegen szavakban előforduló többjegyű mássalhangzókat is egynek]
- S5b. Ha latin betűs ábécére rendezzük a tárgyszavakat, akkor a sorrend a következő: C < Csajkovszkij < Csehov < Curie < Czuczor. [nem támogatja, helyette ld. S5a, ami gyakoribb igény]
- S6. A szavak belsejében vagy a szavak végén a kettőzött mássalhangzókat két külön betűnek tekintjük (pl. ggy = gy + gy). Pl. nagz < nagy < nagygyá < nagygyakorlat < nagygyából. [OK]
- S7. A régies írású magyar neveket, illetve az idegen neveket és szavakat nem kiejtés szerint, hanem úgy soroljuk betűrendbe, hogy a kétjegyű betűket felbontjuk (pl. ch=c+h, ew=e+w). A mellékjeles betűket sem a kiejtés szerint csoportosítjuk. Pl. Čapek < Černý < Chianti < Cholnoki. [OK]
- S8a. Az összetett szavakat és a külön- vagy kötőjellel írt szókapcsolatokat úgy soroljuk betűrendbe, mint az egyszerű szavakat. Pl. munkaerő < munkaerő-gazdálkodás < munkaerő nélküli < munkaerős. [OK]
- S8b. Ugyanez vonatkozik a nevekre is. Pl. Széchenyi-hegy < Széchenyi István < Széchenyi Könyvtár < Széchenyi tér. [OK]
- S8c. A három tagból álló nevek is akkor kereshetők vissza a leggyorsabban, ha ugyanezt a logikát alkalmazzuk. Pl. Csáth Géza < Cs. Szabó László < Nagy A. László < Nagy Albert < Nagy A. Márton < Nagy Sándor < Nagy Szabó István < Nagy Zoltán. [OK]
- S8d. Azoknál a neveknél azonban, amelyek írott formája eltér egymástól, már nem használható ez a szabály. Pl. Horvát Aladár < Horvát Károly < Horvát Zoltán < Horváth Imre < Horváth Péter [OK, ha a neveket elválasztó szóközöket a hívó külön megjelöli úgynevezett „fontos szóköz”-nek, azaz pl. ‘Horvát Zoltán’ helyett ‘Horvát{ }Zoltán’-t és ‘Horvát Imre’ helyett ‘Horváth{ }Imre’-t gépel]
- S9. A besoroláskor minden adatot figyelembe kell venni (évszám, egyéb kiegészítés). Pl. Kovács Pál < Kovács Pál{ }(1767–1812) < Kovács Pál{ }(1946–1994) < Kovács Pál fordító < Kovács Pál költő < Kovács Pál (1767–1812) < Kovács Pál (1946–1994). [OK]
- S10a. A mutatókban használt névelők nem részei a besorolási adatnak, éppen ezért névelőket csak akkor használjunk, ha az elkerülhetetlen (pl. címek). Pl. The Television < Television, The < televízió < televíziókészülék [nem ismeri fel a névelőket, LaTeX-ben külön pozicionálni kell, pl. \index{Television@The Television}]
- S10b. Irodalmi művek mutatójában (pl. betűrendes versmutató) viszont a névelők is számítanak. Pl. Ádventi köd < Ady < A festő halála < Alkalmi vers. [OK, amúgy is ez az alapértelmezés]
- S11–S21. [1] részletesen taglalja, hogy személyneveket milyen formában kell felvenni a tárgymutatóba. [Ezek a szabályok igen fontosak, de a nevek sorrendezését nem érintik.]

- S91. S4a., S8a. és S8b. közös általánosítása: whitespace (szóköz, tab stb.) karaktereket és az alábbi írásjeleket: ! " & ' () , - . / : ; ? [] _ { }, kihagyjuk az összehasonlítás során, és a magánhangzók rövid és hosszú változatait egyenértékűnek tekintjük, kivéve ha két nem azonos szó ily módon egyenlőnek találtatna. Az első esetben a sorrend: fontos szóköz < ASCII vezérlő < itt fel nem sorolt ASCII írásjel < szókezdő számjegy < betű < egyéb számjegy. A kivételes esetben a sorrend: fontos szóköz = whitespace < ASCII vezérlő < összes ASCII írásjel < szókezdő számjegy < betű < egyéb számjegy. A kis- és nagybetűk azonos értékűek mindkét esetben, továbbá az egyes írásjelek között sem tesszük különbséget. Ha két nem azonos szó még a második esetben is egyenlőnek bizonyulna, akkor a két karaktersorozatot binárisan hasonlítjuk össze. A használt kódolások (US-ASCII, ISO-8859-2 és T_EX-T1) közös tulajdonsága, hogy a bináris összehasonlításnál a kisbetűk megelőzik a nagybetűket. [OK]
- S92. Az előző pont folyamánya a `husort.pl` által követett Bab < bab < Báb < báb sorrend, mivel a kis- és nagybetűk különbsége már csak akkor számít, ha a kisbetűsített alakok megegyeznek. Egy másik sorrend lehetne: Bab < Báb < bab < báb. [OK]
- S93. A számjegyek azonos értékűek, kivéve ha emiatt két nem azonos szó egyenlőnek találtatna. Ezáltal a nemnegatív egész számok jól rendeződnek. Pl. $0 < 2 < 9 < 28 < 100$. A tizedestörtek és a negatív számok rendezése nem helyes. Pl. $2 < 9 < 2.7$. Pl. $-2 < -10$. [OK]

Hivatkozások

- [1] Gyurgyák János: *Szerkesztők és szerzők kézikönyve*. Osiris kézikönyvek sorozat. A főszöveg, Jegyzetek, hivatkozások, Tipográfia alszakasz. Budapest, 2000, Osiris Kiadó, 116–118. p. ISBN 964 379 227 4. ISSN 1416-6321. 541 p.
- [2] Gyurgyák János: *Szerkesztők és szerzők kézikönyve*. Osiris kézikönyvek sorozat. A főszöveget követő járulékos részek, Mutatók, A mutatók sorrendje alszakasz. Budapest, 2000, Osiris Kiadó, 161–169. p. ISBN 964 379 227 4. ISSN 1416-6321. 541 p.
- [3] Lamport L.: *MakeIndex: An Index Processor For L^AT_EX*. 1987. február 17.
CTAN `indexing/makeindex/doc/makeindex.dvi`. te_TE_X `doc/makeindex/makeindex.dvi`.
- [4] Oetiker T. – Partl H. – Hyna I. – Schlegl E.: *The Not So Short Introduction to L^AT_EX 2_ε*. 1999. január 8.
CTAN `info/lshort/english/lshort.dvi`. te_TE_X `doc/latex/general/lshort.dvi`.

Tartalomjegyzék

1. Telepítés	1
2. Kezdőlépések, a magyar.ldf	2
2.1. A Babel betöltése	2
2.2. Ékezetes betűk	3
2.3. Betöltési opciók használata	4
2.4. <code>\frenchspacing</code> – a pont utáni szóköz szélessége	8
2.5. Kiemelések, idézetek, megszólalások	9
2.6. Az aktív karakter	10
2.7. Címsorok	10
2.8. Automatikus szóelválasztás és a <code>huhyph.tex</code>	11
2.9. A sortörés irányítása	13
2.10. Matematikai képletek	14
2.11. A határozott névelő (a, az)	15
2.12. Betűvel írt számjegyek	16
2.13. Számok toldalékolása	16
2.14. Hivatkozás a dokumentum strukturális egységeire	17
2.15. Tételszerű környezetek címei	18
2.16. Dátumok	18
2.17. Ékezetes betűk hivatkozásokban	19
2.18. Lábjegyzetek	19
2.19. Ami a <code>magyar.ldf</code> -ből kimaradt	22
2.20. A <code>magyar.ldf</code> szerzői	24

3. Magyar irodalmi hivatkozások	24
3.1. Példa a BibTeX és a huplain.bst használatára	25
3.2. Tételtípusok és mezők	26
3.3. A kinézet módosítása	29
4. Magyar tárgymutató készítése	30
4.1. A husort.pl és a makeindex használata	30
4.2. A husort.pl és a makeindex közti főbb különbségek	31
4.3. Xindy	32
4.4. A Perl lecsupaszítása Windows-on	34
4.5. A tárgymutatókészítés szabályai a magyar tipográfiában	34