



## Začínáme s T<sub>E</sub>Xem na [fi | studovna].muni.cz

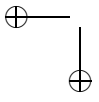
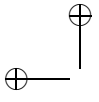
Tomáš Buk, Luboš Lipínský, Petr Sojka, září 2003

---

*Tento dokument se snaží pomoci těm z vás, kteří mají zájem začít používat instalaci sázecího systému T<sub>E</sub>X na fi.muni.cz příp. v celouniverzitní počítačové studovně (CPS) na LF (studovna.muni.cz). Taktéž může pomoci i těm, kteří hledají odpověď na některý specifický, s T<sub>E</sub>Xem související, problém. Naopak, tato brožura nemůže být chápána jako podrobný návod, jak pracovat s T<sub>E</sub>Xem pro ty, kteří ještě nemají s tímto systémem žádné zkušenosti. Těm je v brožuře věnován pro motivaci krátký úvod, v kterém je shrnuta historie a stručná charakteristika T<sub>E</sub>Xu.*

*Části textu, které se vztahují pouze k T<sub>E</sub>Xování na FI a netýkají se instalace v CPS jsou označeny svislými čarami na bočním okraji (zejména jsou takto označeny části popisující unixová specifika). V CPS je v době tisku příručky ještě instalace texlive-6 z roku 2001.*







## Rejstřík

- texlive2003, 7
- AucTeX, 9
- bakalářská práce, 18
- Balík
  - graphics, 17
  - hyperref, 18
- BibTeX, 7, 13–14
- bm2font, 16–17
- citace, viz BibTeX
- CSIndex, viz MakeIndex
- CS TUG, 6
- čeština, viz Kódování
- diplomová práce, 18
- dvips, 10, 17
- Editory
  - TeX Shell, 8
  - emacs, 8–9
  - fte, 8
  - joe, 8
  - pico, 8
  - TeX Shell, 8
  - vim, 8
  - WinEdt, 8
- fonty, 18–19
- hypertext, 18
- Kódování, 9
  - ISO-8859-2, 10
  - Windows 1250, 10
  - změna, 10
- L<sup>A</sup>T<sub>E</sub>X
  - struktura dokumentu, 11
  - úvod, 5–6
- literatura, 20–21
- logo FI, 18
- MakeIndex, 7, 12–13
- METAFONT, 16
- METAPOST, 16
- moduly
  - cstools, 10
  - emacs, 9
  - grafika, 17
  - psutils, 19
  - texlive-6, 7
  - texlive2003, 6
  - tex, 6, 7
- obrázky, 15–17
  - .eps, 17
  - .jpg, 17
  - .pdf, 17
- .tif, 17
- bitmapy, viz bm2font
- packages, 11
- pdfTeX, 10, 17
- picture, 15
- plainTeX, 4, 5
- PostScript, 19
- Prohlížeče
  - Acrobat Reader, 10
  - GhostView, 10
  - Windvi, 10
  - xdvi, 10
- předložky, viz vlna
- rejstříky, viz MakeIndex
- TBN, 20
- TeX Live 2003, 7
- TeX Shell, viz Editory
- texmf.cnf, 18
- tisk, 19
- TST, 20
- vlna, 12
- Windows, 7
- xterm, 8, 9
- Zpravodaj, 21





## 1 Co je to $\TeX$ ?

$\TeX$  (čte se tech) je programový systém pro elektronickou sazbu, jehož autorem je prof. Donald E. Knuth ze Stanfordské univerzity. První verze  $\TeX$ u, která byla dána k dispozici veřejnosti, byla dokončena v roce 1982.

Práci v  $\TeX$ u bychom mohli přirovnat k programování v některém z programovacích jazyků. Nejprve vytvoříte zdrojový text (k tomuto účelu vám poslouží kterýkoliv ASCII editor), přeložíte jej  $\TeX$ em, ten vám případně vypíše chybové hlášky, a poté, co chyby opravíte, získáte přeložený dokument ve formátu DVI.  $\TeX$  jako takový obsahuje zhruba 300 základních značkovacích příkazů. Pomocí makrojazyka, který je součástí  $\TeX$ u, můžete vytvářet nové sázecí příkazy složitějšími konstrukcemi z příkazů základních. Díky tomu si můžete vytvořit prostředí tak, aby vyhovovalo co nejlépe vašim potřebám. Takto vznikla také spousta více či méně známých formátů  $\TeX$ u, jako je  $\LaTeX$ ,  $\AMS\TeX$ , ale i  $\plain\TeX$ .

K velkým přednostem  $\TeX$ u patří jeho ekonomická a strojová nenáročnost.  $\TeX$  jako takový byl prof. Knuthem dán veřejnosti zdarma. Kdokoliv jej tedy může volně využívat, případně jej různě vylepšovat nebo dokonce vytvářet s jeho využitím nové systémy, obecně je však již nesmí označovat  $\TeX$ . Existují však i implementace  $\TeX$ u, které zdarma nejsou.

Co se týče požadavků na hardware, je  $\TeX$  velmi nenáročný. Můžete jej používat i na zastaralých PC, u kterých již jiný současný software s velkou pravděpodobností vůbec nepoběží. Zjednodušeně řečeno, stejného výsledku dosáhnete jak na staříčkém počítači typu PC XT, tak na nejnovějších výkonných stanicích. Zde je také dobré uvést, že  $\TeX$  je implementován v mnoha vzájemně se dosti lišících systémech (UNIX, MS Windows, MacOS X, ...).

K dalším přednostem  $\TeX$ u patří kompatibilita jeho implementací. Ze stejného vstupu získáte vždy stejný výstup, nezávisle na použité verzi  $\TeX$ u.

### 1.1 Zdroje informací o $\TeX$ u

Začátečnickovi v  $\TeX$ u lze vřele doporučit např. publikaci  *$\LaTeX$  pro začátečníky* [1] nebo *Jemný úvod do  $\TeX$ u* [2]. Pro již zkušenější uživatele bude zajisté velmi užitečný manuál  *$\TeX$ book* [3]. Mnoho informací týkajících se  $\TeX$ u naleznete také v knize *Typografický systém  $\TeX$*  [4] a  *$\TeX$ book naruby* [5].

Zmíněné publikace si lze např. zapůjčit v knihovně FI nebo objednat prostřednictvím knihkupectví Mareček (viz <http://marecek.kup.to>). Mnoho užitečného naleznete taktéž na internetové adrese  $\mathcal{C}_S$ TUGu [6].

## 2 Co je to $\plain\TeX$ ?

$\plain\TeX$ , jak již bylo uvedeno v úvodním odstavci, je makrojazykem vytvořené prostředí neboli formát  $\TeX$ u. Tento formát naprogramoval sám autor  $\TeX$ u Donald Knuth.  $\plain\TeX$  je považován za základní formát, od kterého je většina dalších běžně užívaných formátů více či méně odvozena.





Pro představu, plainTeX rozšiřuje počet příkazů, které může uživatel použít, na devět set. Výhodou tohoto formátu je jeho stabilita (např. oproti L<sup>A</sup>T<sub>E</sub>Xu), a rychlost kompilace. Pro práci v tomto formátu se zpravidla rozhodují uživatelé tehdy, pokud nepotřebují psát složité dokumenty, nebo tehdy, pokud chtějí vytvářet svůj vlastní formát či prostředí a užítí jiného, vyššího formátu by bylo pro ně nevhodné (příliš neprůhledné). PlainTeX však vyžaduje schopnost programátorského myšlení, protože většinu maker si musí uživatel vytvořit a odladit sám, a to může být časově náročné.



### 3 Něco o L<sup>A</sup>T<sub>E</sub>Xu...

L<sup>A</sup>T<sub>E</sub>X je nadstavbou T<sub>E</sub>Xu vytvořenou panem Leslieem Lamportem. A k čemu a pro koho je L<sup>A</sup>T<sub>E</sub>X určen? Tento systém byl vytvořen zejména proto, aby zjednodušil sazbu dokumentů v T<sub>E</sub>Xu a zpřístupnil tak jinak poněkud složitý jazyk běžnému uživateli. L<sup>A</sup>T<sub>E</sub>X nabízí např. automatické číslování různých objektů, jako jsou kapitoly, odstavce, tabulky, obrázky apod. Dále umí automaticky sestavit obsah či snadno vytvářet rejstříky. Obsahuje příkazy, pomocí kterých je možné vytvářet jednoduché obrázky. Taktéž má již předvoleny velikosti nadpisů kapitol, odstavců apod. Standardní styly se však řídí americkými typografickými konvencemi, které se značně liší od těch evropských. Proto bylo vytvořeno i mnoho stylů, které respektují evropská typografická pravidla.

Je ovšem vhodné zdůraznit, že zdrojový soubor dokumentu vytvořený v L<sup>A</sup>T<sub>E</sub>Xu se liší od zdrojového souboru v T<sub>E</sub>Xu. Proto, pokud tento L<sup>A</sup>T<sub>E</sub>Xový soubor přeložíte T<sub>E</sub>Xem, dočkáte se z největší pravděpodobností pouze spousty chybových hlášek a přeloženého souboru, který nebude zdaleka odpovídat vašim představám. Jestliže se pokusíte naopak přeložit plainTeXový soubor programem (cs)latex, nepodaří se vám to vůbec.

Problém je v poněkud odlišné syntaxi L<sup>A</sup>T<sub>E</sub>Xu oproti základnímu T<sub>E</sub>Xu. Charakteristickým rysem vstupních textů ve formátu L<sup>A</sup>T<sub>E</sub>X jsou prostředí vymezená příkazy `\begin` a `\end`, které jste nuceni při tvorbě textu dodržovat. Tyto bloky mohou být do sebe vnořovány stejně jako např. u některých programovacích jazyků pascalovského typu. Pro názornost uvedeme příklad jednoho a téhož dokumentu vytvořeného nejprve v plainTeXu a poté v L<sup>A</sup>T<sub>E</sub>Xu. U obou z těchto vstupních textů vznikne po přeložení příslušnými programy (csp<sub>l</sub>ain a csl<sub>a</sub>tex) velmi podobně vysázený dokument, ale zdrojové soubory se evidentně liší.

#### Vstupní soubor v plainTeXu:

```
\input czech.sty           Zprávu, kterou právě čtete,  
\nopagenumbers            nečtěte, nic se v~ní  
                           nedočtete...  
  
\rightline{  
V~Brně dne \today}       \vskip 0.6cm S~pozdravem  
  
{\bf Vážený čtenáři,}    \rightline{\it Kdosi cosi}  
  
\vskip 0.9cm             \bye
```





### Vstupní soubor v $\LaTeX$ u:

```

\documentclass{article}          \vspace{0.9cm}
\usepackage{czech}              Zprávu, kterou právě čtete,
\pagestyle{empty}              nečtěte, nic se v`ní
\begin{document}               nedočtete\dots
                                \vspace{0.6cm}

\begin{flushright}
V`Brně dne \today              Š`pozdravem
\end{flushright}              \begin{flushright}
                                \textit{Kdosi cosi}
\textbf{Vážený čtenáři,}      \end{flushright}
                                \end{document}

```

$\LaTeX$  některé příkazy obsažené v základním formátu nezná a místo nich má předdefinována prostředí (v uvedeném příkladu je to např. místo příkazu `\rightline` prostředí `flushright`). Jiným příkladem nekompatibility  $\LaTeX$ u vzhledem k `plainTeX`u jsou příkazy, které mají v obou formátech stejný název, ale různou funkci (např. `\line`). Avšak  $\TeX$ ová primitiva lze samozřejmě v  $\LaTeX$ u použít.

Na tomto místě je také vhodné upozornit na skutečnost, že formát  $\LaTeX$  se stále vyvíjí. Původní verze  $\LaTeX$ u označovaná jako verze 2.09 již není ani autorem ani  $\LaTeX$ -tým nadále podporována a udržována. Na místo toho vznikl a rozšířil se  $\LaTeX$  2<sub>ε</sub> (pracuje se na verzi označované  $\LaTeX$  3).

K formátu  $\LaTeX$  existuje manuál L. Lamporta  *$\LaTeX$ —A Document Preparation System* [7]. Velmi užitečná je také série knih *The  $\LaTeX$  Companion* [8, 9, 10], [10]. Z česky psaných materiálů je velmi vhodnou příručkou pro začínající uživatele  $\LaTeX$ u výše uvedená kniha  *$\LaTeX$  pro začátečníky* [1]. Doporučujeme také navštívit webovské stránky  $\mathcal{C}\mathcal{S}\mathcal{T}\mathcal{U}\mathcal{G}\mathcal{U}$  [6].

## 4 $\TeX$ pod UNIXem

Uživatelům na FI jsou v současnosti k dispozici mimo jiné pracovní stanice s operačním systémem UNIX, konkrétně pak se systémem IRIX, Solaris a Linux.

První krok, který musíte udělat, pokud chcete začít  $\TeX$ ovat na těchto unixových strojích, je nahrát si modul `texlive2003` (příp. `tex`, `texlive-6`), a to příkazem `add gs acrobat texlive2003`. Základní informace o tomto modulu získáte pomocí příkazu `module help texlive2003`.

Po spuštění výše uvedeného příkazu se vám zpřístupní např. programy `tex`, `cslatex`, standardní fonty, některá makra a také další užitečné programy, jako např. `csindex`, `vlna` atd. Zjednodušeně řečeno, bude vám dostupný celý adresář `texlive2003`. Pro ty z vás, kteří ještě nevědí, jak pracovat se systémem modulů na FI, doporučuji použít příkaz `module help modules` či zabrousit na fakultní technické stránky <http://www.fi.muni.cz/tech> [11].





## 4.1 Co to je T<sub>E</sub>X Live 2003?

T<sub>E</sub>X Live 2003 je distribuce T<sub>E</sub>Xu pro UNIX a Windows32 systémy. Zahrnuje T<sub>E</sub>X, L<sup>A</sup>T<sub>E</sub>X 2<sub>ε</sub>, METAFONT, METAPOST, MakeIndex a BibT<sub>E</sub>X, velké množství maker, fontů a dokumentace. Instalační CD/DVD T<sub>E</sub>X Live 2003 je možné si zapůjčit v knihovně nebo zakoupit u ČS TUGu a nainstalovat si jej např. na svém domácím počítači. Dokonce je možné spouštět programy přímo z CD/DVD. Kompletní dokumentace je k dispozici v [12], domovská stránka projektu je <http://tug.org/texlive/>.

Pozn.: instalace T<sub>E</sub>X Live 2003 na fakultních strojích se průběžně doplňuje o další programy apod., a proto se může (obsahem adresáře `texmf-local`, viz níže) lišit od případné vaší instalace doma.

Soubory přidávané do instalace `texlive2003` na FI jsou důsledně instalovány do adresáře `/packages/share/texlive2003/texmf-local/`. Informace o těchto změnách jsou pak zaznamenány v souboru `/packages/share/modules-2.0/modulefiles/texlive2003`.



## 4.2 Modul `texlive2003` vs. moduly `texlive-6`, `tex`

Na unixových strojích je možné T<sub>E</sub>Xovat kromě `texlive2003` také v rámci modulu `texlive-6` nebo `tex`. Tyto moduly jsou předchůdcem instalace `texlive2003`. Modul `tex` však již není v současnosti nadále udržován a je dostupný pouze kvůli zpětné kompatibilitě, modul `texlive-6` bude fungovat ještě nějaký čas, ale nedoporučujeme jej využívat, protože bude v brzké době odstraněn. Případné problémy, které se vám vyskytnou při jejich používání, proto zkuste vyřešit v modulu `texlive2003`. Přepnutí z modulu `tex` do `texlive2003` docílíte příkazem `module switch tex texlive2003`.

Jedním z rozdílů mezi moduly `tex` a `texlive2003` je např. způsob práce s česky psanými dokumenty. Pokud chcete přeložit český dokument s českým dělením slov apod., v rámci modulu `tex` to provedete programem `tex` (příp. `latex`). V modulu `texlive2003` jsou pro tyto účely vyhrazeny dávky `csplain` a `cslatex`. Programy `tex` a `latex` jsou zde ryze anglické a české dokumenty vám nepřeloží.

## 5 T<sub>E</sub>X pod Windows

Uživatelé mohou v současné době pracovat v celouniverzitní počítačové studovně na strojích s operačním systémem Windows 2000, stejně tak jsou k dispozici stroje s tímto operačním systémem i uživatelům na FI.

První, co musíte udělat, pokud chcete T<sub>E</sub>Xovat ve Windows na FI, je spustit dávku `TeX.cmd` (Start → Programs → `TeX.cmd`). Tato dávka namapuje potřebné binární soubory a provede základní nastavení, aby mohly být použity. Pro usnadnění práce při příštím spuštění je výhodné vytvořit si odkaz na soubor `TeX.cmd` a umístit jej do složky Start → Programs → Startup a celý postup se provede automaticky po nastartování Windows (podobně lze přidat spuštění dalších programů).

Dávka dále odmapuje disky `o:`, `p:` a `q:` (tyto disky nesmí používat žádný proces, jinak se odmapování nezdaří). Namapuje se `aisa:/packages/share` na `p:`, `aisa:/`





packages/share/texlive2003 na o: a aisa:~ na q: . Proměnná HOME bude nastavena na home z aisy (q:). Z tohoto důvodu je nutné, abyste měli, i přestože chcete T<sub>E</sub>Xovat na FI pouze ve Windows, zřízen účet na unixových strojích. Ve studovně je T<sub>E</sub>X Live 2003 nainstalován na síťovém disku J: v adresáři \texlive2003\.

Jako editor pro přípravu zdrojových textů dokumentů je možné použít Emacs, který se spouští příkazem runemacs (příp. emacs -nw pro spuštění v cmd-okně).

Z dalších editorů je vám k dispozici např. editor T<sub>E</sub>X Shell (příkazem texshell).

V počítačových učebnách B311 a B117 na FI můžete také využít editor WinEdt (viz [www.winedt.com](http://www.winedt.com)). K diskusi o užití tohoto editoru slouží diskusní skupina [winedt+list-subscribe@wsg.net](mailto:winedt+list-subscribe@wsg.net). Pod Windows jsou vám k dispozici také programy Windvi, GhostView, Acrobat Reader a další.

Ke spuštění příkazů využívejte Příkazovou řádku (Start → Programs → Command Prompt, příp. Cmd.exe).

## 5.1 Unixově ve Windows...

Další možností, jak T<sub>E</sub>Xovat na windowských strojích, je přihlásit se po síti na libovolný málo vytížený unixový počítač a přeměrovat displej na váš windowsovský počítač. Pak můžete v pohodlí T<sub>E</sub>Xovat v prostředí unixu. A jak se to dělá?

Ve Windows spustíte program Astec-X (Start → Programs → Astec-x → ASTEC-X), což je X-Server pro Windows. Poté se programem Tera Term (Start → Programs → → TTSSH) přihlásíte na některý málo vytížený unixový počítač a příkazem `export DISPLAY=adresa_vaseho_pocitace:0` přeměrujete grafický výstup na váš monitor (adresa vašeho počítače je uvedena na štítku počítače, u kterého sedíte, např. `sirene02.fi.muni.cz`).

Ve studovně je možné využít systém Cygwin (nainstalován je na síťovém disku J:). Po jeho spuštění stačí napsat příkaz `ssh -X adresa_serveru -l login`, a poté spouštět T<sub>E</sub>X přímo na unixovém počítači (parametr `-X` automaticky přeměruje displej a nastaví xhost).

## 6 Editory

Pro vytváření T<sub>E</sub>Xovských dokumentů můžete využít řadu editorů. V podstatě jediným omezením, co se týče možnosti využít ten či onen editor pro psaní T<sub>E</sub>Xovských dokumentů, je to, aby tento editor byl schopen textový soubor ukládat v čisté formě, tedy bez svých řídicích znaků. Zejména na unixových strojích je to např. základní editor `vim` (příp. `vi`). Díky svému poněkud odlišnému ovládní oproti jiným zejména neunixovým editorům může `vim` dělat začátečníkovi při vytváření dokumentů určité problémy. Proto je nezbytné se seznámit alespoň se základními prvky ovládní tohoto editoru, a to např. přímo příkazem `:help` v editoru `vim`, na manuálové stránce (`man vi`) nebo v příručkách o Unixu. Mezi další editory, které jsou vám na unixových strojích k dispozici, patří `pico`, `joe` nebo `fte`.

Na windowsovských strojích je vám také k dispozici např. editor T<sub>E</sub>X Shell nebo WinEdt (viz předchozí kapitola).







## 6.1 Emacs / AucTeX

Asi nejlépe uzpůsobeným nástrojem pro tvorbu nejen TeXovských dokumentů, který můžete na fakultních strojích používat, je editor emacs.

Pokud jej chcete používat na unixových strojích, je potřeba si nejprve přidat příslušný modul, tedy `module add emacs`. (v současné době jsou instalované verze 20.5. a 21.3)

Emacs nabízí mnoho konfiguračních možností, naprogramovaných v jazyce Lisp, včetně podpory pro psaní textů v TeXu. Můžete z něj přímo volat TeX a různé obslužné programy, jednoduše pomocí konkrétních sekvencí vkládat do textu některé časté TeXovské konstrukce apod. Kromě toho můžete také využít balík lispovských marker pro Emacs – AucTeX, který nabízí integrované prostředí pro tvorbu textů zejména v L<sup>A</sup>TeXu.

Vzhledem k tomu, jaké možnosti Emacs skýtá, má také přiměřeně obsáhlou dokumentaci (zhruba 300 stran uživatelského manuálu + 700stránkový manuál pro psaní lispovských kódů). Je také třeba upozornit, že stejně jako u vim editoru, může být pro začátečníky poněkud problematické si zvyknout na ovládání a vůbec jiné prostředí Emacsu ve srovnání s DOSovskými editory. Kdo se však nenechá odradit těmito zdánlivými nevýhodami a rozhodne se Emacs používat, určitě nebude litovat :-).

Pozn.: Na fakultních technických stránkách [11] již bohužel *nej*sou vystaveny podrobnější informace jak o konkrétních zvláštnostech instalace Emacsu na FI, tak o Emacsu jako takovém.



## 7 Kódování, čeština a spol.

Velmi důležité pro tvorbu česky psaných TeXovských dokumentů je „nastavení češtiny“. Toto můžeme rozdělit na dvě oblasti.

Jednak je důležité, aby se vám korektně zobrazovaly české znaky při tvorbě dokumentu, tedy v terminálovém okně resp. v prostředí některého z editorů.

Pokud váš xterm píše místo českých znaků číslice, stiskněte klávesu Scroll Lock (příp. zaráz obě klávesy Shift) nebo klávesu Pause. Jestliže čeština stále nefunguje, můžete se ji pokusit nastavit příkazem

```
xrdb /usr/lib/X11/app-defaults/csxterm.qwerty resp. qwertz
```

Toto nastavení se projeví až v dalších xtermech (po následném spuštění nového terminálu: `xterm &`).

Jestliže se vám i nadále zobrazují místo českých znaků nějaké nesmyslné znaky, bude zřejmě problém v komunikaci s některým z fontserverů. Zkuste zadat příkaz

```
xset fp+ tcp/font:7100
```

Další důležitou věcí je, v jakém kódování máte napsaný váš vstupní TeXovský soubor. Zejména je to aktuální, jestliže si přinášíte již rozpracovaný dokument odjinud. Může se stát, že takový text bude v jiném kódování, než jaké je implicitně nastaveno na unixových strojích, tedy ISO-8859-2. Toto mívá někdy za následek špatně zobrazené české znaky v přeloženém výstupním souboru. Pozor! Např. pokud přeložíte donesený TeXt, který je kódovaný ve Windows 1250, přeložený dokument se vám bude možná zdát na první pohled v pořádku. Ona se totiž většina českých znaků zobrazuje stejně





jak ve Windows 1250, tak v ISO-8859-2. Neplatí to ale např. pro „ž“, a tak bude váš přeložený dokument prošpikován znaky „I“...

Abyste se vyhnuli těmto problémům, měli byste vstupní T<sub>E</sub>Xty, které jsou uloženy v jiném kódování, převést do ISO-8859-2, což je implicitní vstupní kódování T<sub>E</sub>Xu na fakultních unixových strojích (ve Windows je to pak kódování Windows 1250). Jinou možností je překódovávat „on the fly“, a to pomocí přepínače `-translate-file=cp1250t1.tcx` při spouštění překladu.

## 7.1 Jak převádět texty mezi různými kódováními

Na fakultních strojích existuje v modulu `cstools` program `cstocs`, který umí převádět mezi jednotlivými kódováními češtiny.

Např. pokud budete chtít převést dokument `priklad.tex` v kódování Windows 1250 do kýženého kódování ISO-8859-2, budete postupovat takto:

Přidáte modul `cstools`, tedy `module add cstools`. A nyní už jen stačí spustit `cstocs 1250 il2 <priklad.tex >priklad.il2.tex` a v souboru `priklad.il2.tex` máte váš soubor v kódování ISO-8859-2. Pro úplnost zde uvedeme, s kterými kódováními program `cstocs` umí pracovat: 1250 1252 ascii cork ill il2 kam koi8 mac macce pc2 tex vga.

Podrobnosti naleznete na manuálové stránce (viz příkaz `man cstocs`).

Další možností, jak převést text z jednoho kódování do druhého, je využít skript `cnv` Libora Škarvady uložený v adresáři `/packages/share/CHARSETS/`.

Použít jej lze např. pro převod „Kamenický do ISO 8859-2“ následovně:

```
/packages/share/CHARSETS/cnv kam.isolat2 soubor1>soubor2.
```

Pozn.: mezi jakými kódováními je možné tímto skriptem převádět, viz příkaz

```
ls /packages/share/CHARSETS/.
```

## 8 Prohlížeče a konverze mezi formáty

Při práci s T<sub>E</sub>Xem se většinou setkáte s několika různými formáty, ve kterých bude váš dokument přeložen. Jakmile přeložíte dokument programem `(l)atex`, získáte `.dvi` soubor. Na fakultních strojích je vám k dispozici pro unixové platformy program `xdvi`, který vám umožní tento soubor si prohlédnout. Obdobou tohoto prohlížeče pro windowsovské stroje je program `windvi`.

Jestliže chcete získat dokument ve formátu PostScript, můžete použít pro převod z `dvi` formátu program `dvips`. Získáte tak `.ps` soubor. Zobrazit si jej můžete v unixu pomocí programu `Ghostview` (příkazem `gv`, pod Windows pak příkazem `gsviiew32`).

Do PDF formátu je možno převést dokument z PostScriptu např. pomocí programu `epstopdf`. Další možnost, jak `.pdf` soubor získat, je využít program `pdftex` (`pdflatex`, `pdfcslatex`,...), který vám jej vytvoří přímo ze vstupního souboru (L)T<sub>E</sub>Xu [13]. Pro prohlédnutí získaného `.pdf` souboru lze využít program `Acrobat Reader` (na unixových strojích nutno přidat příslušný modul: `module add acrobat`), `Ghostview` nebo na některých unixových platformách program `xpdf`.





## 9 ... a jdeme (L<sup>A</sup>)T<sub>E</sub>Xovat...

Pro přehlednost nejprve uvedeme, jakou má dokument určený ke zpracování systémem L<sup>A</sup>T<sub>E</sub>X strukturu.

```
\documentclass[volby]{třída}[datum vytvoření]
.
. preamble
.
\begin{document}
.
. textová část
.
\end{document}
```

Příkaz `\documentclass` spolu s parametrem *třída*, který určuje styl sazby dokumentu, je povinný. Ze standardních tříd máte na výběr `article` (pro článek), `book` (pro knihu), `letter` (pro dopis), `report` (pro zprávu) a `slides` (pro průsvitné fólie). Příslušné textové soubory s příponou `.cls`, v kterých jsou uloženy definice tříd, můžete samozřejmě libovolně upravovat, a vytvářet si tak své vlastní třídy uzpůsobené pro vaše vlastní použití.

Další, volitelný, parametr příkazu `\documentclass` vám umožňuje modifikovat činnost příkazů ve zvolené třídě. Uvést můžete i několik voleb najednou, musíte je však oddělit čárkami bez mezer, např.

```
\documentclass[10pt,a4paper]{article}
```

V preambuli dokumentu uvedené příkazy mají platnost pro celý text. V této části lze k dokumentu připojit další balíky příkazů (packages), a to pomocí příkazu

```
\usepackage[volby]{balík}[datum vytvoření].
```

Význam a struktura parametrů tohoto příkazu jsou prakticky stejné, jako u příkazu `\documentclass`.

Jakmile některý z balíčků takto připojíte do preambule dokumentu, můžete příkazy v balíku nadefinované používat v celém dokumentu.



## 10 Balík pro psaní českých dokumentů (czech.sty)

Pro tvorbu česky psaných dokumentů je určen balík `czech.sty`. Po jeho zavolání (např. v L<sup>A</sup>T<sub>E</sub>Xu přidáním `\usepackage{czech}` do preambule dokumentu) se změní činnost (L<sup>A</sup>)T<sub>E</sub>Xu v několika bodech:

- Zapne se české dělení slov. Znamená to, že bude (L<sup>A</sup>)T<sub>E</sub>X pracovat s českou tabulkou dělení slov a tabulkou výjimek, přičemž tabulku výjimek lze postupně rozšiřovat na libovolných místech dokumentu, a to pomocí příkazu `\hyphenation`.
- Mezery za tečkami a čárkami budou menší, odpovídající lépe normám české sazby.
- Je definován příkaz `\uv{...}` pro psaní českých uvozovek.





- Předefinuje se příkaz `\chapter` tak, že způsobí sazbu slova „Kapitola“ místo původně definovaného slova „Chapter“. Stejně tak se „počestí“ další automaticky generovaná slova, jako např. „Literatura“, „Obrázek“ apod.
- Příkaz `\today` vysází datum česky.

Dalším nástrojem pro psaní dokumentů v češtině nebo jiných jazycích je systém maker zvaný Babel. Jde o soubor konvencí a maker pro cizojazyčné styly.

## 11 Program vlna

Řádky v textu mohou být děleny buď „uvnitř“ slova (o tom jsme se zmínili výše), a nebo v místě mezer.  $\TeX$ u je možné zakázat, aby v určité mezeře řádek rozdělil. Proveďte se to znakem `~`. Např. je nesprávné z hlediska typografické normy, aby se na konci řádku vyskytovaly neslabičné předložky. V takových případech znak `~` lze účelně využít (Např.  $V^{\sim}\TeX\{u$  využíváme `vlnku...`).

Při psaní textu však není třeba, co se týče neslabičných předložek, na vkládání „vlnek“ stále pamatovat. Místo toho stačí kdykoliv provést jejich doplnění automaticky, a to programem `vlna`.

Bohužel však existuje mnoho dalších míst, kde podle pravidel sazby nelze řádky rozdělít a kde `vlna`, „vlnku“ nedoplní (např. za iniciálami:  $J. \sim K. \sim Ty1$  apod.). Na taková místa je potřeba dávat při psaní dokumentu pozor.



## 12 Rejstříky a citace

Pro snazší tvorbu rejstříků a seznamů citací jsou vám k dispozici např. speciální programy `MakeIndex` ( $\mathcal{C}\mathcal{S}$ Index) a `Bib $\TeX$` .

### 12.1 MakeIndex

Pokud chcete, aby váš dokument obsahoval také rejstřík, musíte v textu konkrétním způsobem označit ta hesla, která se v rejstříku mají objevit. Provádí se to pomocí příkazu `\index`, za jehož argument uvedete heslo spolu s příp. dalšími nastaveními (podhesla, namísto čísel stránek zobrazovat stránkový rozsah apod.). Tyto příkazy se vkládají přímo do míst v textu, na které má příslušné heslo rejstříku odkazovat.

Např.:

```
Těleso ponořené do kapaliny
\index{těleso!ponořené do kapaliny} je nadlehčováno silou,
která se rovná tíze \index{tíha} kapaliny tělesem vytla-
čené.
```

Tento příklad nám v konečné fázi vysází rejstřík takto:

```
těleso
  ponořené do kapaliny, 3
tíha, 5
```





Nejprve je třeba přidat do preamble dokumentu příkaz `\makeindex`. Ten způsobí, že se všechny příkazy `\index` uvedené v textu vypíší do souboru s příponou `.idx`. V daném souboru tak vznikne jakýsi polotovar budoucího rejstříku – nesetříděný, neupravený do tvaru seznamu. Z tohoto `.idx` souboru vytvoří plnohodnotný rejstřík program `MakeIndex` nebo jeho česká verze `CSIndex`. K jejich použití je potřeba připojit do preamble dokumentu balík `makeidx` příkazem `\usepackage{makeidx}`.

Nyní už jen vložíte na místo v textu, kde chcete, aby byl rejstřík vysázen, příkaz `\printindex`.

Vzniklý dokument přeložíte (např. `cslatex priklad`), čímž se vytvoří zmiňovaný soubor s příponou `.idx`. Poté spustíte program `makeindex` nebo `csindex` spolu se jménem tohoto souboru (`csindex priklad`). Tím se vytvoří již setříděný rejstřík, který se uloží do souboru s příponou `.ind`. Nakonec dokument ještě jednou přeložíte, přičemž při tomto druhém překladu již příkaz `\printindex` (viz výše) nalezne potřebný `.ind` soubor a načte jej. Tím vytvoří na požadovaném místě utříděný rejstřík. Způsobů, jak vytvořit rejstřík, je pochopitelně víc. Např. lze využít alternativního programu `xindy` (viz modul `xindy`).

## 12.2 BibTeX

Na tomto místě se krátce zmíníme také o možnostech automatického vytváření seznamů citací. K tomuto účelu, zejména pokud tvoříte ve formátu  $\text{\LaTeX}$ , je vám k dispozici program `BibTeX`. Nyní si stručně ukážeme jak jednoduše vytvořit seznam citací pomocí tohoto programu.

Nejprve si vytvoříte databázi publikací, a to v souboru s příponou `.bib` (např. `citace.bib`). Struktura tohoto souboru musí dodržovat přesně danou syntax. Pro představu uvedeme příklad, jak může taková databáze vypadat.

```
@ARTICLE{honza,  
author = "Novák, Jan",  
title = "Můj život",  
journal = "Životopisy",  
number = 4,  
year = 1997,  
pages = "14--16",  
month = "duben",  
}  
  
@BOOK{karel,  
author = "Nikdo, Karel",  
title = "Kniha o {A}rchimédovi",  
publisher = "Nakladatelství Ťuk",  
address = "Brno, Nějaká 15",  
note = "První vydání",  
year = "1992",  
}
```





Jde vlastně o soubor záznamů určitého typu (v tomto případě typu @ARTICLE a @BOOK), které obsahují jednotlivé údaje. Každý typ má svou vlastní skupinu povinných údajů.

Na příkladě vidíme, že každý záznam obsahuje mimo údajů také pracovní značku (zde je to honza resp. karel). Tu použijete v dokumentu spolu s příkazem `\cite` (např. `\cite{karel}`) tam, kde budete chtít na tu či onu publikaci (záznam) ze seznamu citací odkázat.

Zde je vhodné upozornit na skutečnost, že v textech názvů publikací (to, co je v položce `title=`) se všechna písmena kromě prvního vysázejí automaticky jako malá, bez ohledu na to, jak jsou napsána v souboru vaší databáze. Pokud chcete vysázet název s velkými písmeny, musíte je uzavřít do složených závorek (viz předchozí příklad).

Pro to, aby se v dokumentu seznam použitých citací vysázel, je dále potřeba do vstupního textu dokumentu mimo preambuli uvést příkaz `\bibliography` spolu se jménem souboru obsahujícího databázi publikací (např. `\bibliography{citace}`). Dále uvedeme příkaz `\bibliographystyle{jmeno_stylu}`, který specifikuje styl použitý programem BibTeX. Ze základních stylů jsou to např. `plain.bst`, `alpha.bst`, `abbrv.bst` a `unsrt.bst`. Jednotlivé styly se od sebe navzájem liší ve formátu výstupního souboru, ale většinou pouze nepatrně. Např. styl `unsrt.bst` způsobí, že výstup nebude uspořádán podle abecedy, zatímco styl `plain.bst` BibTeXu přikáže, aby výstup uspořádal.

Dokument i databázi máme nyní připravenou. A co dál?

- Dokument přeložíme L<sup>A</sup>T<sub>E</sub>Xem, čímž se vytvoří soubor s příponou `.aux`, který bude obsahovat instrukce (jméno souboru s databází, použitý styl apod.) pro program BibTeX.
- Spustíme program `bibtex` spolu se jménem dokumentu resp. souboru `.aux`, tedy např. `bibtex priklad`. BibTeX vytvoří automaticky podle instrukcí `.bbl` soubor se seznamem citací ve formátu, který je zpracovatelný L<sup>A</sup>T<sub>E</sub>Xem.
- Opět přeložíme náš dokument příkazem `cselatex`. Tím se načte do dokumentu již vytvořený `.bbl` soubor.
- Ještě jednou dokument přeložíme, čímž se zbavíme varovných hlášek o nepřirazených značkách, která se nám u předchozích zpracování L<sup>A</sup>T<sub>E</sub>Xem objevovala (kvůli souboru `.bbl`). A jsme hotovi.

Rejstřík, jakož i seznam literatury, se standardně sází na zvláštní stránku. To ovšem nemusí být vždy žádoucí. Někdy je potřeba např. z důvodu úspory místa vysázet rejstřík tak, aby navazoval na konec textu a sázel se do dvou sloupců. K tomu je třeba modifikovat okolí `theindex` ze stylu dokumentu, který používáte. Soubor s definicemi stylu naleznete v adresáři instalace `texlive2003`. K vyhledání můžete využít program `kpsewhich` (např. spustíte `kpsewhich book.cls`). Zkopírujete si z něj definici okolí `theindex` do preambule vašeho dokumentu mezi závorky `\makeatletter` a `\makeatother`, a poté nadefinujete novou verzi okolí příkazem `\renewenvironment`. Zejména se zaměřte na příkaz `\twocolumn`, nebo použijte (doporučeno) balík `multicol`.





V dokumentu využívajícím třídu `article` by tato úprava mohla vypadat například takto:

```
\usepackage{multicol}
\makeatletter           % Umožňuje použít @ v názvech maker
\renewenvironment{theindex}
{\section*{\indexname}%
 \begin{multicols}{2}
   \@mkboth{\MakeUppercase\indexname}%
   {\MakeUppercase\indexname}%
 \thispagestyle{plain}\parindent\z@
 \parskip\z@ \@plus .3\p@\relax
 \let\item\@idxitem}
{\end{multicols}}
\makeatother           % Zakazuje použít @ v názvech maker
```

Řešení pro bibliografii (okolí `thebibliography`) je analogické.

## 13 Práce s obrázky

Obrázky můžete do dokumentu vkládat různými způsoby. Zejména záleží na tom, zda jde o obrázky rastrové nebo vektorové.

$\TeX$  jako takový nebyl navržen pro tvorbu nebo vkládání obrázků. Přesto existují makra, která vám umožní jednoduché obrázky sestavovat. Vektorové obrázky jste tak schopni vytvářet v omezené míře např. přímo v  $\LaTeX$ u v prostředí `picture`. Druhou možností je využít systému `METAFONT` nebo `METAPOST`. Rastrové, ale i vektorové obrázky, které byly vytvořeny některým jiným specializovaným programem, se do textu vloží buď převedením na písma znakové sady, nebo využitím vlastností ovladače pro zobrazení souboru `.dvi`.



### 13.1 Prostředí `picture`

$\LaTeX$ ové prostředí `picture` je uzpůsobeno pro tvorbu spíše jednoduchých schémat a obrázků. Podrobněji se s tímto prostředím můžete obeznámit např. v knize  *$\LaTeX$  pro začátečníky* [1]. Pro představu uvedeme příklad obrázku:

```
\begin{picture}(35,55) \end{picture}
\linethickness{1pt}
\put(5,15){\framebox(40,30){\TeX}}
\thinlines
\put(10,20){\framebox(30,20){}}
\put(15,15){\line(0,-1){5}}
\put(35,15){\line(0,-1){5}}
\put(0,0){\framebox(50,10){}}
\put(30,5){\framebox(10,1){}}
```





Jak je vidět, vytváření obrázků touto cestou není příliš pohodlné, zejména, pokud tvoříme obrázky složené z více objektů. Jen pro zajímavost existuje program `TEXcad`, ve kterém můžete obrázky vytvářet v grafickém prostředí pomocí myši a po uložení získáte textový soubor obsahující `\begin{picture} ... \end{picture}` vašeho obrázku. Tento program sice není na FI dostupný, ale najdete ho v `emTEXu`, což je distribuce pro MS DOS a OS/2. Na podobném principu funguje také program `xfig`, který můžete na fakultě na některých unixových platformách využít (nutno přidat příslušný modul příkazem `module add xfig`).

### 13.2 Krátce o METAFONTu a METAPOSTu

Pokud však chcete vytvářet složitější vektorové obrázky, které se nebudou skládat pouze ze základních grafických elementů, jako je tomu u `LATEX`ového prostředí `picture`, můžete využít schopností programu METAFONT. Tento program určený k návrhu a realizaci písem je možné logicky využít i pro vytváření jiných obrázků než je návrh písma. Zmíňme se nyní také krátce o METAPOSTu. Je to programovací jazyk, vzniklý z jazyka METAFONT, určený pro popis obrázků. Hlavním rozdílem mezi METAFONTEM a METAPOSTem je jejich rozdílný výstup. Zatímco u METAFONTu je výstupem bitová mapa a metrika, u METAPOSTu je to program v jazyce PostScript.

Vzhledem k tomu, že problematika tvorby obrázků v METAFONTu je značně obsáhlá a rozebírat ji zde by zabralo příliš mnoho místa, nezbývá než odkázat na příslušnou literaturu.

#### Zdroje informací o METAFONTu

Jako úvod do METAFONTu může posloužit *Zpravodaj ČS<sub>T</sub>UG* [14] 1/1998, který můžete buď zakoupit v knihkupectví Mareček, zapůjčit v knihovně FI, příp. stáhnout v elektronické podobě z adresy <http://bulletin.cstug.cz>.

Cenné informace můžete získat také z dalších čísel *Zpravodaje* (např. 3/1991, 2/1992, 4/1994, 3-4/1998 aj.) nebo z webovských stránek ČS<sub>T</sub>UGu [6].

Stejně jako *The T<sub>E</sub>Xbook* [3] k T<sub>E</sub>Xu, sestavil D. Knuth také manuál k METAFONTu – *The METAFONTbook* [15] (možno také zapůjčit v knihovně FI).

Vcelku podrobně se METAFONTEM zabývá taktéž již výše zmiňovaná kniha *Typografický systém T<sub>E</sub>X* [4].

### 13.3 Program `bm2font`

Pokud chcete vkládat do vašeho T<sub>E</sub>Xového dokumentu rastrové obrázky (bitmapy), poslouží vám k tomu na unixových strojích např. program `bm2font`. Lze ho použít pro vstupní soubory formátu PCX, GIF, BMP, IFF/LBM, TIFF, IMG a CUT.

Program `bm2font` pracuje v principu asi takto:

- Rozdělí obraz na pravoúhlé oblasti a vygeneruje z nich „písmena“ jednoho nebo více fontů (vytvoří se tak `.pk` soubory a k nim příslušné `.tfm`).
- Vygeneruje další `.tex` soubor s definicemi příkazů, jimiž bude možno písmena nově vzniklých fontů opět spojit dohromady a vytvořit tak původní obrázek.







Aby se do textu obrázků vložil, je potřeba doplnit ve zdrojovém  $\text{T}_\text{E}\text{X}$ ovém souboru vašeho dokumentu příkaz `\input{jmeno_souboru}` pro připojení výše vygenerovaného definičního souboru. Pro samotné vykreslení obrázku se pak použije příkaz `\set...` Jeho definici, a tedy i jeho konkrétní tvar lze nalézt v právě připojeném definičním souboru. Zpravidla bývá příkaz tvaru `\set{jmeno_obrazku}` (např. pokud se soubor s obrázkem jmenuje `mapa.bmp`, příslušný příkaz bude `\setmapa`). Přesný tvar tohoto příkazu také vypíše na obrazovku `bm2font` během své činnosti (viz výše). Nevýhodou tohoto způsobu práce je nutnost vygenerování fontů v takovém rozlišení, jaké používá prohlížeč.

### 13.4 Balík `graphics`

Další možností, jak pracovat s obrázky v  $\text{L}_\text{A}\text{T}_\text{E}\text{X}$ u, je využít možností standardního balíku `graphics` příp. balíku `graphicx`. U těchto balíků se předpokládá existence ovladače, který je schopen pracovat s obrázky (standardně se jedná o zpracování grafiky zapsané v jazyce PostScript).

Pro vložení PostScriptového obrázku (zpravidla má příponu `.eps`) je definován příkaz `\includegraphics{jmeno_souboru}`.

Balík `graphics` vyžaduje, aby byl výstup v jazyce PostScript. Proto musí být obvyklým překladem získaný výstupní `.dvi` soubor převeden do PostScriptového souboru (např. pomocí programu `dvips`).

Pokud byste však chtěli vkládat PostScriptové obrázky (`.eps`) pomocí programu `pdflatex`, nespějete. Ten totiž v sobě neobsahuje interpret jazyka PostScript. Musíte tedy daný obrázek nejprve zkonvertovat do `.pdf` (příp. `.jpg`, `.tif` nebo `.png`). To lze provést např. programem `epstopdf` nebo pomocí programu `convert`, který je obsažen v modulu `grafika`. Takto zkonvertovaný obrázek již můžete vložit do dokumentu pomocí příkazu `\includegraphics`. Je však potřeba v preambuli dokumentu, kde voláte příslušný balík `graphics`, uvést navíc parametr `pdftex`, tedy `\usepackage[pdftex]{graphics}`.

Více podrobností nejen o vkládání obrázků pomocí programu `pdflatex` naleznete např. na stránkách *PdftEX Support* [13]. Podrobně se problematice vkládání externí grafiky věnuje také např. příručka *Using Imported Graphics in L<sup>A</sup>T<sub>E</sub>X 2<sub>ε</sub>* [16].



## 14 Jak vložit do dokumentu logo...?

Při psaní dokumentu bývá občas potřeba vložit do textu nějaké logo. Zde si uvedeme příklady některých z nich:

**$\text{T}_\text{E}\text{X}$  resp.  $\text{L}_\text{A}\text{T}_\text{E}\text{X}$**  — příkazem `\TeX` resp. `\LaTeX`

**METAFont resp. METAPOST** — po přidání balíku `mfllogo` příkazem `\MF` resp. `\MP`





### CSIndex, CS TUG apod.

```
\newcommand{\CS}{%
  \mbox{$\cal C\kern-.1667em\lower.5ex\hbox{%
    $\cal S$}\kern-.075em $}}
\newcommand{\cstug}{\CS\thinspace TUG}
\newcommand{\csindex}{\CS\thinspace index}
```

### Logo Fakulty Informatiky

např. `\font\fimu=fi-logo600 at 4cm`  
`{\fimu SL}`

## 15 Tvorba diplomových a bakalářských prací

Jako styl pro sazbu diplomových a bakalářských prací v systému L<sup>A</sup>T<sub>E</sub>X je vám k dispozici například třída `fithesis` (načte se příkazem `\documentclass{fithesis}` v preambuli dokumentu). Standardně je tato třída určena pro vytváření DP a BP na FI (řídí se pravidly pro psaní DP (BP) v rámci FI, vysází logo FI apod.), ale lze ji využít i pro sazbu prací studentů jiných fakult, jak se lze dočíst v aktuální dokumentaci balíku, kterou najdete příkazem `kpsewhich fithesis.dvi` a prohlédnete jakýmkoli dvi ovladačem. Třídu `fithesis` a modul `texlive2003` využívá pro sazbu i module `xslt2` pana Pavloviče pro přípravu DP/BP v XML Docbook.

Pokud máte zájem vytvářet také elektronickou podobu vaší práce obohacenou o hypertextové odkazy, můžete k tomuto účelu využít např. balík `hyperref`. Podrobné informace o tom, jak s tímto balíkem pracovat, lze nalézt např. na adrese <http://www.tug.org/applications/hyperref/> [17].

Další informace vztahující se k tvorbě hypertextových dokumentů je možné mimo jiné nalézt také v CSFAQ na stránkách CS TUGu [6].

## 16 Jak používat jiné fonty?

Fakulta zakoupila některé sady komerčních písem, u nichž však v současné době chybí podpora pro T<sub>E</sub>X. Na té se v současné době na půdě fakulty intenzivně pracuje a během několika následujících měsíců by měla být dokončena a dána k dispozici všem studentům. Bližší informace o postupu prací na zavádění nových písem do modulu `texlive2003` se dozvíte na [18].

Je pochopitelné, že na všech fakultních strojích nemohou být standardně k dispozici některé komerčně šířené fonty. Důvod je prozaický: fakulta nevlastní na takové fonty licenci. Jestliže vlastníte některé fonty, ať už jsou to komerční fonty, nebo vámi vytvořené, které nejsou ve fakultním `texlive2003` obsaženy, a chcete je při práci v `texlive2003` na fakultních strojích využívat, musíte provést některé změny nastavení.

Aby T<sub>E</sub>X „znal“ vaše fonty resp. ty, které `texlive2003` automaticky neobsahuje, musíte mu říct, kde je má hledat. Cesty, které T<sub>E</sub>X prochází při hledání fontů, jsou nastaveny v konfiguračním souboru `texmf.cnf`. Cesta k tomuto souboru je (po





přidání modulu `texlive2003`) nastavena v proměnné `TEXMFCONF` (viz příkaz `man set`).

Pokud tedy chcete, aby  $\TeX$  našel i vaše fonty (např. uložené ve vašem domovském adresáři), můžete to udělat takto:

- Nejprve zkopírujte konfigurační soubor `texmf.cnf` (adresu jeho umístění naleznete v proměnné `TEXMFCONF` (viz příkaz `echo $TEXMFCONF`) nebo příkazem `kpsewhich texmf.cnf` do svého adresáře. Zde pak můžete tento soubor podle potřeby pozměňovat. Zaměřte se zejména na nastavení cest v částech

```
PKFONTS=...
PostScript Type 1 outline fonts
TrueType outline fonts
Type 42 outline fonts
```

apod., podle toho, jaké druhy písem budete chtít z vašeho adresáře využívat. Výhodou je, že si můžete vytvořit ve svém adresáři takovou strukturu fontů, jaká vám vyhovuje.

(Pozn.: Charakter a rozsáhlost změn závisí na tom, jakou adresářovou strukturu si pro fonty navrhnete.)

- Dále je potřeba se zaměřit na soubor `texfonts.map` a s ním související soubory. V tomto souboru jsou definovány aliasy pro jména fontů v  $\TeX$ u.
- Změňte hodnotu proměnné `TEXMFCONF` tak, aby v ní byla nastavena cesta k vašemu upravenému `texmf.cnf`.

Více informací o problematice fontů v  $\TeX$ u naleznete např. v *The  $\LaTeX$  Companion* [8] nebo v knize *Typografický systém  $\TeX$*  [4]. Tomuto tématu je věnována pozornost také ve *Zpravodaji* [14] (konkrétně např. čísla 3–4/1998, 4/1999, 1–3/2000) a v diskusních fórech (viz stránky *ČS TUGu* [6]).

## 17 Jak vytisknout v PostScriptu více stránek na jednu A4, tisk brožury apod.

K tomuto účelu lze použít možnosti modulu `psutils`. Informace o tomto modulu včetně odpovědí na otázky podobného typu naleznete v nápovědě k příslušnému modulu (viz příkaz `module help psutils`) nebo na příslušných manuálových stránkách.

### Zpětná reakce

Pokud máte jakékoli připomínky či dotazy k obsahu této brožury nebo k instalaci `texlive2003`, zašlete nám je na emailovou adresu `texlive@fi.muni.cz`.





## Literatura

- [1] J. Rybička. *LT<sub>E</sub>X pro začátečníky – 2. vydání*. Brno: Konvoj, 1999.  
Příručka vhodná pro úplné začátečníky v L<sup>A</sup>T<sub>E</sub>Xu.
- [2] M. Doob. *Jemný úvod do T<sub>E</sub>Xu*. Praha: C<sub>S</sub>TUG, 1992.  
Manuál pro samostatné studium.
- [3] D. E. Knuth. *The T<sub>E</sub>Xbook*. Massachusetts: Addison-Wesley, 1994.  
Původní příručka k T<sub>E</sub>Xu přímo od D. Knutha.
- [4] P. Olšák. *Typografický systém T<sub>E</sub>X*. Brno: Konvoj, 1997.  
Vhodný průvodce jak pro pokročilé uživatele T<sub>E</sub>Xu, tak i částečně pro začátečníky. Tato kniha se zabývá zejména technickými souvislostmi T<sub>E</sub>Xu a doprovodného softwaru. Často se pro tuto publikaci používá zkratky TST.
- [5] P. Olšák. *T<sub>E</sub>Xbook naruby*. Brno: Konvoj, 1997.  
Tato publikace mimo jiné podrobně rozebírá všechny algoritmy T<sub>E</sub>Xu a většinu maker plainu a csplainu. Je vhodná zejména pro pokročilejší uživatele, kteří chtějí proniknout hlouběji do mechanismů fungování T<sub>E</sub>Xu. Tato publikace bývá často označována zkratkou TBN.
- [6] Internetové stránky Československého sdružení uživatelů T<sub>E</sub>Xu (C<sub>S</sub>TUG). <http://www.cstug.cz>.  
Na těchto stránkách je možno nalézt jednak odkazy související s T<sub>E</sub>Xem (např. diskusní fóra, FAQ, elektronické publikace a archívy softwaru), dále pak podrobné informace o náplni C<sub>S</sub>TUGu, stanovy, výhody členství v tomto sdružení apod.
- [7] L. Lamport. *LT<sub>E</sub>X—A Document Preparation System—Users's Guide and Reference Manual*. Massachusetts: Addison-Wesley, 1994.  
Původní manuál k systému L<sup>A</sup>T<sub>E</sub>X.
- [8] F. Goosens, M. Mittelbach. *The LT<sub>E</sub>X Companion*. Massachusetts: Addison-Wesley, 1994.
- [9] F. Goosens, M. Mittelbach. *The LT<sub>E</sub>X Graphics Companion*. Massachusetts: Addison-Wesley, 1997.
- [10] F. Goosens, M. Mittelbach. *The LT<sub>E</sub>X Web Companion*. Massachusetts: Addison-Wesley, 1999.
- [11] Webové technické stránky FI. <http://www.fi.muni.cz/tech>.  
Stránky obsahující informace, návody a pokyny týkající se počítačů na FI (rozděleno na oblasti: UNIX, Novell, Win NT, Obecné. . .).
- [12] Příručka T<sub>E</sub>Xlive. [/packages/share/texlive2003/texmf/doc/tldoc/](http://packages/share/texlive2003/texmf/doc/tldoc/), September 2003.
- [13] PdfT<sub>E</sub>X Support. <http://www.tug.org/applications/pdftex>.  
Stránky o programu PdfT<sub>E</sub>X a vše co s ním souvisí.
- [14] *Zpravodaj C<sub>S</sub>TUGu*.  
Tištěný čtvrtletník distribuovaný zdarma členům sdružení. Po uplynutí 12 měsíců





od tištěného vydání je poskytován v elektronické podobě (PDF) na <http://bulletin.cstug.cz>.

- [15] D. E. Knuth. *The METAFONTbook*. Massachusetts: Addison-Wesley, 1986.  
Původní příručka k METAFONTu přímo od jeho autora D. Knutha.
- [16] *Using Imported Graphics in L<sup>A</sup>T<sub>E</sub>X 2<sub>ε</sub>*.  
<ftp://ftp.cstug.cz/pub/tex/CTAN/info/epslatex.pdf>.  
Příručka, která se podrobně věnuje problematice vkládání externí grafiky v L<sup>A</sup>T<sub>E</sub>Xu.
- [17] *Hypertext marks in L<sup>A</sup>T<sub>E</sub>X*. <http://www.tug.org/applications/hyperref/manual.html>.  
Manuál k vytváření klikacích (hypertextových) dokumentů.
- [18] Dokumentace T<sub>E</sub>Xové podpory pro komerční fonty - pdf.  
</packages/share/texlive2003/texmf-local/doc/fifonts/>.

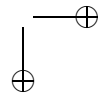
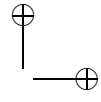




## Obsah

<b>1 Co je to T<sub>E</sub>X?</b>	<b>4</b>
1.1 Zdroje informací o T <sub>E</sub> Xu . . . . .	4
<b>2 Co je to plainT<sub>E</sub>X?</b>	<b>4</b>
<b>3 Něco o L<sup>A</sup>T<sub>E</sub>Xu...</b>	<b>5</b>
<b>4 T<sub>E</sub>X pod UNIXem</b>	<b>6</b>
4.1 Co to je T <sub>E</sub> X Live 2003? . . . . .	7
4.2 Modul texlive2003 vs. moduly texlive-6, tex . . . . .	7
<b>5 T<sub>E</sub>X pod Windows</b>	<b>7</b>
5.1 Unixově ve Windows... . . . .	8
<b>6 Editory</b>	<b>8</b>
6.1 Emacs / AucT <sub>E</sub> X . . . . .	9
<b>7 Kódování, čeština a spol.</b>	<b>9</b>
7.1 Jak převádět texty mezi různými kódováními . . . . .	10
<b>8 Prohlížeče a konverze mezi formáty</b>	<b>10</b>
<b>9 ... a jdeme (L<sup>A</sup>)T<sub>E</sub>Xovat...</b>	<b>11</b>
<b>10 Balík pro psaní českých dokumentů (czech.sty)</b>	<b>11</b>
<b>11 Program vlna</b>	<b>12</b>
<b>12 Rejstříky a citace</b>	<b>12</b>
12.1 MakeIndex . . . . .	12
12.2 BibT <sub>E</sub> X . . . . .	13
<b>13 Práce s obrázky</b>	<b>15</b>
13.1 Prostředí picture . . . . .	15
13.2 Krátce o METAFONTu a METAPOSTu . . . . .	16
13.3 Program bm2font . . . . .	16
13.4 Balík graphics . . . . .	17
<b>14 Jak vložit do dokumentu logo...?</b>	<b>17</b>
<b>15 Tvorba diplomových a bakalářských prací</b>	<b>18</b>
<b>16 Jak používat jiné fonty?</b>	<b>18</b>
<b>17 Jak vytisknout v PostScriptu více stránek na jednu A4, tisk brožury apod.</b>	<b>19</b>







1	Co je to T <sub>E</sub> X?
2	Co je to plainT <sub>E</sub> X?
3	Něco o L <sup>A</sup> T <sub>E</sub> Xu. . .
4	T <sub>E</sub> X pod UNIXem
5	T <sub>E</sub> X pod Windows
6	Editory
7	Kódování, čeština a spol.
8	Prohlížeče a konverze mezi formáty
9	. . . a jdeme (L <sup>A</sup> )T <sub>E</sub> Xovat. . .
10	Balík pro psaní českých dokumentů (czech.sty)
11	Program vlna
12	Rejstříky a citace
13	Práce s obrázky
14	Jak vložit do dokumentu logo. . . ?
15	Tvorba diplomových a bakalářských prací
16	Jak používat jiné fonty?
17	Jak tisknout v PostScriptu. . .
18	Literatura
19	Obsah

