Často kladené otázky o TEXu a odpovědi na ně (CSTUG FAQ)

28. dubna 1999

Poznámka

Části A–R tohoto dokumentu jsou překladem originálu, který napsal Bobby Bodenheimer. Poprvé byl publikován v prosinci 1994 v časopise Baskerville *vydávaném britským UK TUGem, a od té doby je UK TUGem udržován. Část S byla doplněna a obsahuje informace specifické pro CSTUG.*

Obsah

A. Úvod

První verze tohoto dokumentu vznikla překladem z článku vytvořeného komisí UK TEX Users Group (UKTUG)[1]. Ten vznikl z často kladených otázek (FAQ) na *Usenet*ové skupině `comp.text.tex`, kterýžto dokument udržoval Bobby Bodenheimer (`bobby@hot.caltech.edu`).

Usenet je mechanismus pro výměnu informací mezi lidmi, které sdružují společné zájmy a potřeby, na celosvětové síti Internet. Protože valná většina otázek (a odpovědí) se často pravidelně opakuje a zatěžuje jak linky, tak dlouhodobé čtenáře dané skupiny, někteří lidé se dali do sepisování často kladených otázek a odpovědí na ně.

Protože někteří členové CSTUGu nemají přístup k Internetu, byly Často kladené otázky a odpovědi na ně zařazeny do zvláštního čísla 1996/3 Zpravodaje sdružení (viz Otázku **??**).

Často kladené otázky jsou rovněž na CD distribuci TEX Live 3.

Jako východisko byla vzata anglická verze udržovaná UK TUGem a jsme vděčni za svolení k jeho překladu.[2] Překlad první československé verze provedli Leo Hadacz, Michal Kočer, Karol Nemoga, Petr Sojka a Jiří Zelinka. Vše dal dohromady, ujednotil, vysázel a udržuje Libor Škarvada (`cstug-faq@cstug.cz`), jemuž posílejte případné náměty na vylepšení či návrhy otázek a odpovědí k vložení do příští verze. Tomáš Hudec a Aleš Vítek vytvořili skripty pro generování WWW-verze dokumentu na `http://www.cstug.cz/csfaq/` a rozhraní pro jeho prohlížení a vyhledávání v něm.

Kde najít soubory zmíněné v tomto článku

Pokud není řečeno jinak, všechny soubory zmíněné v tomto dokumentu jsou dostupné z archívu CTAN a z jeho zrcadel. Blíže o archívech CTAN viz Otázku **??**, kde se dozvíte jak získat soubory. Pokud nemáte přístup k Internetu, v Otázce **??** se dozvíte o nabídkách a dostupnosti CD-ROM, které obsahují kopii archívu.

---

[1] Pro rok 1995–96: Peter Abbott, Kaveh Bazargan, David Carlisle, Malcolm Clark, Robin Fairbairns, Carol Hewlett, Alan Jeffrey and Sebastian Rahtz

[2] Na tvorbě dokumentu UK TUGu, ze kterého jsme vycházeli, se dále podíleli: Rosemary Bailey, Barbara Beeton, Karl Berry, Damian Cugley, Michael Downes, Jonathan Fine, John Hobby, Berthold Horn, Werner Icking, Ted Nieland, Pat Rau, Chris Rowley, Piet van Oostrum, Joachim Schrod, Philip Taylor, Ulrik Vieth, Rick Zaccone a Reinhard Zierke.

Čtenář by též neměl přehlédnout, že prefix specifikovaných cest se liší pro různá zrcadla CTAN. Pro `ftp://ftp.cstug.cz/` je to `/pub/tex/CTAN` Abychom předešli nedorozuměním, též vynecháváme tečku za větou, která končí URL.

B. Původ

1. Co je to TEX

TEX je sázecí systém napsaný Donaldem E. Knuthem, který v úvodu ke své knize o TEXu (viz Otázku **??**) řekl, že TEX je *„určený pro tvorbu hezkých knih — a obzvlášť knih obsahujících spoustu matematiky“*.

Knuth vytvořil k napsání TEXu systém „dokumentovaného programování“[3] a dokumentovaný zdrojový kód TEXu poskytl k volnému užívání spolu s programy pro zpracování zdrojového textu (`web`) do něčeho, co lze zpracovat překladačem, a do něčeho, co se dá vytisknout. Činnost TEXu není nikdy zahalena tajemstvím. Mimoto systém WEB poskytuje mechanismus, jak implementovat TEX na nových operačních systémech a počítačích. V zájmu zachování důvěry k různým implementacím dodal Knuth testovací soubor, podle kterého lze posuzovat věrnost dané implementace systému TEX (tzv. trip-trap test). TEX a jeho dokumenty jsou tak velmi dobře přenositelné.

TEX je makroprocesor, který nabízí svým uživatelům mocné programovací schopnosti. Z tohoto důvodu je TEX sám o sobě pěkně složitá potvora, se kterou bychom se museli potýkat, a tak Knuth napsal balík maker pro použití s TEXem nazývaný `plain` TEX. `plain` TEX je minimální množina maker, která lze efektivně použít s TEXem, společně s několika demonstrativními verzemi příkazů na vyšší úrovni (které je lepší považovat za vzory, než je používat tak jak jsou). Když člověk řekne, že „programuje v TEXu“, většinou tím myslí, že programuje v `plain` TEXu.

2. Jak mám vyslovovat „TEX“

Písmeno „X“ znamená řecké písmenko chí (χ), a anglickými mluvčími je vyslovováno buď trochu jako „ch“ ve slově „loch“ ([x] v IPA[4]) nebo jako „k“. Rozhodně není vyslovováno jako „ks“. V českém a slovenském jazyce se většinou používá výslovnost [tech].

3. Co to je METAFONT

METAFONT byl napsán Knuthem jako souputník TEXu. Zatímco TEX definuje rozložení znaků na stránce, METAFONT definuje tvary znaků (glyfy) a vztahy mezi nimi. METAFONT přesně vymezuje velikosti glyfů a grafický rastr k jejich reprezentaci. Informaci o velikosti glyfů využívá TEX, zatímco rastr využívají programy, které produkují tisknutelný výstup, což je další zpracování toho, co vyprodukuje TEX.

Jazyk METAFONTu pro definici fontu dovoluje vyjádřit několik věcí: za prvé (ovšem) geometrické tvary glyfů, za druhé vlastnosti tiskového zařízení, pro které je určen výstup, a za třetí, „meta“-informaci, pomocí které lze rozlišit různé velikosti stejného fontu nebo rozdíly mezi dvěma fonty, které patří do stejné (nebo příbuzné) rodiny.

Knuth (a ostatní) navrhli velmi mnoho fontů s použitím METAFONTu, ale návrh fontů pomocí METAFONTu je mezi uživateli rozšířen mnohem méně, než psaní TEXovských maker. Dokonalý uživatel TEXu nicméně potřebuje vědět o METAFONTu a být schopný spouštět METAFONT ke generování vlastních souborů s fonty v různých velikostech a rozlišeních.

4. Čo je to METAPOST

METAPOST je systém, ktorého jazyk je veľmi podobný METAFONTu až na to, že jeho výstup je PostScript a nie bitové mapy. METAPOST je silný jazyk pre kreslenie obrázkov do dokumentov,

[3] v originále „literate programming“, což se také někdy překládá jako „kultivované“ nebo „gramotné“ programování

[4] International Phonetic Alphabet, mezinárodní abeceda pro zápis výslovnosti.

ktoré budú tlačené na PostScriptovej tlačiarni. Poskytuje prístup ku všetkým možnostiam PostScriptu a umožňuje spojenie textu a obrázkov. (D. Knuth povedal, že nepoužíva nič iné na kreslenie diagramov, keď píše nejaké texty.)

Veľká časť zdrojového kódu METAPOSTu bola s povolením D. Knutha kopírovaná z METAFONTu.

5. Co je to LaTeX

LaTeX je balík maker TeXu napsaný Leslie Lamportem, který představuje systém pro zpracování dokumentu. LaTeX dovoluje popsat strukturu dokumentu pomocí značkování tak, aby uživatel nebyl nucen přemýšlet o výsledném vzhledu. S použitím dokumentových tříd a přídavných balíků může být tentýž dokument vysázen v mnoha různých podobách.

Lamport říká, že LaTeX *„představuje vyváženost mezi funkčností a snadností použití"*. Toto se objevuje jako nepřetržitý konflikt vedoucí k potřebě článků podobných tomuto: LaTeX *může* splnit většinu uživatelských požadavků, ale vědět *jak* není jednoduché.

6. Jak mám vyslovovat „LaTeX", „LaTeX 2_ε"

Lamport nikdy nedoporučil, jak sa má vyslovovat LaTeX, ale mnoho lidí jej vyslovuje „La TeX" nebo snad „LejTeX" (kde TeX je vyslovován jako samotný program, viz Otázku **??**).

O písmenku „epsilon" v LaTeX 2_ε se předpokládá, že bude naznačovat malé zlepšení starého LaTeXu 2.09. Nicméně většina lidí vyslovuje toto jméno jako „LaTeX-dvě-é".

7. Mám používat `plain`TeX nebo LaTeX

Na tuto otázku neexistuje přímá odpověď. Mnoho lidí nedá dopustit na `plain`TeX a s jeho použitím produkuje velmi obdivuhodné dokumenty (Knuth je samozřejmě příkladem). Ale stejně tak mnoho lidí rádo přenechá rozhodování o designu někomu jinému a přistoupí na malou ztrátu flexibility výměnou za ušetření duševního výkonu.

Úvahy kolem tohoto tématu mohou vyvolat vášnivé diskuse, aniž by vnesly do této věci jasno. Nejlepší je zjistit, co používají ti okolo, a jít s davem. Později se člověk vždycky může přiklonit k něčemu jinému. Nemá cenu se tím trápit.

Jestliže připravujete rukopis pro vydavatele nebo časopis, zeptejte se, jaké značkování se požaduje, ještě předtím, než vyvinete svoje vlastní. Mnoho velkých vydavatelů vyvinulo své vlastní LaTeXovské styly pro časopisy a knihy a trvají na tom, aby se autoři jejich značkování pevně drželi.

8. Co to jsou AMS balíky ($\mathcal{AMS}$-TeX, atd.)

$\mathcal{AMS}$-TeX je TeXovský balík maker původně napsaný Michaelem Spivakem pro Americkou matematickou společnost (AMS) během let 1983 – 1985. Je popsán v knize *The Joy of TeX* od Michaela D. Spivaka (druhé vydání, AMS, 1990, ISBN 0-821-82997-1). Je založen na `plain`TeXu, ale poskytuje mnoho možností pro tvorbu, které vypadají více profesionálně a méně zatěžují autory. Věnuje se pozornost jemnějším detailům nastavování velikosti a umísťování, na které dbají vydavatelé matematické literatury. Je brán ohled na víceřádkové samostatně sázené rovnice, číslování rovnic, tečky, matice, dvojité akcenty, víceúrovňové indexování, konrolu syntaxe (rychlejší zpracování při počátečních průchodech TeXu pro kontrolu chyb), a jiné věci.

Jakmile LaTeX získal na popularitě, autoři chtěli posílat AMS články v LaTeXu, a tak AMS vyvinula $\mathcal{AMS}$-LaTeX. Ten zahrnuje LaTeXovské balíky a třídy a nabízí autorům většinu schopností $\mathcal{AMS}$-TeXu.

9. Co je to LAMS-TeX

LAMS-TeX byl spojením $\mathcal{AMS}$-TeXu a LaTeXu od Michaela Spivaka. Jeho silné stránky jsou:

- balík pro komutativní diagramy, který produkuje velmi uspokojivý výstup;

- samostatný program *dvipaste* pro přípravu složitých tabulek odděleně od hlavního dokumentu (čímž se předejde problémům s překročením kapacity hlavní paměti TEXu);
- rozsáhlá kontrola na uživatelské úrovni týkající se formátování automaticky generovaných čísel.

Nicméně LAMS-TEX se objevil v době, kdy byl vydán AMS-LATEX, a tak se LAMS-TEX nikdy skutečně neuchytil.

10. Co je to Eplain

Balík maker Eplain rozšiřuje a doplňuje definice v `plain` TEXu. Eplain není zamýšlen k tomu, aby poskytoval „generické typografické nástroje", tak jako to činí LATEX nebo Texinfo viz Otázku **??**. Namísto toho poskytuje definice, které mají být užitečné bez ohledu na příkazy vyšší úrovně, které se vlastně používají při přípravě rukopisu.

Např. Eplain nemá příkaz `\section`, který by sázel záhlaví oddílů „vhodným" způsobem, tak jako `\section` v LATEXu. Filosofie Eplainu je, že někteří lidé budou vždy potřebovat nebo chtít jít za názor tvůrce makra na to, co je „vhodné". Taková konzervovaná makra jsou dobrá — dokud je člověk ochoten akceptovat výsledný výstup. Pokud výsledek není vyhovující, nebo se člověk pokouší o jiný formát, narazí na těžkosti.

Na druhé straně, takřka všichni by uvítali takové možnosti, jako je vytváření křížových referencí návěštími tak, aby nebylo třeba dávat skutečná čísla stránek do rukopisu. Karl Berry, autor Eplainu, říká, že neví o žádných obecně dostupných balících maker, které by nevnucovaly jejich typografický styl autorovi a přesto poskytovaly takové možnosti.

11. Co je to Lollipop

Lollipop je balík maker napsaný Victorem Eijkhoutem. Byl použit při přípravě knihy *„TEX by Topic"* (viz Otázku **??**). Manuál o něm říká:

> Lollipop je „usnadněný TEX". Lollipop je balík maker, který funguje jako nástroj pro psaní TEXovských maker. Bylo mým úmyslem udělat psaní maker tak jednoduché, aby implementace zcela nového vzhledu dokumentu v TEXu se stala záležitostí méně než hodiny pro průměrný dokument a aby byla úkolem, který by mohl být vykonán někým, kdo má pouze velmi základní znalosti programování v TEXu.
> Lollipop je pokusem zpřístupnit formátování strukturovaného textu pro prostředí, ve kterých předtím mohly být používány pouze WYSIWYG balíky, protože přizpůsobení vzhledu dokumentu je s nimi o mnoho jednodušší než s tradičními TEXovskými balíky maker.

Manuál pokračuje rozpravou o ambicích „získat část trhu LATEXu". Je to velmi důmyslný balík, ale je pozorovatelný určitý náznak přebírání z LATEXu... Článek o Lollipopu vyšel v TUGboatu 13(3), 1992.

12. Co je to Texinfo

Texinfo je dokumentační systém, který používá jeden zdrojový soubor k vytváření on-line informace i tištěného výstupu. Namísto psaní dvou různých dokumentů, jednoho pro on-line help a druhého pro tištěný manuál, je třeba napsat pouze jeden zdrojový soubor dokumentu. Když je práce revidována, je potřeba revidovat pouze jeden dokument. Je možné číst on-line informaci, známou jako „soubor Info", pomocí programu Info pro čtení dokumentace. Podle konvence jména zdrojových souborů Texinfo končí s příponou `.texi` nebo `.texinfo`. Je možné psát a formátovat Texinfo soubory do Info souborů v GNU *emacs*u a číst je pomocí zabudovaného Info prohlížeče. Pokud není k dispozici *emacs*, je možné formátovat Texinfo soubory do Info souborů pomocí programu *makeinfo* a číst je pomocí programu *info*.

Množina TEXovských maker pro formátování Texinfo souborů je dostupná v `macros/texinfo`

13. Je-li TEX tak dobrý, jak to, že je volně šiřitelný

Je volně šiřitelný z rozhodnutí jeho autora D. Knutha. Nicméně podle všeho Knuth není zrovna šťastný, že ostatní sklízejí peníze za to, že poskytují služby a produkty založené na TEXu. Některé hodnotné nástroje a balíky týkající se TEXu jsou pod ochranou autorských práv podle tzv. GNU General Public Licence („Copyleft"), ale TEX sám takto chráněn není.

Jsou k dostání komerční verze TEXu. Některé uživatele uklidňuje, pokud mají placenou podporu. Navíc některé komerční implementace mají schopnosti, které volně šiřitelné verze nemají. (Ale platí to i naopak: některé volně šířené implementace nabízejí věci, které žádná komerční neumí.)

Obvykle tento článek nepopisuje komerční verze. Otázka ?? podává přehled hlavních dodavatelů.

14. Jaká je budoucnost TEXu

Knuth prohlásil, že už nebude TEX dále vyvíjet. Chce pouze opravovat všechny chyby, o kterých se dozví (ačkoliv těchto chyb je málo). Toto rozhodnutí udělal krátce poté, co byla vydána verze TEXu 3.0. Při každém dalším vydání, ve kterém jsou opraveny některé chyby, je číslo verze doplněno o jednu další číslici tak, že konverguje k číslu π. Knuth chce, aby po jeho smrti byl TEX zmrazen na verzi π Po tomto okamžiku už nesmějí být prováděny žádné další změny v Knuthově zdrojovém kódu. (Podobné pravidlo je aplikováno na METAFONT. Jeho číslo verze konverguje k číslu e.)

Existují projekty (některé z nich jsou dlouhodobé, viz např. Otázky ?? a ??) na vybudování od základu nového balíku maker založeného na TEXu. Pro ještě delší časové období existují různé projekty na vybudování *následovníka* TEXu. Viz Otázky ?? a ??.

15. Co je to TUG a TUGboat

TUG je zkratka za TEX Users Group (česky skupina uživatelů TEXu). TUGboat je název hlavního časopisu organizace TUG, který osahuje užitečné články o TEXu a METAFONTu. TUG rovněž vydává pro členy noviny TEX and TUG News, každý rok organizuje konferenci, provádí školicí kursy, prodává většinu knih věnovaných TEXu a distribuuje na disketách TEXovský software pro mikropočítače. TUG má Technickou Radu (Technical Council) pro koordinaci činností kolem TEXu. (Viz Otázku ??). Dotazy mohou být směrovány na:

TEX Users Group
1466 NW Front Avenue, Suite 3141
Portland, OR 97209
USA

Tel: (+1) 503-223-9994
Fax: (+1) 503-223-3960
E-mail: `tug@tug.org`
Web: `http://www.tug.org/`
CTAN: `usergrps/tug`

16. Jsou také národní skupiny uživatelů TEXu

Následující skupiny publikují informace z jejich členské základny elektronicky v archívech CTANu:

DANTE, Deutschsprachige Anwendervereinigung
TEX e.V.
Postfach 10 18 40
D-69008 Heidelberg
Germany

Tel: (+49) 06221 2 97 66
Fax: (+49) 06221 16 79 06
E-mail: `dante@dante.de`

Web: `http://www.dante.de/`
CTAN: `usergrps/dante`

GUTenberg,
BP 10,
93220 Gagny principal,
France

`gut@irisa.fr`
Web: `http://www.ens.fr/gut/`
CTAN details: `usergrps/gut`

NTG
Postbus 394, 1740AJ Schagen,
The Netherlands

E-mail: `ntg@nic.surfnet.nl`
Web: `http://ei0.ei.ele.tue.nl/ntg/ntg.html`
CTAN details: `usergrps/ntg`

UK TEX Users' Group,
c/o Peter Abbott,
1 Eymore Close,
Selly Oak,
Birmingham B29 4LB
UK

Tel: (+44) 0121 476 2159
E-mail: `UKTuG-Enquiries@tex.ac.uk`
Web: `http://www.tex.ac.uk/UKTUG/home.html`
CTAN details: `usergrps/uktug`

Seznam všech známých skupin uživatelů je k dispozici v dokumentu `usergrps/info/usergrps.tex`, informace o CSTUGu viz Otázku ??.

17. Technické pracovní skupiny TUGu

TUG (viz Otázku ??) má autonomní Technickou Radu, která dohlíží na několik pracovních skupin spolupracujících na společném zájmu TEXového společenství. Rada má tři členy (současným předsedou je Michael Ferguson, kterému asistují Yannis Haralambous a Sebastian Rahtz), kteří mají spojení s vedoucími každé pracovní skupiny. Každá skupina si určuje svoje vlastní pracovní metody i členství a kdokoliv se zájmem spolupracovat může kontaktovat vedoucího. Návrhy na nové skupiny mají být adresovány na Michaela Fergusona (`mike@inrs-telecom.uquebec.ca`).

Následuje stručný přehled aktivních skupin:

WG-92-00 (IRP-TWG) *Nezávislý výzkumný projekt TWG.*
Pro uznání důležitých projektů nezávislých na TUGu, ale týkajících se celé TEXové obce.
Kontakt: Alan Hoenig (`ajhjj@cunyvm.cuny.edu`)

WG-92-01 *Rozšířené kódování matematických fontů v TEXu.*
Pro vytváření standardů na kódování matematických fontů, které se používají v TEXovských systémech.
Kontakt: Barbara Beeton (`bnb@math.ams.org`)

WG-92-03 *Vícejazyčné dokumenty.*
Primárním účelem této pracovní skupiny je získat pro TEXovské systémy konzistentní prostředky pro implementaci, přístup a popis fontů, pravidel pro slitky, dělících vzorů a jiných speciálních požadavků na danou lingvistickou skupinu.
Kontakt: Yannis Haralambous (`Yannis@pobox.com`)

WG-92-04 *TEX pro postižené.*
Primárním účelem této pracovní skupiny je fórum pro ty, kteří se zajímají o používání a/nebo vylepšení TEXu, aby uspokojili potřeby lidí se zrakovou nebo jinou vadou.
Kontakt: T.V. Raman (`raman@adobe.dec.com`)

WG-92-05 *Směrnice pro archív TEXu.*
Účelem této technické pracovní skupiny je vyvinout směrnice pro efektivní management a používání hlavních TEXovských archívů a iniciovat komunikaci mezi správci existujících archívů za účelem koordinace a synchronizace.
Kontakt: George Greenwade (`bed_gdg@shsu.edu`)

WG-94-07 *Struktury adresářů TEXu.*
Primárním účelem této technické pracovní skupiny je najít univerzální strukturu adresářů pro makra, fonty a jiný TEXovský software tak, aby mohla být přijímána doporučení pro všechny dodavatele TEXovského softwaru.
Kontakt: Norm Walsh (`norm@ora.com`)

WG-94-08 *Otázky implementace a standardizace DVI ovladačů.*
Hlavním cílem by mělo být zkoumání nových požadavků na DVI ovladače vynucených změnami potřeb a technologií a dále přijímání doporučení pro implementaci a standardizaci takových ovladačů, aby se zvýšila jednotnost jejich použití. Práce bude zahrnovat, ale nebude se omezovat, na prověřování použitelnosti, syntaxe a sémantiky příkazů `\special{...}`.
Kontakt: Michael Sofka (`sofkam@rpi.edu`)

WG-94-09 *TEX a SGML.*
Hlavním cílem je zkoumat požadavky a těžkosti ve vývoji technologie rozhraní pro TEX a SGML.
Kontakt: Ken Dreyhaupt (`kend@springer-ny.com`)

WG-94-10 *TEX a lingvistika.*
Hlavním cílem je studovat a diskutovat požadavky pro sázení lingvistiky v TEXu a prostředky identifikace, prověřování, testování a porovnávání maker, fontů, souborů se styly a jiných pomůcek pro sázení lingvistiky.
Kontakt: Christina Thiele (`cthiele@ccs.carleton.ca`)

C. Dokumentace a nápověda

18. Knihy o TEXu a příbuzná literatura

Knuthova kniha je vyčerpávající referenční příručkou pro TEX, ale existují i jiné knihy pokrývající TEX:

The TEXbook od Donalda Knutha (Addison-Wesley, 1984, ISBN 0-201-13447-0, brožované ISBN 0-201-13448-9)

A Beginner's Book of TEX od Raymonda Seroula and Silvia Levyho, (Springer Verlag, 1992, ISBN 0-387-97562-4)

Introduction to TEX od Norberta Schwarze (Addison-Wesley, 1989, ISBN 0-201-51141-X)

A PlainTEX Primer od Malcolma Clarka (Oxford University Press, 1993, ISBNs 0-198-53724-7 tvrdá vazba, a 0-198-53784-0 brožované)

TEX by Topic od Victora Eijkhouta (Addison-Wesley, 1992, ISBN 0-201-56882-9)

TEX for the Beginner od Wyntera Snowa (Addison-Wesley, 1992, ISBN 0-201-54799-6)

TEX for the Impatient od Paula W. Abrahamse, Karla Berryho a Kathryn A. Hargreavesové (Addison-Wesley, 1990, ISBN 0-201-51375-7)

TEX in Practice od Stephana von Bechtolsheima (Springer Verlag, 1993, 4 svazky, ISBN 3-540--97296-X celek, nebo Vol. 1: 0-387-97595-0, Vol. 2: 0-387-97596-9, Vol. 3: 0-387-97597-7, a Vol. 4: 0-387-97598-5).

TEX: Starting from $\boxed{1}$[5] od Michaela Dooba (Springer Verlag, 1993, ISBN 3-540-56441-1)

K LATEXu viz:

LATEX, a Document Preparation System od Leslieho Lamporta (druhé vydání, Addison Wesley, 1994, ISBN 0-201-15790-X)

A guide to LATEX od Helmuta Kopky a Patricka W. Dalyho (Addison Wesley, ISBN 0-201-56889-6)

[5] To znamená „Starting from Square One“.

The LaTeX Companion od Michela Goossense, Franka Mittelbacha, a Alexandera Samarina (Addison Wesley, 1993, ISBN 0-201-54199-8)

LaTeX Notes: Practical Tips for Preparing Technical Documents od J. Kennetha Shultise (Prentice Hall, 1994, ISBN 0-131-20973-6)

LaTeX Line by Line od Antoniho Dillera (John Wiley & Sons, 1993, ISBN 0-471-93471-2)

LaTeX for Scientists and Engineers od Davida J. Buergera (McGraw-Hill, 1990, ISBN 0-070-08845-4)

Math into TeX: A Simplified Introduction using AMS-LaTeX od George Grätzera (Birkhäuser, 1993, ISBN 0-817-63637-4)

Z tohoto seznamu pouze Lamport a Goosens s Mittelbachem a Samarinem pokrývají LaTeX 2_ε, nicméně jsou zprávy o tom, že se připravují aspoň některá nová vydání.

Seznam pro METAFONT je o dost kratší:
The METAFONTbook by Donald Knuth (Addison Wesley, 1986, ISBN 0-201-13445-4)

Kniha pokrývající široké spektrum témat (včetně instalace a udržování) je:
Making TeX Work by Norman Walsh (O'Reilly and Associates, Inc, 1994, ISBN 1-56592-051-1)

19. Kde lze najít tento dokument

Tento dokument vychází z jiného dokumentu, který je každý měsíc zasílán do usenetové skupiny `comp.text.tex`, a dále též do `news.answers` a `comp.answers`. Jeho aktuální verze je na CTANu v `help/TeX-FAQ` Je též archivován na všech místech, která archivují `news.answers`, jako např. `rtfm.mit.edu` (18.181.0.24). Zde je dostupný anonymním `ftp` (v adresáři `pub/usenet/news.answers/tex-faq`). Pro ty, co nemají přístup k `ftp`, ale jen k emailu, je zde možnost poslat zprávu obsahující text „`SENDME FAQ`" na adresu `fileserv@shsu.edu`.

Jistá verze tohoto dokumentu je na `http://www.cogs.susx.ac.uk/cgi-bin/texfaq2html?introduction=yes`[6]

20. Diskusní skupiny o TeXu a příbuzných programech

Existují (stále) lidé, kteří pracují na síti, ale nemohou číst Usenet news. Ani pro ně není vše ztraceno, pokud mohou posílat a příjmat elektronickou poštu.

TeXhax je moderovaná elektronická konference typu digest. Chcete-li se k ní připojit, pošlete zprávu „`subscribe texhax`" na adresu `texhax@tex.ac.uk`.

Usenetová skupina `comp.text.tex` (dost objemná) přebírá příspěvky z elektronické konference `info-tex`. Do ní se můžete přihlásit posláním zprávy
`subscribe info-tex <vaše jméno>`
na `listserv@shsu.edu`.

Zmíněná usenetová skupina je též dostupná ve formě digestu jako elektronická konference `ctt-digest`. Přihlásit se lze posláním zprávy „`subscribe ctt-Digest <vaše jméno>`" na `listserv@shsu.edu`. `ctt-Digest` zasílá zhruba stejné množství informace jako `info-tex`, ale počet zpráv je menší, což je na některých systémech výhoda.

Několik elektronických konferencí věnovaných TeXu je přístupných přes `listserv@vm.urz.uni-heidelberg.de`. Pošlete zprávu obsahující slovo „`help`" na výše uvedenou adresu.

Usenetová skupina pro dokumentované programování (viz Otázku ??) `comp.programming.literate` je zrcadlena do elektronické konference `litprog`. Přihlášení se provede zasláním zprávy
`subscribe litprog <vaše jméno>`
na `listserv@shsu.edu`.

21. Dokumentace k BibTeXu

BibTeX je program původně určený k produkci bibliografií ve spojení s LaTeXem. Je popsán v oddíle 4.3 a příloze B v příručce o LaTeXu od Leslieho Lamporta (viz Otázku ??). Úplnější popis podává

[6] Toto je dočasné URL. Definitivní umístění tohoto dokumentu bude oznámeno, až přijde čas.

dokument „BibTeXing“ obsažený v souboru `btxdoc.tex`. *The LaTeX Companion* (viz Otázku ??) rovněž obsahuje informace o BibTeXu a psaní BibTeXových stylů.

Dokument „Designing BibTeX Styles“ je v souboru `btxhak.tex` a popisuje postfixový zásobníkový jazyk, který se používá pro psaní BibTeXovských stylů (soubory `.bst`). Soubor `btxbst.doc` je předlohou pro čtyři standardní styly (`plain`, `abbrv`, `alpha`, `unsrt`). Obsahuje též jejich dokumentaci. Úplná kolekce dokumentace k BibTeXu (zahrnující i zmíněné soubory) je v `biblio/bibtex/distribs/doc`

Existuje též unixová manuálová stránka BibTeX v balíku *web2c* (viz Otázku ??). Nepoužívejte kopie zastaralých manuálových stránek, napsaných v roce 1985, dříve než se objevily dokumenty „BibTeXing“ a „Designing BibTeX Styles“.

22. Manuál k programu PICTeX

PICTeX je soubor maker pro kreslení diagramů a obrázků. Tato makra jsou volně dostupná v `graphics/pictex`. Nicméně manuál k PICTeXu není volně šiřitelný. Je k dostání za 30 $ (35 $ i s disketou) u TeX Users Group (viz Otázku ??). Zisky z prodeje jdou autorovi PICTeXu, kterým je Michael Wichura, a organizaci TUG.

23. Hledání TeXovských/LaTeXovských balíků maker

Než se budete ptát po TeXovském makru nebo LaTeXovské třídě nebo balíku, abyste udělali „to a to“, zkuste prohledat index TeXovských maker, jehož autorem je David M. Jones (`dmjones@theory.lcs.mit.edu`), a který je dostupný v `help/Tex-index/TeX-index`

Tento index je vynikající referenční příručkou se spoustou křížových odkazů, ale je poněkud neaktuální. Dokud nebude dokončena práce na jeho aktualizaci, doporučuje se ignorovat v něm všechny informace o umístění souborů. Najdete-li soubor, který se zdá být zajímavý, zjistěte, zdali je na CTANu (viz dále a viz Otázku ??).

CTAN archívy (viz Otázku ??), můžete prohledávat po navázání anonymního `ftp` spojení s nějakým CTAN počítačem, když napíšete příkaz „`quote site index <term>`“. Archív vygeneruje seznam souborů, v jejichž jménech[7] je obsažen řetězec `<term>`. (Někteří `ftp` klienti nemají implementován příkaz „`quote`“. Ve většině těchto případů stačí napsat „`site index <term>`“.)

Katalog balíků ve tvaru web-stránky je na `ftp://ftp.cstug.cz/pub/tex/CTAN/help/Catalogue/catalogue.html`

24. Hledání balíků LaTeX 2_ε

Uživatel může být zmaten tím, že nenajde na CTANu žádné `.sty` nebo `.cls` soubory příslušející k balíku nebo třídě, která mu byla doporučena. Například existuje mnoho balíků, které jsou uvedeny v knize *The LaTeX Companion* (viz Otázku ??) bez podrobností o jejich umístění na CTANu. Soubor `info/companion.ctan` by mohl pomoci při hledání takových souborů, ale není to univerzální řešení.

Balíky a třídy LaTeX 2_ε se kanonicky distribuují jako `.dtx` soubory (??). Pokud zadáte příkaz „`site index`“ s parametrem „`package.sty`“, nic nenajdete — musíte prostě hledat „`package`“ (zadaný řetězec se bere jako podřetězec vyhledávaných jmen).

V případě, že to, co hledáte, je opravdu tímto způsobem distribuováno, měli byste si přenést celý adresář obsahující hledaný `.dtx` soubor, čímž získáte všechny doprovodné soubory, především jeden nebo více `.ins` souborů (installation). Zpracováním „vhodného“ `.ins` souboru pomocí LaTeX 2_ε se vygenerují všechny soubory, které potřebujete.

[7] V podstatě `<term>` je unixovský regulární výraz. Zadaným příkazem se prohledává indexový soubor s použitím příkazu *grep*

D. TEXové drobnosti

25. Čo je to DVI súbor

DVI súbor (teda súbor, ktorý má príponu `.dvi`) je hlavný výstupný súbor TEXu (použijúc TEX v širšom zmysle, zahŕňajúc v tom aj LATEX a ďalšie verzie a varianty TEXu). DVI znamená device independent, čo je nezávislý od zariadenia, a znamená to, že tento súbor by mal byť vytlačený rovnako, bez ohľadu na typ zariadenia, na ktorom to bolo vykonané. DVI súbor je čítaný ovládačom zariadenia (pozri Otázku ??) ako je tlačiareň (napr. LaserJet) alebo je vstupom pre prehliadač na obrazovke monitora. V DVI súbore je použité TEXovské vnútorné kódovanie. Z TEXovského vstupného súboru by sme mali urobiť vždy ten istý DVI súbor bez ohľadu na implementáciu TEXu, ktorú použijeme.

DVI súbor obsahuje všetky informácie, ktoré sú potrebné pre tlač alebo prezeranie, ale nie konkrétne bitové mapy alebo iné popisy písma. Snáď ešte materiál, ktorý je uvedený v príkaze `\special` (pozri Otázku ??).

Základný odkaz na štruktúru DVI súboru je zdrojový text *dvitype*.

26. Čo je to ovládač

Ovládač (driver) je program, ktorý má za vstup DVI súbor (pozri Otázku ??) a (obvykle) produkuje súbor, ktorý môže byť poslaný do typografického výstupného zariadenia, napríklad tlačiarne.

Ovládač je obvykle špecifický pre danú tlačiareň alebo triedu tlačiarní, aj keď každá PostScriptová tlačiareň by mala vytlačiť výstup PostScriptového ovládača.

Ovládač potrebuje DVI súbor, ako aj informáciu o písme. Písma môžu byť zadávané vo forme bitových máp alebo odkazom na písmo, ktoré je priamo obsiahnuté v tlačiarni. Každý ovládač očakáva informáciu o písme v špecifickej podobe. Viac informácií o formách písem môžete nájsť v otázkach ??, ??, ?? a ??.

27. Čo sú to PK súbory

PK súbory (od slov packed raster) obsahujú bitmapy písem. Výstup z METAFONT (pozri Otázku ??) obsahuje GF súbor (generic font). Tento konvertuje program *gftopk* do PK súboru. Existuje mnoho PK súborov, pretože pre každé písmo potrebujeme jeden. Dokonca pre každú bodovú veľkosť aj magnification – zväčšenie, každý typ každej rodiny písem. Navyše, PK súbor pre jednu tlačiareň nemusí nevyhnutne byť dobrý pre druhú tlačiareň. V takom prípade je treba mať celú množinu PK súborov aj pre túto ďalšiu tlačiareň. Výsledkom sú často komplikované adresárové štruktúry alebo FLI (font library file) súbory pre väčšiu usporiadanosť prístupu.

28. Čo sú to TFM súbory

TFM je skratka pre TEX font metric, čiže TFM súbor obsahuje informácie o rozmeroch charakterov (písmeniek), o ligatúrach (typografické spojenie dvoch alebo viacerých znakov) a o medziznakových medzerách v danom písme. Pre každé písmo je potrebný jeden TFM súbor, dokonca pre každú bodovú veľkosť. Ale jeden TFM súbor je pre všetky zväčšenia (magnifications), preto je vo všeobecnosti TFM súborov menej ako PK súborov. Dôležité je, že TFM súbory používajú programy TEX, LATEX, ale nie sú vo všeobecnosti potrebné napríklad pre ovládače tlačiarní.

29. Virtuální fonty

Virtuální fonty pro TEX jako první implementoval David Fuchs už na začátku jeho historie, ale pro většinu lidí začaly existovat, až když Knuth v roce 1989 změnil formát a napsal několik podpůrných programů. Použití virtuálních fontů je způsob, jak říci TEXu o něčem složitějším, než je vzájemně jednoznačné mapování znaků. Entity, které se definují ve virtuálním fontu, vypadají jako TEXovské znaky (objevují se s popisem jejich velikosti a souborem metriky fontu), ale `dvi` procesor je může expandovat do něčeho zcela odlišného. Tato schopnost se dá využít pro přeskládání znaků, pro vytvoření fontu poskládaného z glyfů pocházejících z různých zdrojových souborů, nebo obecně k vytváření různě složitých efektů — virtuální font může obsahovat cokoliv, co je přípustné v `dvi` souboru. Ve většině praktických případů se virtuální fonty používají k přeskládání postscriptových fontů (viz Otázku **??**) nebo k vytváření „nepravých“ matematických fontů.

Je důležité si uvědomit, že samotný TEX virtuální fonty *nevidí*. Pro každý virtuální font, který čte `dvi` ovladač, existuje odpovídající `tfm` soubor, který čte TEX. Virtuální fonty jsou normálně vytvořeny v jednom ASCII souboru (`vpl`, Virtual Property List), který obsahuje oba druhy informace. Program *vptovf* pak vytvoří binární `tfm` a `vf` soubory. Nejběžnějším způsobem (v dnešní době) jak vygenerovat `vpl` soubor, je použít programový balík *fontinst*, který je detailně popsán v Otázce **??**. Dalším prostředkem pro ad-hoc vytváření virtuálních fontů je `fonts/utilities/qdtexvpl` od Eberharda Mattese.

30. Příkazy typu `\special`

TEX poskytuje prostředy k vyjádření věcí, které mohou dělat ovladače zařízení, ale o kterých TEX sám neví nic. Například TEX neví nic o tom, jak zařadit postscriptový obrázek do dokumentu, nebo jak nastavit barvu tisknutého textu. Ale některé ovladače zařízení to umějí.

Takové věci jsou umožněny prostřednictvím příkazů `\special`. Jediná věc, kterou s takovým příkazem TEX udělá, je, že expanduje jeho argumenty a pak příkaz přepíše do `dvi` souboru. Ve většině případů jsou k dispozici balíky (často i s ovladačem), které poskytují srozumitelné rozhraní k příkazům `\special`. Například při vkládání obrázku do textu je nutno vynechat patřičnou mezeru. Anebo změna barvy může být náročnější operace vyžadující znalost triků. LATEX 2_ε má standardní balíky pro grafiku a barvu, s jejichž pomocí jsou vkládání souborů, rotace, změny měřítka a barvy prostřednictvím příkazů `\special` jednoduché.

Argumenty povolené pro příkaz `\special` závisejí na ovladači zařízení. Kromě předchozích příkladů existují příkazy `\special` pro ovladače z emTEXu (např. *dvihplj*, *dviscr*, *atd.*, které kreslí přímky s libovolným sklonem, a příkazy pro ovladač *dvitoln03*, které umožňují, aby stránka byla vysázena naležato).

31. Dokumentované LATEXovské zdroje (`.dtx` soubory)

LATEX 2_ε a spousta podpůrných balíků maker jsou nyní psány metodou dokumentovaného programování (viz Otázku **??**), se zdrojovým kódem i dokumentací ve stejném souboru. Autorem tohoto formátu, který je znám jako „doc“, je Frank Mittelbach. Dokumentované zdroje mají podle konvence koncovku `.dtx` a dokumentace by z nich měla být před používáním odstraněna. Případně lze soubor `.dtx` zpracovat LATEXem, čímž vznikne pěkně zformátovaná verze dokumentovaného kódu. Většinou je k dispozici instalační dávka (s příponou `.ins`), která potřebuje standardní balík LATEX 2_ε *docstrip*. Ten odstraňuje všechny poznámky, jejichž obsahem je právě ta dokumentace. V průběhu instalace se (kromě jiných věcí) odstraňuje ze souborů jejich dokumantace, aby se zvýšila rychlost čtení těchto souborů běžícím LATEXem. Do jednoho `.dtx` souboru může být zahrnuto více balíků, s podmíněnými sekcemi a prostředky pro indexy maker atd. Soubory `.dtx` může psát každý. Jejich formát je popsán v *The LATEX Companion* (viz Otázku **??**). Zatím neexistují programy podporující psaní `.dtx` souborů.

Poté, co byly z `.dtx` souborů vytvořeny `.sty` nebo `.cls` (a další) soubory, nejsou už soubory `.dtx` LATEXem používány. Nemusí být uchovávány s fungujícím systémem. Nicméně pro mnoho balíků jsou primárním zdrojem dokumentace, takže je někdy dobré si je ponechat.

32. Čo sú to DC písma

Písmo (font) pozostáva z množiny znakov (písmen, číslic a špeciálnych znakov). Aby sme ich mohli vytlačiť, musíme mať nadefinovaný prístup k nim. V TEXu sú usporiadané v určitom poradí, ktoré voláme kódovanie. Z rôznych príčin D. Knuth zvolil trochu výstredné kódovanie. Konkrétne, pre rôzne písma zvolil rôzne kódovania.

Po uvolnení TEXu, verzie 3, boli niektoré „výstrednosti“ odstránené. Na stretnutí TUGu (TEX User Group) v Corku bolo definované nové kódovanie pre množinu 256 znakov. Cieľom bolo pokryť väčšinu európskych jazykov, v zmysle zahrnutia všetkých možných akcentovaných znakov (napríklad Knuthovo kódovanie nepokrýva poľské alebo islandské akcentované znaky a Corkovské kódovanie áno). LATEX 2_ε (pozri Otázku ??) umožňuje použiť Corkovské kódovanie (T1), a tak sa umožňuje vyhnúť konfliktom a problémom s delením slov s akcentami (pozri Otázku ??).

METAFONTovské písma, ktoré sú v súlade s Corkovským kódovaním, sa volajú DC písma (dostupné ako `fonts/dc`; ubezpečte sa, že máte verziu 1.2, úroveň opráv 1, z decembra 1995 alebo neskoršiu). Tieto písma vyzerajú ako Computer Modern písma. Mali by byť chápané ako medzikrok ku písmam EC (ktoré budú, ako dúfam, dostupné niekedy v roku 1996). Ich vážnou nevýhodou pre náhodného používateľa je, že súbory sú veľké – každé DC písmo je zhruba dvakrát väčšie ako zodpovedajúce CM písmo. Preto a pokiaľ zodpovedajúce matematické písma nie sú vytvorené, musia byť CM písma zachované.

Corkovské kódovanie je implementované aj v PSNFSS systéme (pozri Otázku ??) pre PostScriptové písma.

E. Získávání softwaru

33. Co je CTAN

Za účelem snadné archivace a získávání TEXovského softwaru vyvinula pracovní skupina TUGu tzv. Comprehensive TEX Archive Network (CTAN). Každý počítač zapojený v síti CTAN obsahuje stejný materiál a udržuje jeho aktuální verze. Tato data jsou dosti rozsáhlá. Především všechno, co je zmíněno v těchto otázkách, je většinou na CTANu archivováno.

Servery s CTANem jsou například `ftp.dante.de` (129.206.100.192) nebo `ftp.tex.ac.uk` (128.232.1.87). Struktura TEXových souborů je na všech těchto místech identická a začíná v `tex-archive/`. Pro snížení zatížení počítačové sítě používejte prosím vám nejbližší CTAN počítač nebo jeho kopii — „zrcadlo“ (viz Otázku ??). Kompletní aktuální seznam CTAN počítačů a zrcadel je dostupný jako soubor `CTAN.sites`

Pro hledání softwaru na CTANu lze použít anonymní `ftp` (viz Otázku ??).

Existuje též kolekce DECUS TEXovského materiálu pro VMS, Unix, MS-DOS a Macintosh. Materiál pro VMS není udržován up-to-date, ale běží i pod OpenVMS na počítačích VAX. Je dostupný přes anonymní `ftp` na `wuarchive.wustl.edu` (128.252.135.4) v `decus/tex/`. V USA ho lze získat i z DECUS knihovny (referenční číslo VS0058) nebo mimo USA přes úřad DECUSu. Bližší informace na

DECUS
LIBRARY ORDER PROCESSING
334 South Street, SHR3-1/T25
Shrewsbury, MA 01545-4195
USA

Tel: 800-DECUS55 (within the USA, for information)
Fax: (+1) 508-841-3373 (for inquiries)

nebo pošlete e-mail editorovi DECUS TEX sbírky, kterým je Ted Nieland (`nieland@ted.hcst.com`).

Konečně, uživatel TEXu, který nemá žádný přístup k síti, si samozřejmě může zakoupit kopii archívu na CD-ROM (viz Otázku ??).

34. Jak umístit vlastní příspěvek do archívů

Použijte anonymní `ftp` na jakýkoliv CTAN počítač (viz Otázku ??) a stáhněte si soubor `README.uploads` v kořenovém adresáři. Obsahuje instrukce pro umísťování souborů a informování lidí, kteří mají místní archív na starosti.

Nemůžete-li použít anonymní `ftp`, pošlete svůj příspěvek e-mailem na `ctan@urz.uni-heidelberg.de`. Všem ulehčíte život, jestliže vyberete pro svůj produkt trefné a jednoznačné jméno, takže by bylo dobré předem zkontrolovat, jestli se jméno vašeho `.sty` souboru již náhodou nepoužívá, což lze prostřednictvím příkazu „`site index`" (viz Otázku ??). Mějte na paměti, že veřejně lze vystavovat pouze software, jehož autoři proti tomu nemají námitek.

35. Hledání nových fontů

Podrobný seznam METAFONT fontů posílá do usenetových skupin `comp.fonts` a `comp.text.tex` zhruba každých šest týdnů Lee Quin (`lee@sq.sq.com`). Je též dostupný jako `info/metafont-list`

V seznamu jsou informace jak o komerčních, tak volně šiřitelných fontech. Většina z nich je dostupná anonymním `ftp` z CTAN archívů (viz Otázku ??).

36. TEX na CD-ROMech

Jestliže nemáte přístup na Internet, můžete obdržet kopii CTAN archívu na CD-ROMu. Je to i pro ty, kteří si prostě myslí, že mít 600MB souborů kolem TEXu po ruce je prima. Prime Time Freeware vyrobil *TEXcetera* 1.1 v červenci 1994, což

je kopie CTANu z června 1994. Pravidelné aktualizace jsou v plánu. Materiál je komprimován v ZIP formátu, aby se vešel na jedno médium a aby se obešlo omezení adresářů systému souborů dané normou ISO 9660. CD-ROM můžete zakoupit od:

Prime Time Freeware
370 Altair Way, Suite 150
Sunnyvale CA 94086
USA
Tel: (+1) 408 433 9662
Fax: (+1) 408 433 0727
E-mail: `ptf@cfcl.com`

nebo od redistributorů nebo od organizace TUG (viz Otázku ??). Cena je kolem 60 $. Nezapomeňte prosím, že PTF není velká komerční firma a že je dobrým přítelem TEXovské komunity.

Chcete-li TEX systém na CD-ROM, který je připraven ke spuštění, je k dispozici jen pro DOS (zatím). Holandská skupina uživatelů TEXu (NTG) publikuje tzv. 4AllTEX na CD-ROMu, kde je sbalen kompletní TEXovský software pro MS-DOS, všechna makra a všechny fonty, které si můžete přát. Je dostupné přímo od NTG (viz Otázku ??) nebo od TUG za 40 $ a od UK TUG za £30 (včetně manuálu). Je to užitečný zdroj nejen pro uživatele DOSu.

Viz též otázku ??.

F. Systémy TEX

37. TEX/LATEX pro různé počítače

Uvádíme seznam volně šiřitelných nebo shareware balíků. Informace o komerčních balících viz Otázku ??.

Unix Instrukce pro získávání unixovských distribucí TeXu přes anonymní `ftp` jsou dostupné v `systems/unix/unixtex.ftp`

AIX TEX pro IBM RS6000 s operačním systémem AIX je dostupný v `systems/unix/aix3.2`

386/ix Binární tvary programů pro 386/ix jsou v `systems/unix/386ix`

Linux Existují přinejmenším dvě dost úplné implementace TEXu pro Linux. Distribuce Slackware zahrnuje NTEX (dostupný na CTANu v `systems/unix/linux/ntex`), který asi obsahuje víc TEXovského materiálu, než by si člověk mohl přát. Nejnovější teTEX (dostupný jako `systems/unix/teTeX`) je založen na prohledávacím mechanismu cest od Karla Berryho a je víc kompaktní než NTEX, ale stále dost podrobný.

PC Balík emTEX pro PC s operačním systémem MS-DOS nebo OS/2 zahrnuje LATEX, BIBTEX, prohlížeče a ovladače a je k dispozici v `systems/msdos/emtex` jako řada obsahů disket. Autorem je Eberhard Mattes. Dokumentace je k dispozici jak v němčině, tak v angličtině. Uživatelé Windows nebo ti, kteří chtějí spouštět současný LATEX na počítačích s procesorem 386 a lepším, by si měli získat beta-test vydání emTEXu, a paměťový extender RSX (`systems/msdos/dpmigcc`).

Druhý balík je gTEX, který běží pod MS-DOSem nebo Windows (a má kladný uživatelský ohlas). Je dostupný z `systems/msdos/gtex`

TUG (a některé další pracovní skupiny) nabízejí všechen volně dostupný software pro PC. Katalog je volně k dispozici od TUG (viz Otázku ??).

Mac OzTEX je sharewarová verze TEXu pro Macintosh. Obsahuje i `dvi` prohlížeč a ovladač pro PostScript. Měla by běžet na jakémkoliv Macintosh Plus, SE, II, nebo novějším modelu, ale nebude běžet na 128K nebo 512K Macovi. Napsal ji Andrew Trevorrow, a je dostupná na disketách od TUG (viz Otázku ??). UK TUG předplácí sharewarový poplatek, takže členové UK TUGu mohou získat tento software bez dalších plateb. Otázky k OzTEXu mohou být směřovány na `oztex@midway.uchicago.edu`

Jiný částečně sharewarový program je CMacTEX (dostupný jako `systems/mac/cmactex`), který dal dohromady Tom Kiffe. Ten se více podobá unixovské implementaci (např. používá *dvips*).

VMS TEX pro VMS je možno nalézt v archívu CTAN v souboru `systems/vms/Alpha/tex_axp_exe.zip` (pro počítače založené na procesoru Alpha) nebo `systems/vms/VAX/tex_vax_exe.zip` (pro počítače VAX). Standardní distribuce na pásce je k mání u DECUS (viz Otázku ??).

Atari TEX je dostupný pro Atari ST v `systems/atari`

Nemáte-li přístup k `ftp`, pošlete zprávu obsahující slůvko „help" na `atari@atari.archive.umich.edu`

Amiga Úplné implementace TEXu 3.1 (PasTEX) a METAFONT 2.7 jsou dostupné v `systems/amiga`

Můžete si také objednat CD-ROM obsahující tento software a jiný software pro Amigu od Walnut Creek CDROM, telefon (+1) 510-947-5997.

TOPS-20 TEX byl původně napsán na počítače DEC-10 pod systémem WAITS a tak byl jednoduše portován na TOPS-20. Distribuce, která běží pod TOPS-20 je k dispozici anonymním `ftp` z `ftp.math.utah.edu` (128.110.198.34) v `pub/tex/pub/web`

38. Editory a shelly vhodné pro TEX

Pro většinu operačních systémů existují dobrá prostředí a editory pro psaní v TEXu. Některé jsou níže popsány, ale je to jen osobní výběr:

Unix Vyzkoušejte GNU *emacs* a AUCTEX mód (`support/auctex`). Poskytuje položky menu a řídicí sekvence pro běžné činnosti, kontroluje syntaxi, provádí vhodné značkování, dovoluje volat TEX a ovladače z prostředí editoru a všechno možné, na co člověk pomyslí. Složité, ale velmi mocné.

Existují však i editory velmi podobné *emacs*u, ale jednodušší a menší. Údajně dobrý je editor *jed*. Umí i zvýrazňování syntaxe (tj. změnu fontu pro různá slova) a má podporu pro AUCTEX.

VMS Mód *lsedit* pro editace TEXovských zdrojů je dostupný od TUG (viz Otázku ??) jako TEXniques 1, VAX Language-Sensitive Editor. Autorem je Kent MacPherson (1985). Pro VMS existuje i implementace *jed*u.

MS-DOS Zde je několik možností:

- Balík 4AllTEX (`systems/msdos/4alltex`) poskytuje velmi obsáhlé prostředí napsané ve 4DOS (za jeho používání se musí zaplatit sharewarový poplatek), které umožňuje pracovat s většinou TEXovského softwaru uživatelsky přítulným způsobem. Můžete si vybrat vlastní editor. Vhodný je např. *QEdit* nebo *Brief*. Celý balík je dostupný ve snadno použitelné formě na CD-ROMu od pracovních skupin TEXu.
- TEXshell (`systems/msdos/texshell`) je jednodušší, snadněji nastavitelné prostředí, ve kterém může být používán editor dle vlastní volby.

Můžete též použít GNU *emacs* a AUCTEX pod MS-DOSem, nebo MS-DOSovou implementaci *jed*u.

Windows Z volně přístupného softwaru je pravděpodobně nejlepší používat MicroEmacs jako editor a programy řídící TEX. Balík gTEX (`systems/msdos/gtex`) je součástí MicroEmacsu v ihned použitelné podobě. Obsahuje TEX, prohlížeč, *dvips* a *GhostScript*.

Na PC s dostatkem paměti lze používat verzi programu GNU *emacs* pro Windows. Takže pak můžete používat pod Windows i AUCTEX.

Také pod Windows existuje *jed*.

Komerční (a velmi kvalitní) prohlížeč pod Windows *dviwindo* od firmy Y&Y může být používán i jako dobrý TEX shell. Umí volat programy jako TEX, ovladače a editory (firma Y&Y dodává volně přístupný PE a doporučuje komerční Epsilon) z nastavitelných menu (pro více detailů o Y&Y viz Otázku **??**).

Scientific Word je editační program typu WYSIWYG, silný v matematice, který používá LATEXovský výstup (kontaktní adresa je v Otázce **??**).

OS/2 *Emacs* i *jed* fungují i pod OS/2. Dobrý je editor *EPM* (`http://www.sci.muni.cz/~mikulik/os2/pmCSTeX.html`). Makra jsou též součástí modulu `os2.zip` z distribuce CSTEXu (`http://math.feld.cvut.cz/olsak/cstex`).

Macintosh Komerční program Textures poskytuje vynikající integrované prostředí Macintoshe s vlastním editorem. Jako editor je však stále lepší shareware *Alpha* (`systems/mac/support/alpha`), který je dost rozšiřitelný na to, aby se dala provádět téměř jakákoliv činnost s TEXem. Pracuje dobře s OzTEXem.

Uživatelé počítačů Atari, Amiga a NeXT též mají hezká prostředí. Uživatelé LATEXu, kteří mají rádi příkaz „make“ mohou najít pomoc *např.* v `support/latexmk`

Existuje rovněž balík pomocných programů pro manipulaci s databází BIBTEXu.

39. Komerční implementace TEXu

Existuje mnoho komerčních implementací TEXu. První se objevila nedlouho poté, co se objevil TEX sám. Nejdéle přežili (od poloviny nebo začátku 80. let) dodavatelé ArborText (dříve Textset) a Personal TEX.

To, co následuje, je asi neúplný seznam. Zahrnutím dodavatele do tohoto seznamu přirozeně není garantována jeho užitečnost. Pro možnost ověření je poskytnut i zdroj informace.

Obecně jsou komerční implementace „úplné“, tzn. s vhodnými prohlížeči a ovladači tiskáren. Většinou též obsahují obsáhlou dokumentaci (tj. nejenom TEXbook!) a některé druhy služeb podpory. V některých případech to znamená bezplatné telefonní číslo (použitelné asi jen v USA nebo Kanadě), ale jinak ostatní mají e-mail a normální telefon a fax.

Unix; TEX Silicon Graphics Iris/Indigo, Solaris 2.1, IBM RS/6000, DEC/RISC-Ultrix, HP 9000. „Všechny TEXovské balíky. Připraveno k použití, úplná dokumentace a podpora.“

ArborText Inc
1000 Victors Way
Suite 400
Ann Arbor MI 48108
USA

Tel: (+1) 313-996-3566
Fax: (+1) 313-996-3573

Zdroj: *TUGboat*, **15**(1), 1994

VAX/VMS; Convergent TEX Úplný systém pro počítače VAX/VMS (připravuje se verze pro procesor Alpha). Zahrnuje LATEX, mezinárodní podporu, METAFONT a Web.

Northlake Software, Inc.
812 SW Washington, Ste 1100
Portland, OR 97201
USA

Tel: (+1) 503-228-3383
Fax: (+1) 503-228-5662
E-mail: `rau@nls.com`

Zdroj: E-mail, odesilatel Pat Rau, listopad 1994

PC; TEX „Bitmap free TEX pro Windows.“

Y&Y, Inc.
45 Walden Street

Concord MA 01742
USA

Tel: (+1) 508-371-3286
Fax: (+1) 508-371-2004
E-mail: `sales-help@YandY.com`, `tech-help@YandY.com`
Web: `http://www.YandY.com/`

Zdroj: zpráva Y&Y, únor 1995

pcTEX Dlouze vžitý: nyní má implementaci pro Windows.

Personal TEX Inc
12 Madrona Street
Mill Valley, CA 94941
USA

Fax: (+1) 415-388-8865
E-mail: `pti@crl.com`
Web: `http://www.pctex.com/`

Zdroj: TUGboat, **16**(2), 1995

PC; VTEX Rovněž „Bitmap-free“.

MicroPress Inc
68-30 Harrow Street
Forest Hills, NY 11375
USA

Tel: (+1) 718-575-1816
Fax: (+1) 718-575-8038
E-mail: `sales@micropress-inc.com`
Web: `http://www.micropress-inc.com/`

Zdroj: stránka MicroPress, duben 1996

PC; microTEX MicroTEX a TEX tools.

Micro Programs, Inc.
251 Jackson Ave.
Syosset, NY 11791
USA

Tel: (+1) 516-921-1351
E-mail: `sales@microprograms.com`

Zdroj: AMS listing, listopad 1994

PC; Scientific Word Scientific Word a Scientific Workplace nabízejí mechanismus pro skoro WYSIWYG vstup LATEXovských dokumentů. Dodávají se s balíkem TrueTEX od firmy Kinch (viz výše). Objednávky v rámci UK mají být adresovány na Scientific Word Ltd., v dalších zemích přímo na dodavatele: TCI.

Dr Christopher Mabb
Scientific Word Ltd.
98 Pont Adam
Ruabon
Wrexham
Clwyd, LL14 6EF
UK

Tel: 01978 823088 (within the UK)
Fax: 01978 823066 (within the UK)
E-mail: `christopher@sciword.demon.co.uk`

TCI Software Research Inc.
1190 Foster Road
Las Cruces
New Mexico, 88001
USA

Tel: (+1) 505-522-4600
Fax: (+1) 505-522-0116
E-mail: `info@tcisoft.com`
Web: `http://www.tcisoft.com/tcisoft.html`

Zdroj: Dopis, odesilatel Christopher Mabb, listopad 1994

Macintosh; Textures „Systém TEX pro zbytek světa". Též poskytuje implementaci METAFONTu a několik utilit pro manipulaci s fonty.

Blue Sky Research
534 SW Third Avenue
Portland, Oregon 97204
USA

Tel: 800-622-8398 (within the USA)
Tel: (+1) 503-222-9571
Fax: (+1) 503-222-1643
E-mail: `sales@bluesky.com`
Web: `http://www.bluesky.com/`

Zdroj: TUGboat, **15**(1), 1994

AmigaTEX Úplná implementace pro počítač Commodore Amiga. Zahrnuje úplnou podporu pro obrazovku i tiskárnu pro PostScript, grafiku a fonty, rastrovou grafiku IFF, automatické generování fontů a všechna standardní makra a utility.

Radical Eye Software
PO Box 2081
Stanford, CA 94309
USA

Zdroj: Dopis, odesilatel Tom Rokicki, listopad 1994

G. DVI ovladače a prohlížeče

40. Konverzní programy z DVI do PostScriptu

Nejlepší volně šiřitelný konverzní program z `dvi` do PostScriptu, který běží pod mnoha operačními systémy, je *dvips* Toma Rokickiho. *dvips* je napsán v C a snadno se přenáší do jiných operačních systémů; je dostupný v `dviware/dvips`

Verze pro VMS jsou dostupné přes knihovnu DECUS (viz Otázku **??**) a také z CTAN: `systems/vms/Alpha/dvips555_axp_exe.zip` (pro stroje s procesory Alpha), `systems/vms/VAX/dvips555_vax_exe.zip` (pro počítače VAX); podpůrné soubory jsou k dispozici v CTANu v souboru `systems/vms/dvips555_support.zip` a sada fontů pro použití s LaTeXem 2_ε je dostupná v `systems/vms/dvips_fontsupport.zip`

Přeloženou verzi tohoto ovladače pro MS-DOS lze najít na CTANu v souboru `systems/msdos/dviware/dvips`

Verze *dvips* od Karla Berryho (nazvaná *dvipsk*) má konfigurační popis a kód pro prohledávání cesty podobné jako jeho další programy (např. *web2c*); je dostupná z `dviware/dvipsk`

Další dobrý přenositelný program je *dvitops* od Jamese Clarka, který je také napsán v C, lze přeložit pro Unix, MS-DOS, VMS a Primos, nepodporuje ale virtuální fonty. Je možné jej získat v `dviware/dvitops`

Uživatelé Macintoshů mohou použít znamenité ovladače zabudované do OzTEXu nebo Textures, či verzi *dvips* v balíku CMacTEX.

41. DVI ovladače pro HP LaserJet

Balík emTEX (viz Otázku **??**) obsahuje ovladač *dvihplj* pro LaserJet.

Verze 2.10 Beebeho ovladačů podporuje LaserJet. Tyto ovladače lze přeložit pod Unixem, VMS, na Atari ST s DEC-20 a jsou dostupné z `dviware/beebe`

Program Karla Berryho *dviljk*, který má stejný kód pro prohledávání cesty jako jeho *dvipsk* (viz Otázku **??**), lze získat v `dviware/dviljk`

42. DVI prohlížeče

EmTEX a gTEX pro PC a OzTEX pro Macintosh přicházejí s prohlížeči, jež mohou být použity na těchto platformách. Existuje dobrý OS/2 Presentation Manager prohlížeč v emTEXu a volně dostupný prohlížeč pro Windows (`dviware/dviwin`). Komerční balíky TEXu pro PC (viz Otázku ??) mají dobré DOS a Windows prohlížeče.

Dostupné prohlížeče pro ostatní operační systémy:

xdvi Nejvíce používaný prohlížeč pro X Window System (a tedy pro téměř každý Unix a moderní VMS pracovní stanici); lze jej získat na `dviware/xdvi`
Verze *xdvi* Karla Berryho nazvaná *xdvik* má podobné rysy jako jeho *dvipsk* (viz Otázku ??; je dostupná v `dviware/xdvik`

dvipage Pro SunView na (poněkud starších) pracovních stanicích Sun. Byl publikován v čísle 15 `comp.sources.unix` a je archivován na `dviware/dvipage`

xtex Starší prohlížeč pro X Windows System; k nalezení na CTANu v `dviware/seetex`

dviapollo Pro pracovní stanici Apollo Domain; dostupný na CTANu v `dviware/dviapollo`

dvidis Pro (poněkud starší VMS) VAX stanice běžící s VWS; dostupný v `dviware/dvidis`

dvitovdu Pro Tektronix 4010 kompatibilní a další terminály pod Unixem a VMS; dostupný v `dviware/dvitovdu`

dvi2tty Konverzní program z `dvi` do ASCII, pro běžné terminály; dostupný v `dviware/dvi2tty`

texsgi Pro SGI pod Irixem. Dostupná je jak binární verze tak zdrojový text, nezapomeňte na fonty. Dostupný v `dviware/texsgi`

H. Podpůrné balíky pro TEX

43. Fig, TEXu přátelský kreslicí balík

Fig (*XFig*) je nástroj ovládaný pomocí menu, který umožňuje kreslit objekty na obrazovce X Windows pracovní stanice. *transfig* je sada nástrojů převádějících kód, jenž produkuje *fig*, do dalších grafických jazyků včetně PostScriptu a LATEXového prostředí picture. Jsou dostupné v `graphics/xfig` a `graphics/transfig`

Fig ošetřuje Micah Beck (`beck@cs.cornell.edu`) a *transfig* je spravován Brianem Smithem (`bvsmith@bl.gov`). Dalším prostředkem pro konverzi z *fig* je *fig2mf*, který generuje METAFONT kód ze vstupu, jenž dává *fig*. Lze jej získat v `graphics/fig2mf`

44. TEXCAD, kreslicí balík pro LATEX

TEXCAD je program pro PC, který umožňuje uživateli vytvářet kresby na obrazovce myší nebo kurzorovými klávesami za pomoci obrazovkové nabídky dostupných základních obrázků. Výstupem je kód pro LATEXovské prostředí picture. Volitelně se dá nastavit, aby šlo vkládat čáry pod libovolnými úhly za použití rodiny ovladačů emTEXu (`\specials` — viz Otázku ??). TEXCAD je součástí distribuce emTEXu.

Verze pro Linux, `systems/unix/linux/xtexcad-2.1.tar.z`, by měla fungovat i na jiných operačních systémech Unix.

45. Korektory překlepů pro práci s TEXem

Pro Unix je pravděpodobně nejlepší program *ispell*. Je dostupný v `support/ispell`. Pozor na verzi s číslem `4.x` — tahle verze reprezentuje odchýlenou verzi, jež postrádá mnoho užitečných rysů série 3.x.

Pro MS-DOS existuje několik programů. *amspell* může být vyvolán z editoru (dostupný jako `support/amspell`). *jspell* je rozšířená verze programu *ispell* (lze získat v `support/jspell`).

Pro Macintosh je možností program *Excalibur*. Běží v módu native na obou druzích Macu a je dostupný na CTANu v adresáři `systems/mac/support/excalibur/Excalibur-20-sea.hqx` (v tomtéž adresáři jsou další slovníky).

Korektor překlepů pro VMS lze najít v `support/vmspell`

Korektor pro češtinu je pro členy CSTUGu součástí distribuce CSTeXu. České korektory bývají také standardní složkou textových editorů komerčně distribuovaných v České republice.

46. Balík VORTeX

VORTeX (dostupný v `support/vortex`) je balík programů napsaný na Kalifornské Univerzitě v Berkley a byl popsán Michalem A. Harrisonem v článku „*News from the VORTeX project*“ v TUGboat 10(1), s. 11–14, 1989. Zahrnuje několik pěkných prohlížečů a několik *emacs* módů pro TeX a BibTeX. Distribuce VORTeXu není spravována a v současnosti je zastaralá (nebyl udělán upgrade pro TeX, verzi 3).

I. Dokumentované programování

47. Co je dokumentované programování

Dokumentované programování je kombinace dokumentace a zdrojového textu do společné formy vhodné ke čtení lidmi. Stručně řečeno, dokumentované programy sdružují zdrojový text a dokumentaci do jednoho souboru. Nástroje dokumentovaného programování pak zpracováním tohoto souboru vytvoří buď čitelnou dokumentaci nebo kompilovatelný kód. Styl dokumentovaného programování WEB vytvořil D. E. Knuth při vývoji svého sázecího systému TeX.

Diskuse o dokumentovaném programování je soustředěna do usenetové skupiny `comp.programming.literate`, která je přístupná i jako elektronická konference na adrese `litprog@shsu.edu` (viz Otázku ??). FAQy o dokumentovaném programování jsou uloženy v `help/LitProg-FAQ`

48. WEB pro C, FORTRAN a další jazyky

TeX je napsán v programovacím jazyku WEB. Jedná se o prostředek, který implementuje koncept „dokumentovaného programování“.

CWEB, WEB pro programy v C, který vytvořil Silvio Levy, je dostupný jako `web/c_cpp/cweb`

Spidery WEB podporuje mnoho jazyků včetně Ada, awk a C. Byl napsán Normanem Ramseyem a ačkoli není volně šiřitelný, lze jej používat bezplatně. Dá se získat v `web/spiderweb`

FWEB je verze pro Fortran, Ratfor a C napsaná Johnem Krommesem. Je dostupná v `web/fweb`

SchemeWEB je Unixový filtr, který překládá SchemeWEB do zdrojového kódu LaTeXu nebo Scheme. Napsal ho John Ramsdell a je dostupný v `web/schemeweb`

APLWEB je verze WEBu pro APL a dá se získat v `web/apl/aplweb`

FunnelWeb je verze WEBu, která je jazykově nezávislá. Je dostupná v `web/funnelweb`

Další jazykově nezávislé verze WEBu jsou *nuweb* (napsaná v ANSI C), dostupná v `web/nuweb`, a *noweb*, dostupná na `web/noweb`

J. Formátové konverze

49. Konverze mezi TeXem/LaTeXem a ostatními

troff *troff-to-latex* (k dispozici jako `support/troff-to-latex`), jenž napsal Kamal Al-Yahya na Standfordské universitě (Kalifornie, USA), napomáhá při překladu *troff* dokumentu do formátu LaTeXu. Rozlišuje většinu `-ms` a `-man` maker, k tomu většinu *eqn* a některé *tbl* příkazy preprocesoru. Složitější věci je potřeba dodělat ručně. Jsou podporovány dva styly souborů. Existuje také manuálová stránka (která se velmi dobře konvertuje do LaTeXu). Program má copyright, ale je zdarma. Rozšířená verze tohoto programu, *tr2latex*, je dostupná v `support/tr2latex`

Distribuce DECUS TeX (viz Otázku ??) také obsahuje program konvertující *troff* do TeXu.

Scribe Mark James (`jamesm@dialogic.com`) vlastní kopii programu *scribe2latex*, kterou nemohl vyzkoušet, ale kterou přenechá každému, kdo projeví zájem. Program napsal Van Jacobson z Lawrence Berkeley Laboratory.

WordPerfect *wp2latex* (lze získat v `support/wp2latex`) je program pro PC napsaný v Turbo Pascalu R. C. Houtepenem na univerzitě v Eindhovenu v Holandsku. Konvertuje dokumenty z formátu *WordPerfect*u 5.0 do LaTeXu. Zdrojové texty v Pascalu jsou také k dispozici. Uživatel jej přes některá omezení shledá „celkem ucházejícím". Je vysoce hodnoceno jeho rozlišování změn fontů, ale neumí dělat indexy, obsah, okraje a grafiku a nezvládá nové prvky *WordPerfect*u verze 5.1, speciálně formátování rovnic. Na program je copyright, ale je zdarma.

Glenn Geers z univerzity v Sydney, Austrálie (`glenn@qed.physics.su.oz.au`) přepisuje *wp2latex* do jazyka C a přidává některé rysy *WordPerfectu* 5.1, zejména jeho práci s rovnicemi. Program je v podadresáři `glenn` na `support/wp2latex`

PC-Write `pcwritex.arc` je k dispozici v `support/pcwritex`

Jedná se o ovladač tisku pro PC-Write, který „tiskne" dokument formátu PC-Write V2.71 do souboru kompatibilního s TeXem. Napsal ho Peter Flynn z University College, Cork, Irská republika.

runoff Konverzní program Petera Vanroose (jeho adresa je `vanroose@esat.kuleuven.ac.be`) je napsán ve VMS Pascalu. Zdrojové texty a běhuschopný kód pro VAX jsou dostupné v `support/rnototex`

refer/tib Existuje několik programů pro konverzi bibliografických dat mezi BibTeXem a formátem *refer/tib*. Lze je nalézt v adresáři `biblio/bibtex/utils/refer-tools`

Ačkoli to jeho název nenapovídá, adresář také obsahuje dávku pro převod BibTeXu do *refer*. Kolekce není udržována.

RTF Program pro konverzi z Microsoft Rich Text Format do TeXu je k mání na `support/rtf2tex`. Napsal a spravuje jej Robert Lupton (`rhl@astro.princeton.edu`). Existuje také konvertor do LaTeXu od Erwina Wechtla v `support/rtf2latex`

Microsoft Word Základní program pro převod z MS-Wordu do LaTeXu je *wd2latex* pro MS-DOS (v `dviware/wd2latex`). Lepší nápad ale je konvertovat dokument do formátu RTF a použít některý z RTF konvertorů zmiňovaných výše.

Skupina na Ohio State University (USA) pracuje na společném dokumentovém formátu založeném na SGML s cílem, aby se libovolné formáty daly vzájemně převádět. „Vstupní filtry" poskytuje *FrameMaker* s cílem překladu z cizího formátu (pravděpodobně včetně TeXu) do vlastního.

Další informace viz `http://www.kfa-juelich.de/isr/1/texconv/texcnv.html`

50. Konverze z TeXu/LaTeXu do holého ASCII

Zde je snahou emulovat unixový *nroff*, který formátuje text na obrazovce, jak jen to jde nejlépe, a to ze stejného vstupu jako unixový sázecí program *troff*.

Ralph Droms (`droms@bucknell.edu`) má stylový soubor a program, který způsobuje, že je LaTeX rovnocenný s *nroff*, ale neporadí si dobře s tabulkami a s matematikou. Software je dostupný v `support/txt`. Původní *dvi2tty*, který často poskytuje přijatelný výsledek, lze získat v `dviware/dvi2tty`

Jinou možností je použít *screen.sty* (k nalezení na CTANu v adresáři `macros/latex209/contrib/misc/screen.sty`). Použijte některý z programů typu *dvi2tty*, můžete také zkusit `dviware/crudetype`. Dále je možné použít konverzní program z LaTeXu do ASCII, *l2a* (`support/l2a`), ačkoliv je to skutečně hodně „odTeXovávací" program.

Základní odTeXovávací program je *detex* (`support/detex`), který před zápisem na výstup smaže všechny komentáře a řídicí sekvence ze vstupu. Původním záměrem bylo připravit vstup pro jednoduchý korektor překlepů.

51. Převod z HTML či SGML do TeXu

Nathan Torkington (`Nathan.Torkington@vuw.ac.nz`) vyvinul z HTML ovladače v balíku Xmosaic od NCSA *html2latex*. Program načítá HTML soubor a generuje z něj soubor LaTeXový. Konverzní kód je předmětem omezení NCSA, ale kompletní zdrojový text lze získat na `support/html2latex`

Peter Flynn vyvinul *pcl* program *sgml2tex* (pro PC). Je ovládán konfiguračním souborem a je přiložen příklad konfigurace pro překlad HTML souboru. Program je k dispozici v support/sgml2tex

52. Konverze do HTML

Převod z TeXu do HTML není triviální. TeX není značkovací, ale sázecí systém, takže se koncepčně od HTML velmi liší. S LaTeXem máte víc naděje, ale ani zde není převod zadarmo. Je dobré poznamenat že a) chcete-li skutečně kvalitní Webový dokument, je lepší přepsat ho znovu, a b) HTML (i HTML3) má velmi chudé sázecí schopnosti a cokoliv netriviálního transformuje do grafiky.

LaTeX2HTML (support/latex2html) je balík (většinou *perl* dávky), který rozkládá LaTeXovský dokument na jednu či více komponent, které pak spojuje dohromady, takže se dají číst přes World-Wide Web jako hypertextový dokument. Definuje zobrazení mezi LaTeXovskými vnitřními odkazy a hyperodkazy a rozšiřuje referenční mechanismus LaTeXu, aby se zpřístupnily odkazy na jiné připojitelné dokumenty a další prostředky Internetu. Překládá akcentované a další znaky LaTeXu (jak jen to jde nejlépe) do formy, kterou dokáže zobrazit World-Wide Web prohlížeč, a zprostředkovává mechanismus, kterým se matematika a další věci, jež prohlížeč nezvládá, dají převést do obrázku, který může být začleněn do hypertextového dokumentu.

LaTeX2HTML vyžaduje *Perl*, PBM utility, *dvips*, *GhostScript* a jiné programy. Běží pod Unixem. V *TUGboatu* **16**(2) je publikován podrobný popis programu LaTeX2HTML a návod na jeho konfiguraci.

53. Vytváranie hypertextových dokumentov v TeXu

Ak chcete vytvoriť hypertextový dokument pomocou LaTeXu napríklad pre World-Wide Web, uvážte štyri technológie (prekrývajúce sa).

1. Skúste priamu konverziu z LaTeXu do HTML (pozri Otázku **??**);
2. Prepíšte svoj dokument použijúc Texinfo (pozri Otázku **??**) a skonvertujte ho do HTML;
3. Pozrite sa na Adobe Acrobat, elektronický systém na uchovávanie a prezeranie dokumentov. Tento systém môže zachovať úplne sadzbu Vášho dokumentu. (Pozri Otázku **??**.)
4. Využite hyperTeX konvencie (štandardizované príkazy \special). Existujú makro-balíky, ktoré ich podporujú pre plain TeX a LaTeX.

Projekt HyperTeX má za cieľ rozšíriť funkčnosť všetkých odkazových príkazov pre LaTeX (vrátane obsahu) vytváraním príkazov \special, ktoré sú sledované DVI procesormi a vytvárajú hypertextovú štruktúru. Poskytujú všeobecné hypertextové spojenia, vrátene napojenia na vonkajšie dokumenty.

HyperTeXovská špecifikácia určuje, že príslušný prezerač/prekladač musí rozpoznávať nasledujúce príkazy \special:

href: html:
name: html:
end: html:
image: html:<img scr = "href_string">
base_name: html:<base href = "href_string">

Príkazy *href*, *name* a *end* sa používajú na definovanie základných hypertextových operácií spojenia jednotlivých častí dokumentu.

Ďalšie detaily sú dostupné na http://xxx.lanl.gov/hypertext/. Existujú zatiaľ len dve všeobecne používané implementácie špecifikácií: modifikované *xdvi* a modifikované *dvips*. Výstup z modifikovaného *dvips* môže byť použitý modifikovaným programom *GhostScript* alebo programom Acrobat Distiller.

54. Vytváranie akrobatovských (PDF) dokumentov v TeXu

Existujú dve možnosti: (a) použiť program *pdftex* Han The Thanha pre priamy prevod z TeXu do PDF, alebo (b) previesť PostScript do PDF programom *Distiller*.

V prvom prípade preložíte zdrojový súbor programom *pdftex*. *pdftex* spracováva dokument rovnakým spôsobom ako TEX, ale výstup je do PDF. Informácie o tomto programe sú na `http://www.cstug.cz/pdftex`

V druhom prípade použijete program *dvips*, t.j. program konvertujúci DVI súbor do PS, teda PostScriptového súboru. Výsledok necháte spracovať programom Adobe Acrobat *Distiller* alebo, ešte jednoduchšie, ak používate počítač Mac alebo Windows TEX, môžete inštalovať Acrobat Exchange. Vtedy môžete nadefinovať „vytvárač" PDF súboru ako ďalšiu tlačiareň. V tomto prípade ale nemôžete nadefinovať ďalšie hypertextové spojenia a teda to nie je dobrá cesta, hodí sa iba na jednoduché dokumenty. Preto je treba použiť Adobe Distiller, ktorý podporuje špeciálne postscriptovské operátory, tzv. `pdfmark`, pre prechod informácií do PDF súboru.

Aby sme preložili všetky krížové odkazy v LATEXu do Acrobatovských spojení, potrebujeme LATEXovský balík, ktorý vhodne predefinuje vnútorné príkazy LATEXu. Existujú dva takéto balíky pre LATEX 2_ε, obidva založené na HyperTEXovských špecifikáciách (pozri Otázku ??). Prvý je *hyperref* od Sebastiana Rahtza (dostupný z `macros/latex/contrib/supported/hyperref` a druhý je *hyper* od Michaela Mehlicha (`macros/latex/contrib/supported/hyper`). Potom použijete *dvips* (modifikovaný *dvips*) a preložíte DVI súbor do PostScriptu, ktorý je akceptovateľný pre program Distiller. Alternatívne, keď chcete použiť iba Acrobat, *hyperref* má v sebe tzv. „native PDF mode", ktorý pracuje so štandardným *dvips* (a väčšinou iných prekladačov) a dáva všetky možnosti využitia `pdfmark`.

Bohužiaľ Distiller nie je voľne dostupný program. Pokiaľ nemáte Acobat Reader, tak *GhostScript* (verzia 3.51 a neskoršie) môže zobrazovať PDF súbory. Čoskoro sa dočkáme aj prekladača z DVI do PDF formátu.

Inou možnosťou je použiť

K. METAFONT

55. Jak dostat z METAFONTu to, co chcete

METAFONT vám umožňuje vytvořit si vlastní font, ale většina uživatelů TEXu ho nikdy nepoužívá. METAFONT, na rozdíl od TEXu, vyžaduje nekterá speciální nastavení. Každé výstupní zařízení, pro které se generuje font, potřebuje přidružený mód. Módy se definují použitím konvencí `mode_def` popsaných na straně 94 v *The METAFONTbook* (viz Otázku ??). Je potřeba mít soubor zpravidla nazvaný `local.mf` obsahující `mode_defs`, které budete používat. Jestliže `local.mf` ještě neexistuje, dobrým výchozím bodem může být kolekce módů od Karla Berryho, dostupná jako `fonts/modes/modes.mf`. Lze ji použít jako `local.mf` beze změn pro „dostatečně širokou" implementaci METAFONTu. Výpis nastavení pro nejrůznější výstupní zařízení bývá také pravidelně zveřejňován v TUGboatu (viz Otázku ??). Nyní vytvořte bázový soubor `plain` s použitím *inimf*, `plain.mf` a `local.mf`:

```
% inimf
This is METAFONT...
 **plain                 zadejte „plain"
 (output)
 *input local            zadejte toto
 (output)
 *dump                   zadejte toto
 Beginning to dump on file plain...
 (output)
```

Tím se vytvoří bázový soubor pojmenovaný `plain.base` (případně něco podobného; například v MS-DOSu to bude `PLAIN.BAS`), který je třeba přemístit do adresáře, jenž ve vašem systému obsahuje bázové soubory. (Některé systémy obsahují i dva nebo více takových adresářů, každý pro jinou „velikost" použitého METAFONTu.)

Nyní se potřebujete ujistit, že METAFONT při spuštění čte tuto novou bázi. Jestliže METAFONT načítá na vašem systému implicitně `plain` bázi, jste připraveni. Pod Unixem (za použití běžné distribuce *web2c*) to tak skutečně funguje, ale mohli bychom třeba definovat příkaz *mf*, který vykonává virmf &plain načtením plain bázového souboru.

Obvyklý způsob, jak vytvořit font s plain METAFONTem, je začít řádkem

```
\mode=<jméno módu>; mag=<zvětšení>; input <jméno fontového souboru>
```

jako odpověď na výzvu „**“ nebo na příkazové řádce METAFONTu. (Pokud <jméno módu> je neznámé nebo neplatné, bude použit implicitní mód a METAFONT vyrobí výstupní soubor nazvaný < jméno fontového souboru>.2602gf) <Zvětšení> je číslo v plovoucí řádové čárce nebo „magstep“ (magstepy jsou definovány v *The METAFONTbook* a v *The TeXbook*). Jestliže mag=<zvětšení> je neplatné, bude implicitně 1 (magstep 0). Například, pro generování cmr10 zvětšeného na 12pt pro tiskárnu epson byste měli napsat

```
mf \mode=epson; mag=magstep 1; input cmr10
```

Poznamenejme, že pod Unixem jsou znaky \ a ; obvykle v uvozovkách nebo se zpětným lomítkem, takže by to zpravidla mělo vypadat takhle:

```
mf '\mode=epson; mag=magstep 1; input cmr10'
```

Pokud nemáte *inimf* nebo potřebujete speciální mód, který není v bázi, můžete vložit jeho příkazy do souboru (např. ln03.mf) a vyvolat ho za běhu příkazem \smode. Tak například, aby se vytvořil cmr10.300gf pro tiskárnu LN03 za použití souboru

```
% This is ln03.mf as of 2/27/90
% mode_def courtesy of John Sauter
proofing:=0;
fontmaking:=1;
tracingtitles:=0;
pixels_per_inch:=300;
blacker:=0.65;
fillin:=-0.1;
o_correction:=.5;
```

(všimněte si nepřítomnosti příkazů mode_def a enddef), je třeba napsat

```
mf \smode="ln03"; input cmr10
```

Tato technika není jediná, kterou byste správně měli používat, ale může se ukázat užitečná, jestliže máte novou tiskárnu a chcete experimentovat s parametry, či z nějakých jiných důvodů editujete používané parametry. Jakmile jste si jednou ujasnili přípustnou množinu parametrů, můžete je uplatnit při přebudování bázového souboru, který používáte.

Souhrn výše uvedeného od Geoffrey Tobina a upozornění na nejčastější nástrahy a úskalí při používání METAFONTu lze nalézt v dokumentu info/metafont-for-beginners.tex

56. Které fontové soubory je třeba si nechat

METAFONT produkuje při svém běhu tři soubory: metrikové (tfm) soubory, soubory s generickými fonty (gf) a soubory s popisem překladu (log). Všechny tyto soubory mají stejné bázové jméno jako vstup, (např. pokud vstupní soubor byl cmr10.mf, výstupem bude cmr10.tfm, cmr10.nnngf[8] a cmr10.log).

Pro práci TeXu s fontem je zapotřebí tfm soubor, takže si ho musíte nechat. Ale obyčejně chcete generovat tentýž font ve více velikostech a pokaždé se bude (automaticky) generovat další tfm soubor. Tyto soubory jsou stejné, proto stačí nechat si jediný z nich.

K výrobě obrazovkového nebo tiskového výstupu potřebuje dvi procesor rastrový soubor fontu; to je to, co poskytuje gf soubor. Avšak, zatímco dříve používané dvi procesory uměly pracovat s gf soubory, moderní procesory používají komprimované rastrové (pk) soubory. Proto je třeba generovat pk soubor z gf souboru. Tohle za vás udělá program *gftopk* a jakmile je to jednou hotovo, můžete soubor gf odstranit.

Pokud nenastanou nějaké problémy, neměl by být soubor log nikdy použit. Normálně tedy není třeba nechávat si jej.

[8]Všimněte si, že jméno souboru může být přetvořeno u takových operačních systémů jako MS-DOS, které nepřipouštějí dlouhé názvy souborů.

57. Vytažení bitových map z archívu

Většina lidí, kteří začínají používat TEX s laserovou tiskárnou s rozlišením 300 dpi (dots-per-inch) a fonty Computer Modern pro toto rozlišení, je zásobena množstvím TEXovských balíků. Na CTAN jsou také dvě takové sady: `fonts/cm/pk/pk300.zip` (pro černý tisk) a `fonts/cm/pk/pk300w.zip` (pro bílý tisk). Ale někteří uživatelé chtějí poslat svou práci na vysoce kvalitní sázecí stroje (obyčejně s rozlišením 1270 dpi nebo více). Též se častěji používají 600 dpi laserové tiskárny. Proč archívy a knihovny neposkytují bitové mapy fontů v těchto velikostech? Důvody jsou dva:

1. Pokud je bitová mapa fontu vytvořena pomocí METAFONTu, je třeba znát charakteristiky zařízení; kdo může vědět, jaký 600 či 1270 dpi přístroj vlastníte? (Tahle námitka se samozřejmě stejně dobře týká 300 dpi tiskáren.)
2. Při vysokých rozlišeních jsou bitové mapy *objemné*. Kdo ví, které fonty a s jakou velikostí potřebujete?

Bylo by možné poskytovat v archívech nějakou sadu 1270 dpi bitových map fontů, ale zabralo by to spoustu místa a možná by to pro vás nebylo to pravé.

Tak co s tím? Můžete si sami vytvořit potřebné fonty pomocí METAFONTu; není to tak složité a některé ovladače vám pomohou (*dvips* a ovladače emTEXu) zkonstruovat příkazy METAFONTu. Můžete se podívat na CTAN do kolekce módů METAFONTu Karla Berryho, do souboru `fonts/modes/modes.mf`. Na druhé straně, pokud vlastníte PostScriptové zařízení, uvažujte o fontech ve formátu Type 1. Je možné koupit všechny fonty Computer Modern v PostScriptovém tvaru od Blue Sky Research nebo od Y&Y (adresa je uvedena v odpovědi na Otázku ??), či použít veřejně přístupnou verzi Basila Malysheva v `fonts/cm/ps-type1` (kolekce Paradissa je kompletní, ale postupně je nahrazována lepší kolekcí BaKoMa).

L. PostScript a TEX

58. Použití PostScriptových fontů v TEXu

Ještě než začnete: aby mohl použít PostScriptové fonty, TEX potřebuje soubory s *metrikami* (zvané `tfm`). Několik sad metrik je dostupných v archívech; mechanismus generování nových je popsán v Otázce ??. Také potřebujete vlastní fonty. PostScriptové tiskárny se sice dodávají již se sadou fontů zabudovaných, ale pro rozšíření svého repertoáru si většinou stejně musíte koupit jednu z komerčních nabídek fontů (viz Otázku ??).

Pokud používáte LATEX 2_ε, nejlepším způsobem, jak dostat do svého dokumentu PostScriptové fonty, je použít balík PSNFSS, který spravují Sebastian Rahtz a Alan Jeffrey (`macros/latex/packages/psnfss`). Je podporován projekčním týmem pro LATEX3, takže zprávy o chybách by mohly a měly být brány do úvahy. PSNFSS poskytuje sadu balíků pro změnu standardních roman, sans-serif a typewriter fontů. Např. `times.sty` nastaví fonty Times Roman, Helvetica a Courier místo Computer Modern, zatímco `avant.sty` změní rodinu fontů sans-serif na AvantGarde. Pro práci s těmito balíky jsou potřebné soubory s metrikou (pozor na problémy s kódováním! – viz Otázku ??) a soubor s popisem fontu (`.fd`) pro každou rodinu, kterou chcete použít. Výhodné je, že metriky pro společných 35 PostScriptových fontů pro většinu tiskáren jsou začleněny do PSNFSS, zkomprimovány jako `macros/latex/packages/psnfss/lw35nfss.zip`

Pro starší verze LATEXu existují nejrůznější schémata, z nichž nejjednodušší pro použití jsou pravděpodobně makra PSLATEX distribuovaná s *dvips*.

Pro `plain`TEX můžete použít fonty, jaké chcete. Pokud kódování fontů není stejné jako u Computer Modern, je na vás, abyste si předefinovali různá makra a akcenty, anebo můžete použít mechanismus pro překódování fontů dostupný v mnoha ovladačích a v *ps2pk* a *afm2tfm*.

Rafinovaný balík Lollipop Victora Eijkhouta (`macros/lollipop`) podporuje deklaraci rodin fontů a styly podobným způsobem jako LATEXovský NFSS, takže se dá snadno použít s PostScriptovými fonty.

Některé společné problémy, které byly zmíněny, jsou diskutovány i jinde (viz Otázku ??).

59. Prohlížení souborů s PostScriptovými fonty

Většina TEXovských prohlížečů zobrazuje pouze bitové mapy `pk` fontů. Pokud chcete prohlížet dokumenty s PostScriptovými fonty, máte tři možnosti:

1. Převést `.dvi` soubor do PostScriptu a použít PostScriptový prohlížeč. Některé moderní Unixové X implementace mají tohle zabudováno (jako třeba NeXT-step). Uživatelé Unixu (X11), Windows, OS/2 a MS-DOSu mohou použít volně dostupný GhostScript (`support/ghostscript`), implementaci kompletního PostScriptu úrovně 2.
2. Pod Windows na PC, nebo na Macintoshi umí Adobe Type Manager zobrazit PostScriptové fonty. TEXtures (Macintosh) pracuje podobně, pod Windows pak lze použít program *dviwindo* od Y&Y pro prohlížení bitových map. (Detaily těchto doplňků: viz Otázku ??.)
3. Jestliže máte PostScriptové fonty formátu Type 1, použijte *ps2pk* (`fonts/utilities/ps2pk`) nebo *gsftopk* (navržený pro použití s fonty programu *ghostscript*, `fonts/utilities/gsftopk`) pro výrobu `pk` bitových map fontů, kterým váš prohlížeč bude rozumět. Tak lze dosáhnout skvělých výsledků, které jsou vhodné i pro tisk na zařízeních bez PostScriptu. Prověřte si právní podmínky, pokud chcete tyto fonty získat. Nejznámější PostScriptové fonty jako Times a Courier se dodávají s formátem Type 1 na disku s Adobe Type Managerem (často dohromady s Windows a část s OS/2).

60. Soubory metrik TEXovských fontů pro PostScriptové fonty

Výrobci fontů takoví jako Adobe opatřují metrikovým souborem každý font, a to v `afm` (Adobe Font Metric) formě. Tento formát lze konvertovat do tvaru `tfm` (TEX Font Metric). CTAN archivy obsahují předem připravené metriky, které budou pro mnoho lidí více než dostatečné, ale můžete si udělat konverzi i sami, pokud máte speciální potřebu. Zajímavou otázkou je *kódování* fontů. Zatímco všichni více či méně souhlasíme s pozicemi 96 znaků ve fontech (základní ASCII sada), zbytek z (běžných) 256 se různí. Nejobvyklejší problémy jsou s plovoucími akcenty a se speciálními znaky, jako je třeba označení libry šterlinků. Existují tři způsoby, co s tím: buď změníte makra TEXu, která odkazují na znaky (nic moc a náchylné na chyby), nebo změníte kódování fontu (jednodušší, než si myslíte), anebo použijete virtuální fonty, (viz Otázku ??) abyste TEXu předstírali, že kódování je totéž, jaké používá on. Používání LATEXu dovoluje změnu kódování v TEXu. Podrobnosti si lze přečíst v *LATEX Companion* (viz Otázku ??). V praxi, pokud často píšete jinak než anglicky (ale latinkou), se silně doporučuje použití `fontenc` balíku s volbou „`T1`“ pro výběr T1 (také známé jako „Cork“ nebo „DC“) kódování. PSNFSS balík (viz Otázku ??) podporuje PostScript (s virtuálními fonty) v tomto kódování.

Balík *fontinst* Alana Jeffreye (`fonts/utilities/fontinst`) je konvertor z `afm` do `tfm` napsaný v TEXu. Používá se pro generování souborů s použitím LATEXovského PSNFSS balíku k podpoře užívání PostScriptových fontů. Jedná se o komplikovaný balík, nic pro slabé nátury, ale je dost silný pro uspokojení většiny potřeb. Mnoho z jeho síly se vztahuje na použití virtuálních fontů (viz Otázku ??).

Pro poněkud jednodušší problémy je použitelný rychlý a efektivní Rokickiho *afm2tfm*, distribuovaný s *dvips* (`dviware/dvips`). Poznamenejme, že metriky a styly, které se dodávají s *dvips*, *nejsou* kompatibilní se současným LATEXem.

Pro MacIntosh existuje program nazvaný *EdMetrics*, který se tímhle (a dalšími věcmi) zabývá. Dodává se společně s distribucí Textures, ale de facto je to volný software, který lze najít na CTANu v `systems/mac/textures/utilities/EdMetrics.sea.hqx`

Uživatelé MS-DOSu mohou koupit (viz Otázku ??) balík Font Manipulation Tools od Y&Y, který mezi jinými lahůdkami obsahuje silný program *afmtotfm*.

61. Problémy s použitím PostScriptových fontů

Před typickým uživatelem LATEXu, zkoušejícího balík PSNFSS (viz Otázku ??), se často vynoří tři problémy.

První: máte oznámit `dvi` ovladači, že používáte PostScriptové fonty. V případě, že se jedná o *dvips*, znamená to přidání řádku do souboru `psfonts.map`. Jinak *dvips* zkusí najít `pk` soubory.

Pokud font není zabudován do tiskárny, musíte jej nějak získat (v mnoha případech to znamená koupi od komerčního distributora). Pak je třeba instruovat ovladač, aby zavedl fonty při každém tisku (mechanismus závisí na vašem ovladači). Takže není zrovna dobré prostě instalovat *metriky* pro Optimu a čekat, že to pojede. Musíte zaplatit těžké peníze za vlastní font, který vám dodají (pro uživatele Unixu a MS-DOSu) v `pfb` (Printer Font Binary) tvaru.

Druhá: nemůžete očekávat, že váš prohlížeč najednou začne zobrazovat PostScriptové fonty. Mnohé z nich vědí pouze o PK bitových mapách fontů jako třeba Computer Modern. Můžete je vytvořit z PostScriptových fontů, pokud jste si koupili `pfb` soubory, použitím *ps2pk* (`fonts/utilities/ps2pk`). Tak se také dají použít tyto fonty pro ovladače zařízení bez PostScriptu, jaké jsou třeba v emTEXu. Je na vaší zodpovědnosti ujistit se, že neporušujete licenční omezení na fonty, které jste koupili.

Třetí: natahování a smršťování (rozpal) mezi slovy je funkcí metriky fontu. Není specifikováno v `afm` souboru, takže rozdílné konvertory vybírají rozdílné hodnoty.

Když nejste pozorní, PostScriptová metrika, kterou přináší PSNFSS, dělá celkem těsné nastavení, které může produkovat příliš velký počet rozdělených slov nebo přetečených boxů.

Pokud máte potíže, Alan Jeffrey doporučuje následující nastavení v souboru balíku:

```
\tolerance800
\emergencystretch2em
\doublehyphendemerits5000
\hfuzz0pt
\leftskip0pt minus 1pt
\rightskip0pt minus 1pt
```

Budoucí verze těchto fontů bude mít implementovánu volnější implicitní sazbu, takže tato nastavení by se měla stát zbytečnými.

62. Výběr zvětšovatelných obrysových fontů

Pokud se zabýváte samotným textem, můžete použít libovolný z 20 000 fontů(!) ve formátu Adobe Type 1 (zvané PostScriptové fonty ve světě TEXu a ATM fonty ve světě DTP), nebo některý z několika set fontů v TrueType formátu. To samozřejmě za podmínky, že váš prohlížecí a tiskový ovladač podporuje zvětšovatelné obrysové fonty.

TEX sám se stará *jenom* o metriky, nikoliv o skutečný vzhled znaků. Je potřeba pouze vytvořit TEXovský soubor s metrikou `tfm` použitím nějakého prostředku jako *afm2tfm*, *afmtotfm* (od Y&Y, (viz Otázku **??**) nebo *fontinst*. Pro prohlížecí nebo tiskový ovladač potřebujete skutečné obrysové fonty (`pfa` pro zobrazení PostScriptu, `pfb` pro ATM na IBM PC, Macovské obrysové fonty na Macintoshi).

Jestliže také potřebujete matematiku, jste poněkud omezeni požadavky, jež má TEX na matematické fonty (detaily viz příspěvek B.K.P. Horna ve sborníku konference TUGu konané v roce 1993 v Astonu, otištěný v TUGboatu, **14**(3), 1994). Pro matematiku je tedy několik možností:

Computer Modern (75 fontů — optické zvětšování) Donald E. Knuth.

Poznamenejme, že CM jsou dostupné ve zvětšovatelné obrysové formě. Existují komerční i volně dostupné verze, jsou jak Adobe Type 1, tak TrueType verze. Některé z nich mají „komerční úroveň“ s plně ručně laděným ovládáním, další vyhlížejí velmi uboze, zatímco jiné jsou pouze nekompatibilní s Adobe Type Managerem (ATM).

Lucida Bright a Lucida New Math (25 fontů) Chuck Bigelow a Kris Holmes.

Lucida je rodina příbuzných fontů včetně patkových, bezpatkových, bezpatkových s pevnou šířkou, kaligrafických, blackletter, faxových, ručně psaných spojovaných Holmesových fontů atd. Nejsou tak „vřetenovité“ jako Computer Modern s velkým x-ovým rozměrem, zahrnují rozsáhlejší sadu matematických symbolů, operátorů, relačních a ohraničovacích znaků, než CM (přes 800 na rozdíl od 384; mimo jiné obsahuje také AMS `msam` a `msbm` sadu symbolů). Plánované „Lucida Bright Expert“ (14 fontů) připojí patkové písmo s pevnou šířkou, další ručně psaný font, malé kapitálky, tučné matematické písmo, stojatou „matematickou kurzívu“ atd. Distribuční sada obsahuje podporu pro `plain` TEX a LATEX 2.09. Podpora pro LATEX 2_ε je zajištěna v PSNFSS (viz Otázku **??**) díky Sebastianu Rahtzovi.

MathTime 1.1 (3 fonty) TEXplorators (Michael Spivak).

Sada obsahuje matematickou kurzívu, symboly a rozšíření fontů, navržené pro snadnou práci s Times-Roman. Jsou zpravidla používány s písmem Times, Helvetica a Courier (jež jsou rezidentní na mnoha tiskárnách a jsou podporovány některými verzemi PC). Navíc tuto můžete základní sadu doplnit fontem Adobe Times Smallcap a snad i sadou fontů Adobe „Math Pi“, která zahrnuje blackboard bold, blackletter a script. Distribuce obsahuje podporu pro `plain`TEX a LATEX 2.09 (včetně kódu pro spojení s Adobe Math Pi 2 a Math Pi 6). Podpora pro LATEX 2_ε je zajištěna v PSNFSS (viz Otázku **??**) díky Sebastianu Rahtzovi.

Adobe Lucida, LucidaSans a LucidaMath (12 fontů).
: Lucida a LucidaMath jsou obecně považovány za poněkud tučné. Tři matematické fonty obsahují pouze glyfy z matematické kurzívy CM, symboly a rozšířené fonty. Podpora pro použití LucidaMath v TEXu není moc dobrá, musíte si dát práci s překódováním fontů atd. (V jistém smyslu je tahle sada předchůdcem fontové sady LucidaBright a LucidaNewMath.)

Concrete, AMS matematické fonty atd. Donald E. Knuth a AMS.
: Někdy jsou zmiňovány jako alternativa k CM, ale ve skutečnosti jsou doplňkem, poněvadž je třeba s nimi použít alespoň základní matematické fonty CM.

Proprietary fonts Různé zdroje.
: Jelikož vysoce kvalitní sady fontů v obrysovém formátu fungujícím v TEXu poskytují vydavatelům skutečně konkurenceschopné dokumenty, zaplatili si mnozí hodně peněz za výrobu takových fontů na zakázku. Naneštěstí, tyto sady nejsou k dostání volně na trhu, ačkoli jsou pravděpodobně mnohem úplnější než sady jiné.

Mathptm (4 fonty) Alan Jeffrey.
: Sada obsahuje matematickou kurzívu, symboly, rozšířené a virtuální Roman fonty, vytvořené z fontů Adobe Times, Symbol, Zapf Chancery a Computer Modern. Mathptm jsou volně dostupné a výsledné PostScriptové soubory mohou být volně sdíleny. Obsahují mnoho CM matematických symbolů. Podpora pro LATEX 2_ε v PSNFSS (viz Otázku **??**) je díky Alanu Jeffreymu and Sebastianu Rahtzovi.

První tři sady fontů jsou dostupné ve formátech vhodných pro IBM PC/Windows, Macintosh a Unix/NeXT od Y&Y a od Blue Sky Research (detaily viz Otázku **??**). Fonty MathTime lze také získat od: TEXplorators, 1572 West Gray #377, Houston TX 77019, USA. Velmi omezený výběr matematických fontů je přímým důsledkem toho, že musí být navrženy pro použití v TEXu, čímž ztrácí něco ze své přitažlivosti pro jiné trhy. Navíc, trh s komerčními fonty pro TEX je nepatrný (pro porovnání, například balíků fontů #1 Microsoft TrueType se prodalo asi 10 miliónů kopií během několika týdnů po vydání Windows 3.1!).

Textové fonty formátu Type 1 jsou dostupné od mnoha prodejců včetně Adobe, Monotype, Bitstream. Vyhněte se levným produktům neznámých výrobců: nejen že byste podporovali jejich neetické chování a poškozovali skutečné tvůrce písem, ale také velmi pravděpodobně byste pořídili zmetek. Fonty se nemusí dobře zarovnávat, nemusí mít standardní doplněk 228 glyfů, anebo třeba neobsahují soubory s metrikami (je třeba udělat TFM soubory). Také se vyhněte TrueType fontům ode všech možných prodejců kromě velkých.

TrueType fonty jsou o řád obtížněji „hintovatelné“ než fonty Type 1, a proto TrueType fonty odjinud než od Microsoft a Apple mohou být podezřelé. Navíc můžete narazit ještě na další problémy s TrueType fonty, třeba, že servisní střediska s nimi odmítají pracovat.

63. Vkládání PostScriptových obrázků v LATEXu

LATEX 2_ε (viz Otázku **??**) obsahuje standardní balík umožňující včlenění grafiky, otáčení, práci s barvami a další rysy závislé na výstupních ovladačích. Balík je zdokumentován ve druhém vydání Lamportova manuálu (viz Otázku **??**) a dá se získat v `macros/latex/packages/graphics`

Pokud nepoužíváte LATEX 2_ε, asi nejlepší bude použít makro `psfig`, které napsal Trevor Darrell, dostupné v `graphics/psfig`

Budete také potřebovat konverzní program z `dvi` do PostScriptu, který podporuje `\special`. Ovladače zmíněné v Otázce **??** to umějí, a přinášejí s sebou verzi `psfig` připravenou pro použití s nimi. Makro `psfig` pracuje nejlépe s Encapsulovaným PostScriptem (EPS). Zejména `psfig` potřebuje, aby soubory obsahovaly BoundingBox (viz Apendix H v *PostScript Language Reference Manual*). Když nemáte EPS soubor, váš život se může zkomplikovat.

Další věcí kolem vkládání PostScriptového obrázku je, že obrázek není součástí `dvi` souboru, ale je včleněn až když použijete konvertor do PostScriptu. Důsledkem toho je, že většina `dvi` prohlížečů ukáže jenom prázdné místo rezervované TEXem pro váš obrázek, nikoliv obrázek samotný.

Anil K. Goel napsal dlouhý dokument podrobně popisující, kterak vkládat obrazce, obrázky a ilustrace do LATEXovských dokumentů. Je dostupný jako `info/figsinltx.ps`

64. Zrcadlově převrácený tisk

Pro offsetový tisk bývá často potřeba připravit zrcadlově převrácené předlohy. Postscriptovým řešením tohoto problému je napsat na začátek dokumentu v plain TEXu (nebo do preambule v LATEXu) řádek

```
\special{!userdict begin /bop-hook{hsize 0 translate -1 1 scale} def end}
```

Tím dosáhneme zrcadlového otočení kolem svislé osy.

Pokud chceme dokument převracet okolo vodorovné osy, vypadá řádek takto:

```
\special{!userdict begin /bop-hook{0 vsize translate 1 -1 scale} def end}
```

Komplexnější řešení (s možností zvětšování/zmenšování textu) je na `ftp://ftp.cstug.cz/pub/local/cstug/sojka/mirrorps/mirror.ps`

Pokud používáte emTEX s jeho ovladači (*dviscr*, *dvihplj*), lze jimi docílit zrcadlového převrácení i bez PostScriptu. Například převrácení kolem vertikální osy stránky zajistí volba `/tr7`.

M. Osobitné spôsoby sadzby

65. Kreslenie s TEXom

V LATEXu samotnom je veľa prostriedkov na kreslenie obrázkov (čo je lepšie ako importovať obrázky externé), od jednoduchých, ako je prostredie `picture`, trochu lepších ako je balík *epic*, až po výborné (ale pomalé) kreslenie s PICTEXom. V závislosti na type kreslenia by jeden zo štyroch nasledujúcich systémov mohol byť pre Vás špičkový:

1. *pstricks*; umným využívaním príkazu `\special` dáva prístup ku plnej sile PostScriptu z TEXu. Potrebujete slušný prekladač z DVI do PostScriptu, ale výsledok stojí za to. Dobre zdokumentovaný balík Vám dáva možnosť použiť nielen príkazy nižšej úrovne, ktoré vykreslujú priamky, kruhy či kružnice, ale aj makrá vysokej úrovne na manipuláciu s objektami, či textovými rámami, kreslenie stromov, či matíc, pre efekty 3D grafiky a ďalšie. Balík nájdete na `graphics/pstricks`.
2. METAPOST; mali ste radi METAFONT, ale nikdy Vás neuchvátili súbory písem? Vyskúšajte METAPOST (pozri Otázku **??**). Má všetku silu METAFONTu, ale generuje PostScript. D. Knuth ho používa v svojej práci.
3. *Mfpic*; páči sa Vám METAFONT, ale nerozumiete jeho jazyku? Tento balík (`graphics/mfpic`) vytvorí pre Vás kód METAFONTu z jednoduchšie zvládnuteľných TEXovských makier. Nemá všetku silu METAFONTu, ale priateľskejší vzhľad.
4. Páči sa Vám PICTEX, ale nemáte dosť pamäti alebo času? Pozrite sa na makrá `macros/generic/dratex` od Eitana Guarariho, ktoré sú tak silné, ako väčšina ostatných balíkov, ale je to celkom nová implementácia, dobre čitateľná a plne zdokumentovaná.

66. Dvojité riadkovanie v dokumentoch LATEXu

Píšete dizertáciu a chcete dodržať zvyklosti z éry písacích strojov? Alebo píšete článok pre redakciu časopisu, kde je požadované dvojité riadkovanie?

LATEX je systém pre sadzbu a teda konvencie tu rešpektované sú šité na výrobu kníh a časopisov. Ak vaše požiadavky pramenia zo zvyklostí pre prípravu dizertácií, treba nájsť niekoho, kto je zodpovedný za tvorbu pravidiel na písanie dizertácií, a kto vám poskytne znenie zásad pre písanie dizertácií (napr. „ak používate systém pre sadzbu, snažte sa, aby vaša dizertácia vyzerala ako dobre vysádzaná kniha“). (Ak máte požiadavky z časopisu, potom je ešte nepravdepodobnejšie, že si môžete dovoliť tieto požiadavky zmeniť.)

Ak sa vám nepodarí presvedčiť zodpovedných činiteľov alebo chcete sami zmeniť medziriadkové medzery, potom:

- Skúste zmeniť hodnotu `\baselinestretch` na 1.2-násobok základnej: `\renewcommand{\baselinestretch}{1.2}` a to môže byť už dosť, aby ste úradníka presvedčili, že pravidlá dodržiavate. Neskúšajte zmeniť `\baselineskip`: jeho hodnota sa mení pri každom rozmery meniacom príkaze.
- Alternatívne, skúste line-spacing package (balík makier na medziriadkové zmeny). Možnosti sú:
  - jednoduchý balík makier *doublespace* na dvojité riadkovanie, `macros/latex209/contrib/misc/doublespace.sty`, a
  - pokiaľ chcete väčšiu flexibilitu, použite balík makier *setspace* `macros/latex/contrib/supported/setspace/setspace.sty`, ktorý bol upravený pre LaTeX 2_ε.

67. Prostrkávaná sazba

Donald Knuth úmyslně do TEXu nedal možnost jednoduše měnit vzdálenosti mezi písmeny uvnitř slov, protože tento způsob zvýrazňování textu je z typografického hlediska nevhodný. Vznikl jako náhražka při psaní textů na starých psacích strojích.

Z typografického hlediska se dá považovat ještě za únosné, když je tento způsob zvýraznění (nazývaný rozpal) použit pro jedno nebo dvě slova, která se nevyskytují uvnitř hladkého textu, například nadpis D ů k a z.

Na CTANu je balík , v němž je definováno stejnojmenné makro. Používá se podobně jako `\hbox` : `\letterspace to 1.3\naturalwidth{Důkaz:}` @@@@

68. Formátovanie diplomových a disertačních prác v LaTEXu

Štýl pre dizertáciu je obvykle veľmi špecifický pre danú univerzitu. Preto spravidla nie je príliš užitočné pátrať po takomto štýle mimo rámec univerzity alebo inštitúcie. Mnoho univerzít vo svojej excentricite stále používa pre dizertácie dvojité riadkovanie. Vtedy Vás môžeme odkázať na Otázku **??**. Ak chcete napísať vlastný štýl pre dizertácie, môže Vám poslúžiť ako dobrý štart štýl University of California (dostupné v `macros/latex209/contrib/ucthesis`), ale nemá cenu púšťať sa do ťažkostí. (Ak vedenie nepripustí štandardné typografické konvencie, aj tak nebudete schopný vytvoriť esteticky uspokojivý dokument.)

69. Obtekanie obrázkov v LaTEXu

Existuje niekoľko LaTEXovských balíkov, ktoré majú za cieľ toto umožňovať, ale všetky majú svoje ohraničenia, pretože TEX nebol navrhovaný na riešenie takejto úlohy. Piet van Oostrum, ktorý vytvoril prehľad dostupných balíkov, odporúča:

picins `picins.sty` je časť veľkého balíku umožňujúceho zahrnutie obrázkov (napr. so šedými boxami, rôznymi DOS formátmi, ...). Nachádza sa v `systems/msdos/picins/picins.zip`. Príslušný príkaz je:
`\parpic(`*šírka*`,`*výška*`)(`*x-off*`,`*y-off*`)[`*Options*`][`*Pozícia*`]{`*Obrázok*`}`
Text odstavca
Všetky parametre okrem *Obrázok* sú voliteľné. Obrázok môže byť umiestnený vľavo alebo vpravo, orámovaný obdĺžnikom, oválom alebo čiarkovane. Môže mať popis (`\caption`), ktorý bude zahrnutý do zoznamu obrázkov.
Nanešťastie (pre tých, ktorí nerozumejú nemecky) je dokumentácia v nemeckom jazyku. Piet van Oostrum ale napísal anglické resumé, ktoré je v `macros/latex209/contrib/picins/picins.txt`

floatflt `macros/latex/contrib/other/floatflt` je zdokonalená verzia `floatfig.sty` (pre LaTEX 2_ε), a jeho syntax je:
`\begin{floatingfigure}[`*options*`]{`*šírka obrázku*`}`
  Obrázok
`\end{floatingfigure}`
Existuje aj (viac-menej podobné) prostredie `floatingtable`.

Tabuľky alebo obrázky môžu byť dané na pravú/ľavú časť strany a párnu/nepárnu stranu pri dokumentoch, ktoré rozlišujú párne a nepárne strany.

Tento balík umožňuje použiť aj balík *multicol*, ale nespolupracuje s ďalšími blízkymi prostrediami.

wrapfig `macros/latex/contrib/other/misc/wrapfig.sty` má syntax:

`\begin{wrapfigure}[`*výška obrázku v riadkoch*`]%`
`{l,r,`*atď*`}[`*presah*`]{`*šírka*`}`
obrázok, popis obrázku, atď.
`\end{wrapfigure}`

Syntax prostredia `wraptable` je podobná.

Výška môže byť vynechaná. Vtedy je dopočítaná automaticky programom. Program vždy berie väčšiu hodnotu z vypočítanej a udanej hodnoty. Parameter `{l,r,`*atď*`}` môže byť tiež špecifikovaný ako `i` (inside – vnútro) alebo `o` (outside – vonkajšok), pre dokumenty, ktoré rozlišujú párne a nepárne strany. Keď použijeme veľké písmená, obrázok môže byť pohyblivý. Parameter *presah* dovoľuje presah obrázku do okrajov. Obrázok alebo tabuľka môžu byť zahrnuté do zoznamu, keď použijeme príkaz `\caption`.

Uvedené prostredia nepracujú vnútri zoznamov, ktoré končia pred dolným okrajom obrázku alebo tabuľky, ale môžu byť použité v prostrediach `parbox`, `minipage`, a to aj pri dvojstĺpcovej sadzbe.

70. Alternatívne head- a footline v LaTeXu

Štandardný LaTeX zahŕňa v sebe niekoľko štýlov, ktoré špecifikujú riadok záhlavia a spodku strany (headline, footline). Možnosti, ktoré máme na zmeny, sú veľmi obmedzené. LaTeX umožňuje mnoho, ale nie je jednoduché to urobiť. My to robiť nemusíme, urobil to Piet van Oostrum.

Balík je v adresári `macros/latex/contrib/other/fancyheadings`. Tento balík poskytuje jednoduchý mechanizmus na definovanie rôznych variantov záhlavia a spodnej časti strany. Adresár obsahuje aj celkom dobrú dokumentáciu a ďalší jeden alebo dva balíky. Balík *fancyheadins* umožňuje aj predefinovať štýl strany a úvodnej strany (pozri Otázku **??**).

71. Doslovné (verbatim) vloženie súboru v LaTeXu

Dobré riešenie je použitie `verbatim.sty` od Rainera Schöpfa, ktoré poskytuje príkaz `\verbatiminput`, ktorý určuje meno vstupného súboru ako argument. Balík je časťou `macros/latex/packages/tools`.

Iná metóda je použitie prostredia `alltt`, ktoré vyžaduje `alltt.sty` (tento je teraz štandardná časť LaTeXu).

72. Generovanie registra v TeXu/LaTeXu

Vytvorenie registra – indexu, nie je jednoduchá úloha. Čo to je index a ako ho robiť je ťažko rozhodnúť. Rovnako je ťažké urobiť všeobecnú dohodu, ako je správne index robiť. V každom prípade musíte označiť všetky položky v texte (obvykle príkazom `\index`, ktoré majú byť indexované.

Triedenie veľkého indexu priamo v TeXu nie je praktické, preto sa používajú postprocessing programy, ktoré spracovávajú index mimo TeXu a register je dostupný v ďalšom behu programu TeX.

Dostupné sú nasledujúce programy:

makeindex pre LaTeX pod Unixom (ale pracuje aj pod inými operačnými systémami rovnako). Je v `indexing/makeindex`. Verzia pre Macintosh je dostupná ako `systems/mac/macmakeindex.sit`, pre MS-DOS je súčasťou distribúcie emTeXu a gTeXu (emTeX má verziu aj pre OS/2).
Dokumentácia pre *makeindex* predstavuje dobrý zdroj informácií ako vytvárať register. *Makeindex* môžeme použiť aj s niektorými balíkmi makier TeXu, ako je napríklad *Eplain.*

idxtex pre LaTeX pod VMS. Je dostupný (spolu s „poznámkovačom“ `glotex` v `indexing/glo+idxtex`).

texindex Veľmi vtipná malá utilita pre LaTeX pod operačným systémom Unix. Je dostupná v `support/texindex`.

Existujú aj iné programy s menom *texindex*, napríklad ten, ktorý je súčasťou distribúcie Texinfo (pozri Otázku **??**).

73. Používanie BIBTEXu s plainTEXom

Súbor `macros/eplain/btxmac.tex` obsahuje makrá a dokumentáciu pre BIBTEX s `plain` TEXom alebo priamo s Eplainom (pozri Otázku **??**). Pozri aj Otázku **??**, ktorá obsahuje ďalšie informácie o BIBTEXu samotnom.

74. Sadzba nôt v TEXu

Výkonný balík, ktorý umožňuje sadzbu zložitých notových osnov, je MusicTEX, ktorý napísal Daniel Taupin (`taupin@rsovax.lps.u-psud.fr`). Je dostupný v `macros/musictex`. V nedávnej minulosti Daniel (ako aj mnohí ďalší, ako Ross Mitchell a Andreas Egler) pracovali na vývoji MusicTEXu, ktorý má názov MusiXTEX. MusiXTEX je trojprechodový systém (s predprocesorovým programom, ktorý počíta medzery pre prvky) a dosahuje jemnejšiu kontrolu, ako je možné dosiahnuť v nemodifikovanom MusicTEXu. Daniel Taupin a Andreas Egler vyvíjajú rôzne verzie MusiXTEXu. Obe sú dostupné v `macros/musixtex/taupin` a `macros/musixtex/egler`.

Fanúšikovia digitálnej hudby môžu sádzať notáciu použitím *midi2tex*, ktorý prekladá MIDI súbory do zdrojového kódu MusicTEXu. Je dostupný z `support/midi2tex`.

O dosť jednoduchší zápis ako MusicTEX má *abc2mtex*. Je to balík navrhnutý pre zápis melódie vo forme ASCII znakov (`abc` notácia). Táto notácia bola navrhnutá hlavne pre zápis ľudových melódií západoeurópskeho pôvodu (írskych, anglických a škótskych), ktoré môžu byť zapísané aj klasickým spôsobom na jednej notovej osnove. Ale môže byť použitá aj na mnoho iných typov hudby. Balík je dostupný v `support/abc2mtex`.

Pre diskusiu o problémoch sadzby nôt v TEXu je vytvorená diskusná skupina. Pre subskripciu je treba poslať na adresu `mutex-request@stolaf.edu` mail, ktorý obsahuje jediné slovo `subscribe`.

75. Kreslenie Feynmanových diagramov v LATEXu

Makro-balík pre kreslenie Feynmanových diagramov v LATEXu od Michaela Lavine je dostupný v `macros/latex209/contrib/feynman`.

Môžeme tiež použiť `macros/latex/contrib/supported/feynmf`, balík makier pochádzajúci od Thorstena Ohla, ktorý pracuje v kombinácii s METAFONTom (alebo METAPOSTom).

Makrá z balíkov *feynmf* alebo *feynmp* čítajú popis diagramu, ktorý je napísaný v TEXu, a zapisujú svoj kód. METAFONT (alebo METAPOST) potom môže vytvoriť v ďalšom behu LATEXu príslušné písma. Pre nových používateľov, ktorí majú možnosť použiť METAPOST, možno odporúčať PostScriptovú verziu pre lepšiu prenositeľnosť dokumentov, ako aj niektoré ďalšie dôvody.

N. Jak udělám *X* v TEXu nebo LATEXu?

76. Prostředí důkaz (proof)

Není možné vytvořit prostředí `proof` (matematický důkaz), které by umožňovalo automatické vkládání značky konce důkazu. Některé důkazy končí ve vysazené (displayed) matematice, jiné nikoli. Jestliže vstup obsahuje `...\] \end{proof}` pak LATEX ukončí vysazenou matematiku a je připraven sázet nový řádek, aniž by předtím četl jakoukoli informaci spojenou s ukončením důkazu. Značka konce důkazu se tedy vysází na nový řádek, avšak podle tradic sazby se na nový řádek sázet nemá, a proto je ho třeba při psaní každého důkazu vkládat ručně.

77. Matematické věty a definice sázené antikvou

Chceme-li využít výhod příkazu `\newtheorem`, a přitom nechceme být omezeni na to, že obsah bude vysázen skloněným fontem (např. k vytvoření poznámek, příkladů, důkazů,. . .) pak použijeme styl theorem.sty (macros/latex/packages/tools). Popřípadě můžeme definovat prostředí, např. remark, jehož obsah bude sázen antikvou (roman):

```
\newtheorem{preremark}{Poznámka}
\newenvironment{remark}%
  {\begin{preremark}\rm}{\end{preremark}}
```

Prostředí nebude správně fungovat v případě, že mimo LaTeX 2_ε (viz Otázku ??) použijeme NFSS (viz Otázku ??). Příkaz \rm má totiž v tomto případě odlišný význam.

78. Označení číselných množin

Je jistě dobré mít příkazy typu \R pro standardní množinu reálných či jiných čísel. Tradičně byly tyto množiny sázeny tučně. Matematici však obvykle nemají přístup k tlustým křídám, a proto si vymysleli speciální symboly, které se nyní běžně používají pro označení \R, \C, atd. Tyto symboly jsou označovány jako „zdvojená latinka“ (blackboard bold). Dříve, než začneme trvat na použití těchto symbolů, bylo by dobré se zamyslet, zda by nebylo lépe se navrátit k původnímu označování číselných množin prostým tučným písmem, což by činilo věc daleko jednodušší.

Sada velkých písmen zdvojené latinky je součástí AMS-fontů „msam“ (např. „msam10“ pro velikost 10pt) a „msbm“. Fonty obsahují veliké množství matematických symbolů, které podstatně rozšiřují sadu symbolů dodávanou ve standardní distribuci TeXu. Zdrojové soubory fontu jsou v fonts/ams/amsfonts/sources/symbols

Součástí balíku jsou i dva soubory, z nichž jeden usnadňuje vkládání fontů do dokumentu a ve druhém jsou definovány příkazy pro jednotlivé symboly. Oba lze použít jak v TeXu tak v LaTeXu. Otázky a doporučení tykající se těchto fontů směrujte na tech-support@math.ams.org.

Jiná úplná sada fontů zdvojené latinky, rodina fontů bbold, je dostupná ve zdrojových souborech pro METAFONT v adresáři fonts/bbold. Zajímavostí je, že tyto fonty obsahují kromě velkých i malá písmena, které jsou na skutečných tabulích vidět jen zřídka.

Lze vytvořit jednoduchá makra pro tvorbu zdvojené latinky:

```
\newcommand{\R}{{\sf R\hspace*{-0.9ex}%
  \rule{0.15ex}{1.5ex}\hspace*{0.9ex}}}
\newcommand{\N}{{\sf N\hspace*{-1.0ex}%
  \rule{0.15ex}{1.3ex}\hspace*{1.0ex}}}
\newcommand{\Q}{{\sf Q\hspace*{-1.1ex}%
  \rule{0.15ex}{1.5ex}\hspace*{1.1ex}}}
\newcommand{\C}{{\sf C\hspace*{-0.9ex}%
  \rule{0.15ex}{1.3ex}\hspace*{0.9ex}}}
```

která lze s úspěchem použít v případě, že okolní text je sázen cmr10. Avšak, jelikož se nejedná o řádné matematické fonty, nelze je užít v matematických indexech. Krom toho velikost a poloha vertikální linky je ovlivněna výběrem fontu okolního textu (makra obsahují ex).

79. Jak vysázím značku stupně v TeXu

Pomocí makra \degree definovaného takto: `\newcommand{\degree}[1][]{\ensuremath{{#1}^\circ}}`

80. Jak vysázím „středoevropské“ uvozovky

Uvozovky v češtině, slovenštině, jakož i jiných středoevropských jazycích, mají jiný tvar než anglické “”, které TeX sází implicitně. Nejjednodušší způsob, jak vysázet text v „našich“ uvozovkách, je pomocí makra \uv: v~\uv{našich} uvozovkách. Makro \uv je součástí balíků czech.sty a slovak.sty.

Lepším a obecnějším řešením je makro Tomáše Hály \uviq, které dovoluje uvozované texty vnořovat.

Jiným řešením je použití stylu csquote.sty. Po načtení tohoto souboru (v plainu \input csquote.sty, v LaTeXu \usepackage{csquote}) lze na místě všech (otevíracích i uzavíracích) uvozovek psát jediný znak " („neorientovanou uvozovku“). Všechny liché výskyty tohoto znaku se nahradí otevírací uvozovkou, všechny sudé výskyty se nahradí uzavírací uvozovkou.

81. Chci desetinnou čárku místo tečky

TeX implicitně předpokládá, že v zápisu desetinných čísel odděluje desetiny od jednotek desetinná tečka. Čárku zapsanou v matematickém módu TeX považuje za oddělovač položek v posloupnostech, množinách apod., a proto za ni sází malou mezeru.

Chceme-li jednorázově vysázet desetinné číslo s desetinnou čárkou, stačí čárku uzavřít do skupiny: `$3{,}14159$`.

Takové řešení je však nevýhodné pro texty, kde je desetinných čísel mnoho. V takovém případě lze nastavit globálně, že za čárkou v matematickém módu se mezera sázet nemá:

```
\mathcode`,="002C
$3,14159$
```

Pokud při tomto globálním nastavení přesto někde mezeru udělat potřebujeme, musíme ji napsat explicitně: `$(m,\,n)$` nebo `$\{0,\:1,\:2\}$`.

Jiným řešením je ponechat ve vstupním souboru desetinné tečky, ale přemapovat na čárky (to se hodí třeba tehdy, když máme velké soubory desetinných čísel z jiných výstupů).

```
\mathcode`.="002C
$3.14159$
```

82. Označování položek výčtů

Chceme-li, aby nejvyšší úroveň číslovaného výčtu `enumerate` byla označena „I/“, „II/“, ..., musíme použít příkazy:

```
\renewcommand{\theenumi}{\Roman{enumi}}
\renewcommand{\labelenumi}{\theenumi/}
```

Možné styly číslování jsou popsány v oddíle 6.3 Lamportova manuálu (viz Otázku ??). Jsme-li pohodlní a po změně `\labelenumi` provedeme pouze jedno zpracování LaTeXem, budou naše křížové odkazy chybné.

Pro nižší úrovně výčtu `enumerate` je třeba zaměnit `enumi` v předchozím příkladě za `enumii`, `enumiii` či `enumiv`, podle úrovně, kterou chceme upravovat. Je-li nové označení položky širší než označení původní, je také třeba změnit `\leftmargini`, `\leftmarginii` atd.

Použijeme-li styl `enumerate.sty` (`macros/latex/packages/tools`) pro LaTeX 2_ε, který řeší obdobný problém, získáme stejný výsledek jako předchozím příkladě tak, že na začátek výčtu umístíme příkaz `\begin{enumerate}[I/]`.

83. Nečíslované oddíly v obsahu

Nejjednodušším způsobem, jak vytvořit v obsahu položku pro takové zvláštní části textu jako jsou například předmluvy, je užití čítače `secnumdepth`, který je popsán v dodatku C manuálu LaTeXu. Například:

```
\setcounter{secnumdepth}{-1}
\chapter{Předmluva}
```

Samozřejmě, že před opětovným použitím číslovaného oddílu je potřeba nastavit parametr `secnumdepth` zpět na jeho obvyklou hodnotu (ve standardních stylech 2).

Následuje vysvětlení. Použije-li se `\chapter` bez hvězdičky:

1. vloží se něco do `.toc` souboru;
2. je-li `secnumdepth` ≥ 0, zvětší se čítač kapitoly o jedničku a vysází se jeho hodnota.
3. vysází se název kapitoly.

Ostatní příkazy na rozdělování textu (`\section`, `\subsection`,...) pracují obdobně, avšak testují jinou hodnotu.

84. Vzhled nadpisů

Předpokládejme, že editor našeho oblíbeného časopisu vyžaduje, aby nadpisy oddílů byly centrované a vysázené kapitálkami a pododdíly byly zarovnány vpravo a vysázeny kurzívou. Nechce se nám však zaplétat do takového druhu programování jaký je popsán v *The LaTeX Companion* (viz

Otázku ??; programování samotné je diskutováno v Otázce ??). Následující trik pravděpodobně editory uspokojí. Nově definované příkazy `\ssection` a `\ssubsection`:

```
\newcommand{\ssection}[1]{%
   \section[#1]{\centering\textsc{#1}}}
\newcommand{\ssubsection}[1]{%
   \subsection[#1]{\raggedright\textit{#1}}}
```

se pak použijí namísto obvyklých `\section` a `\subsection`. Není to však nejhezčí řešení. Čísla v nadpisech oddílů zůstávají vysázena tučně a hvězdičkovaná forma příkazů musí být předefinována zvlášť. Uvedené definice nebudou správně fungovat v případě, že se použije NFSS (viz Otázku ??) mimo LaTeX 2_ε (viz Otázku ??), jelikož v takovém případě se příkazy měnící typ písma chovají odlišně.

85. Odsazení odstavce po nadpisu

LaTeX používá stylů, jež neodsazují první odstavec za nadpisem. Je mnoho důvodů proč tomu tak je, ale ne každému to vyhovuje (např. to neodpovídá tradicím české typografie). Užijeme-li styl `indentfirst.sty` (`macros/latex/packages/tools`), pak se tento mechanismus potlačí, takže první odstavec bude odsazen.

86. Změna okrajů v LaTeXu

Neměňte okraje v LaTeXu, naučte se LaTeX, napište několik dokumentů a teprve poté se zkuste zeptat znovu: „Jak změním okraje v LaTeXu“.

Nikdy nelze změnit *skutečné okraje* dokumentu pomocí programu, jelikož velikost okrajů závisí na konkrétní velikosti použitého papíru, která se může lišit od velikosti, se kterou pracuje počítač (například standardní americký papír má jinou šířku než papír formátu A4). Co však lze měnit, jsou vzdálenosti od domnělého horního a levého okraje papíru a šířka a výška textového zrcadla. Změna šířky a výšky textu vyžaduje daleko více zručnosti, než by se dalo čekat. Výška textu by měla mít jistý vztah ke vzdálenosti po sobě následujících účaří (`\baselineskip`). Šířka by neměla být větší než 75 znaků. Lamportova upozornění v kapitole „Customizing the Style“ bychom měli brát vážně. Jednopalcové okraje na papíru formátu A4 jsou vhodné pro psaní na stroji, ale nehodí se pro sazbu desetibodovým písmem, jelikož tolik znaků na jednom řádku zhoršuje čitelnost.

87. Trvání na změně okrajů v LaTeXu

V této odpovědi se nejprve naučíme změnit okraje v celém dokumentu, poté se dozvíme, jak změnit okraje jenom v jeho části.

Snad nejsnazším způsobem, jak získat více místa na stránce, je použít stylu `macros/latex209/contrib/misc/fullpage.sty`, který nastaví okraje stránky shodně s `plain` TeXem, t.j. jednopalcové okraje od všech čtyř stran papíru. Kromě standardního amerického formátu obsahuje i nastavení pro formát A4.

Lepší je však `macros/latex/contrib/other/misc/vmargin.sty`, který obsahuje nadefinovanou množinu velikostí papíru (nadmnožinu těch, jež obsahuje LaTeX 2_ε), možnost definování vlastní velikosti papíru, nastavení okrajů a podporu dvoustranného tisku.

Následuje krátké vysvětlení parametrů popisujících stránku v LaTeXu. Podrobnosti jsou probrány v oddíle C.5.3 manuálu LaTeXu (str. 181–182). Počátek souřadného systému `dvi` souboru je jeden inch od horního a jeden inch od levého okraje papíru. To vysvětluje, proč v manuálu LaTeXu je používáno soustavně o jeden inch méně. V souřadném systému `dvi` souboru rostou kladné hodnoty vpravo v horizontální a dolů ve vertikální ose. Proto chceme-li, aby okraje textu byly blíže než jeden inch k levému nebo hornímu okraji, musí být hodnoty odpovídajících parametrů, t.j. `\evensidemargin`, `\oddsidemargin`, `\topmargin`, záporné.

Parametry `\topmargin`, … neumožňují změnit velikosti okrajů uvnitř dokumentu, lze je měnit pouze v preambuli zdrojového textu, t.j. před příkazem `\begin{document}`. K nastavení okrajů uvnitř dokumentu lze definovat prostředí:

```
\newenvironment{changemargin}[2]{%
 \begin{list}{}{%
```

```
    \setlength{\topsep}{0pt}%
    \setlength{\leftmargin}{#1}%
    \setlength{\rightmargin}{#2}%
    \setlength{\listparindent}{\parindent}%
    \setlength{\itemindent}{\parindent}%
    \setlength{\parsep}{0pt plus 1pt}%
   }%
  \item[]}{\end{list}}
```

Prostředí umožňuje odsadit text zleva a zprava o hodnoty uvedené v jeho prvním a druhém parametru. Záporné hodnoty vedou k zúžení okraje; například `\begin{changemargin}{-1cm}{-1cm}` zúží levý a pravý okraj o 1cm. (Je třeba upozornit, že hodnota `0pt plus 1pt` přiřazená `\parsep` je základní hodnotou pro LaTeX; správně by tam však měla být hodnota stejná jako v okolním textu.)

88. Zjištění šířky písmene, slova nebo sousloví

Slovo se vloží do boxu a poté se zjistí šířka tohoto boxu. Například:

```
\newdimen\stringwidth
\setbox0=\hbox{hi}
\stringwidth=\wd0
```

Je třeba však mít na paměti, že je-li v `\hbox`u sousloví, hodnota, kterou určíme, bude pouze přibližnou šířkou, kterou bude vysázené zaujímat ve skutečném textu, neboť mezislovní mezery se při sazbě odstavce mohou změnit.

V LaTeXu lze problém řešit takto:

```
\newlength{\gnat}
\settowidth{\gnat}{\textbf{small}}
```

Hodnota `\gnat` se v tomto případě nastaví na hodnotu šířky „small" vysázeného tučně.

89. Vyloučení části textu z `.dvi` souboru

`verbatim.sty` Rainera Schöpfa zavádí prostředí `comment`, které umožňuje část textu mezi `\begin{comment}` a `\end{comment}` označit jako komentář a tedy vyloučit z `.dvi`. Balík lze nalézt na CTANu v adresáři `macros/latex/packages/tools`.

Obecnější prostředí, které rovněž umožňuje vypouštět části textu, `macros/latex/contrib/other/comment/comment.sty` Viktora Eijkhouta, nabízí navíc jednoduché prostředky pro definici vlastních prostředí pro ponechání nebo přeskočení části textu v dokumentu, přičemž lze nastavit, zda prostředí právě slouží pro ponechání textu nebo jeho vynechání.

90. Definice nových označení matematických funkcí v LaTeXu

Použije se příkaz `\mathop` podobně jako v tomto příkladě:

```
\newcommand{\diag}{\mathop{\rm diag}}
```

Spodní a horní indexy funkce `\diag` budou umístěny obdobně jako u `\lim` (limita). K tomu, aby indexy byly vždy umístěny vpravo od označení funkce, se použije tato definice:

```
\newcommand{\diag}{\mathop{\rm diag}\nolimits}
```

Tyto definice fungují pouze v LaTeXu 2$_\varepsilon$ a LaTeXu 2.09, avšak nelze je použít spolu s NFSS (viz Otázku ??). V LaTeXu 2$_\varepsilon$ lze problém řešit i tak, že se použije příkaz `\operatorname` z `amstex.sty`, které je součástí $\mathcal{AMS}$-LaTeXu (`fonts/ams/amslatex`).

91. Sazba názvů programů používaných spolu s TeXem

Donald Knuth definoval logo TeXu jako konkrétní ukázku toho, co TeX umí. Žel, mnozí si však myslí, že tím přivodil záplavu různých log, jako například $\mathcal{AMS}$-TeX, PiCTeX, BibTeX, atd. Lamport vymyslel logo LaTeX, což pak přirozeně vedlo ke vzniku nynějšího loga LaTeX 2$_\varepsilon$.

Není však nutně třeba uvádět tato loga v jejich komplikovaném tvaru všude tam, kde se příslušný název vyskytne, avšak pro ty, kdo trvají na jejich použití, existuje v souboru `macros/eplain/texnames.sty` veliké množství předdefinovaných log. Fonty loga METAFONTu mohou být přidány k fontům, které LaTeX 2ε již zná (tak, že se pak může měnit jeho rozměr spolu s okolním textem) použitím balíku `macros/latex/contrib/supported/mflogo`.

Pro ty, kdo nepotřebují mít v dokumentu vysázená ta „správná" loga, stačí použít `AMS-\TeX{}` (AMS-TeX) namísto AMS-TeX, `Pic\TeX{}` (PicTeX) namísto PICTEX, `Bib\TeX{}` (BibTeX) namísto BIBTEX, atp.

O. Něco nefunguje tak, jak by mělo

92. Nezvykle rozdělená slova

Pravděpodobně je problém v pomíchaných verzích TeXu. Systém dělení se totiž s přechodem od verze 2.9 na 3.0 změnil. Používá-li se (`plain`) TeX verze 3.0 nebo pozdější, je třeba se ujistit, že soubor `plain.tex` je alespoň verze 3.0. Používá-li se LaTeX 2.09 (poslední verze byla uvedena 25.března 1992), bylo by dobré přejít na LaTeX 2ε, jelikož zdrojové soubory LaTeXu 2.09 již nejsou veřejně k dispozici.

Používá-li se LaTeX 2ε, je příčina chybného dělení pravděpodobně v souboru `lthyphen.cfg`, který musí být vytvořen, používáme-li více jazyků najednou.

Pro toho, koho to zajímá, problém spočívá v tomto: V TeXu do verze 3.0 neumožňoval algoritmus dělení slov rozdělit slovo, pokud první část rozděleného slova neobsahovala alespoň dva znaky a jeho druhá část alespoň znaky tři. Počínaje verzí 3.0 lze řídit minimální velikost těchto částí pomocí celočíselných parametrů `\lefthyphenmin` a `\righthyphenmin`. V nových formátech `plain` a `lplain` jsou nastaveny na hodnoty 2 a 3. Mohou být samozřejmě nastaveny na libovolnou hodnotu, s tím, že překročí-li součet `\lefthyphenmin` + `\righthyphenmin` hodnotu 62, nebude docházet k žádnému dělení slov.

93. cslatex ignoruje `\hyphenation` a `\righthyphenmin`

V českých a slovenských dokumentech `cslatex` tyto deklarace ignoruje, přestože v LaTeXu 2.09 to fungovalo. Příčina rozdílného chování tkví v tom, že styly `czech.sty` a `slovak.sty`, které `cslatex` používá, vkládají přepínací makro `\czech` resp. `\slovak` prostřednictvím `\AtBeginDocument`. Všechny změny nastavované v preambuli se tudíž týkají implicitního `\language 0`, jímž je americká angličtina. Toto chování má racionální důvod. Pokud při zavádění stylu pro češtinu (nadále budeme pro jednoduchost mluvit pouze o češtině, pro slovenštinu platí totéž) použijeme parametr `split`, aby se slova s rozdělovníkem dělila podle českých typografických pravidel, je znak minus aktivní a nebylo by tudíž možno zadávat záporná čísla a záporné rozměry. To by ovšem značně komplikovalo nastavování různých parametrů v preambuli dokumentu. Navíc by to způsoboovalo, že některé další styly by podle okolností někdy přestaly fungovat. Změny `\righthyphenmin` a `\lefthyphenmin` i definici výjimek dělení `\hyphenation` tedy musíme umístit až za `\begin{document}`.

Často je žádoucí, aby tyto změny byly specifikovány ve stylovém souboru. Pak ovšem musíme zajistit, aby se provedly až po nastavení češtiny. Budeme-li např. celý dokument sázet do úzkých sloupců, můžeme použít

```
\RequirePackage[split]{czech}
\AtBeginDocument{\righthyphenmin 2\relax}
\end{document}
```

Primitiv `\relax` chrání před případnou následující chybnou definicí, která by při své expanzi vytvořila číslici. Příkazem `\RequirePackage` si zajistíme, že se `czech.sty` načte dříve než náš styl a nastavení se skutečně provede pro češtinu.

Pokud vytváříme vícejazyčný dokument, je situace o něco složitější. Příkaz `\czech` totiž vždy nastaví standardní hodnoty. Pokud chceme, aby se `\righthyphenmin` po přepnutí češtiny vždy nastavil na hodnotu 2, můžeme toho dosáhnout například následujícími definicemi:

```
\RequirePackage[split]{czech}
```

```
\let\OriginalCzech\czech
\def\czech{\OriginalCzech \righthyphenmin 2\relax}
```

Specifikace výjimek dělení je poněkud jednodušší. Příkaz `\hyphenation` je vždy prováděn globálně. Abychom nepoškodili další makra případnými aktivními znaky, můžeme definice výjimek dělení uzavřít do skupiny například takto:

```
{\czech \hyphenation{po-po-ka-te-pe-tl}}
```

94. Podivné dělení slov v angličtině

Může se stát, že slavné automatické dělění slov TEXu nedělí v místech, která jsou doporučena ve slovníku. Dochází k tomu zřejmě proto, že TEX používá americkou angličtinu, jejíž pravidla pro dělení slov (tak jak jsou uvedena, například, ve Webster's Dictionary) jsou zcela odlišná od pravidel angličtiny britské (tak jak jsou uvedena, například, v Oxford Dictionaries). Na řešení tohoto problému se podílí především Sdružení uživatelů TEXu Spojeného Království (UK TEX User community) (viz jejich časopis *Baskerville*, vydání 4.4), avšak naprosto uspokojivé řešení je náročné na čas. Soubor `language/english/ukhyph.tex` obsahuje prozatímní vzory dělení anglických slov podle britských pravidel dělení.

95. Akcentované slová sa nedelia

TEXovský algoritmus delenia slov neuvažuje ako jedno slovo také, ktoré má v sebe zahrnutý príkaz `\accent`. Sú na to dobré príčiny, ale znamená to, že kvalitná sadzba neanglických jazykov je obtiažna.

Aby sa TEX vyhol týmto ťažkostiam, bolo prijaté Corkovské kódovanie (pozri Otázku **??**), ktoré obsahuje akcentované znaky ako samostatné znaky. Snáď v budúcnosti Omega (pozri Otázku **??**) poskytne celkom nové riešenie.

96. Rozšíření kapacity TEXu

Občas se stane, že se objeví na terminálu zpráva: „memory capacity exceeded“ (překročení kapacity paměti). Většinu takovýchto chyb lze vyřešit *bez* jakéhokoli rozšiřování kapacity TEXu. Nejběžnějšími příčinami jsou nepárové závorky, velmi dlouhé řádky a špatně napsaná makra. Velmi dlouhé řádky (extra-long lines) vznikají často při chybném přenosu souborů mezi rozdílnými operačními systémy a zvláště tehdy, když znaky označující konec řádky nejsou převedeny vhodným způsobem (příznakem chyby zapříčiněné velmi dlouhými řádky je chybová zpráva o překročení velikosti bufferu `buf_size`).

Pokud je skutečně třeba rozšířit kapacitu TEXu, volba vhodné metody závisí na použité instalaci. Nejčistší formou jak rozšíření provést, je změnit parametry přímo v modulu 11 zdrojového WEB souboru.

Méně čistou formou je úprava změnového souboru či změna hodnoty jisté proměnné prostředí. Například emTEX umožňuje nastavit kritéria pro velikost přidělené paměti přímo z příkazové řádky. Podrobnosti hledejte v dokumentaci, která je dodávána k vaší implementaci.

97. Plovoucí se tabulky a obrázky LATEXu

Tabulky a obrázky (figures) mají sklon k tomu, aby nás udivovaly svým *odplouváním* z místa, kde měly být původně vysazeny. Děje se tak proto, aby bylo dosaženo dokonalého vzhledu dokumentu. Každý profesionální systém určený pro sazbu nechává „odplouvat“ tabulky a obrázky na taková místa, kde jejich umístění nevede k narušení typografických pravidel. Dokonce užijeme-li parametru `h` ve zdrojovém textu, abychom dosáhli umístění „zde“ (here), obrázek či tabulka nebude vytištěna „zde“ v případě, že by to tato pravidla narušilo. Pravidla samotná jsou poměrně jednoduchá, a jsou popsána na str. 198, v kapitole C.9 manuálu LATEXu. V horším případě pravidla LATEXu mohou vést k nakupení plovoucích objektů takového rozsahu, jež vede k chybové zprávě „Too many unprocessed floats“ (Přespříliš nezpracovaných plovoucích objektů). To znamená, že omezený počet registrů, ve kterých LaTeX uchovává plovoucí objekty, je naplněn. Následuje jednoduchý přehled toho, co je třeba udělat, abychom takové problémy vyřešili (v přehledu se hovoří o obrázcích, ale totéž platí i o tabulkách).

- Zvolili jsme správné parametry umístění obrázku? Předdefinované (default) parametry (`tbp`) většinou dostačují. Nikdy by se nemělo používat, například, samotného parametru `h`, poněvadž tím je řečeno: „jestliže to nemohu umístit zde (here), nemohu to umístit ani někde jinde“ a následkem toho se za tímto obrázkem nakupí všechny následující plovoucí objekty.
- Můžeme ochránit naše obrázky od odplutí nastavením parametru LaTeXu pro umístění plovoucích objektů? I zde lze říci, že přednastavené hodnoty jsou přiměřené, avšak mohou být v případě problémů přepsány. Blíže jsou popsány na str. 199 – 200, v kapitole C.9 manuálu LaTeXu.
- Najde se v našem dokumentu místo, kam by bylo možné přirozeně vložit příkaz `\clearpage`? Jestliže ano, pak je dobré tak učinit. Nahromaděné plovoucí objekty se příkazem `\clearpage` uvolní. (Pozn.: Příkaz `\chapter` používá `\clearpage`, a proto obrázky nemohou odplout za konec stávající kapitoly.)
- Lze použít balíku `afterpage` pro LaTeX 2_ε (najdeme ho v adresáři `macros/latex/packages/tools`). V dokumentaci k tomuto balíku je na jednom příkladě uvedena myšlenka vložení `\clearpage` *po* aktuální stránce (tam, kde se uvolněním nakupených plovoucích objektů nevytvoří nevzhledná mezera v textu), avšak je tam uvedeno i to, že tento balík je poněkud křehký.
- Poslední možností, jak uvedené problémy obejít, je použití balíku `macros/latex209/contrib/misc/morefloats.sty`, jenž jednoduše zvětší počet možných plovoucích objektů, se kterými může LaTeX pracovat najednou (z 18 na 36), což by mělo ve většině případů dostačovat.
- Jestliže nastane nutnost toho, že všechny obrázky mají být umístěny na konci dokumentu (například, při zasílání článku), nelze spoléhat na to, že to LaTeX učiní za nás. Lépe je použít balíku `endfloat`, který je k takovému účelu určen (najdeme ho na CTANu v `macros/latex/contrib/supported/endfloat`).

98. Užití `\pagestyle{empty}` na první stránce v LaTeXu

Použije-li se `\pagestyle{empty}`, a přesto je první stránka označena číslem, pak je pravděpodobně užit i příkaz `\maketitle`. To není chyba, ale správná vlastnost LaTeXu! Standardní styly LaTeXu jsou psány tak, že u úvodní strany (strany obsahující příkazy `\maketitle`, `\part`, či `\chapter`) se použije styl strany (pagestyle) jiný než ve zbytku dokumentu. Uvedené příkazy spouští `\thispagestyle{plain}`. Tato vlastnost však není příliš vhodná v případě, že veškerý okolní text má nečíslované stránky (`\pagestyle{empty}`).

Změny této vlastnosti dosáhneme:

- Když vložíme `\thispagestyle{empty}` bezprostředně za příkaz `\maketitle` ; tj. mezi příkazy se nesmí vložit prázdný řádek.
- Použijeme-li `fancyheadings.sty`, který umožňuje měnit styl úvodní stránky nezávisle na stylu stránek ostatních. Lze jej nalézt v adresáři `macros/latex/contrib/other/fancyheadings`

99. Podivné chování `\rm`, `\bf`, atd.

Jestliže příkazy jako `\rm` a `\bf` v LaTeXu přestaly náhle fungovat tak, jak se od nich očekávalo, je to zřejmě proto, že administrátor systému nainstaloval verzi LaTeXu 2.09 spolu s NFSS (viz Otázku **??**). V takovém případě je dobré si mu nahlas postěžovat a požádat jej, aby tuto verzi zaměnil za LaTeX 2_ε (viz Otázku **??**), v němž příkazy `\rm`, `\bf` apod., jsou-li užity ve standardních třídách — `article`, `report` a `book` aj. — fungují stejně jako předtím. Než k záměně dojde, lze použít `oldlfont.sty`. Tento styl by měl být instalován zároveň s NFSS.

100. Odkazy na příkazy typu `\tenrm` ve starém LaTeXu

V LaTeXu 2.09 byla definována rozsáhlá množina příkazů, jež zpřístupňovaly vestavěné fonty. Například pro různé velikosti fontu `cmr` se v něm dalo použít příkazů: `\fivrm`, `\sixrm`, `\sevrm`, `\egtrm`, `\ninrm`, `\tenrm`, `\elvrm`, `\twlrm`, `\frtnrm`, `\svtnrm`, `\twtyrm` a `\twfvrm`.

Tyto příkazy nebyly nikde dokumentovány, avšak mnohé balíky jich užívaly k dosažení požadovaných efektů.

Protože však tyto příkazy nebyly veřejné, nestaly se součástí nového LaTeXu 2$_\varepsilon$. Chceme-li použít neupravené balíky původně určené pro LaTeX 2.09 v LaTeXu 2$_\varepsilon$, je potřeba použít zároveň balíku `rawfonts.sty`, který je součástí distribuce LaTeXu 2$_\varepsilon$.

101. Chybějící symboly

Nemůžeme-li se dostat k některým symbolům, jako například `\Box` and `\lhd`, pak zřejmě správce systému aktualizoval stávající verzi LaTeXu buď na NFSS (viz Otázku **??**) či LaTeX 2$_\varepsilon$ (viz Otázku **??**). V případě, že máme NSFS, užijeme `oldlfont.sty`, stejně jako v otázce **??**. V LaTeXu 2$_\varepsilon$ užijeme balík `latexsym`, jenž je součástí standardní distribuce LaTeXu 2$_\varepsilon$, anebo balík `amsfonts`, je-li instalován.

102. Chybné křížové odkazy v LaTeXu

Někdy, i když necháme zdrojový text proběhnout LaTeXem několikrát, dostáváme stále nesprávné křížové odkazy. Zapamatujme si, že příkaz `\label` musí následovat *po* příkazu `\caption`, případně musí být jeho součástí. Například,

```
\begin{figure}             \begin{figure}
\caption{Obrázek}   nebo   \caption{Obrázek%
\label{fig}                       \label{fig}}
\end{figure}               \end{figure}
```

103. `\@` a `@` ve jménech maker

Častým zdrojem problémů v dokumentech psaných v LaTeXu je výskyt příkazu `\@`, či jiných příkazů obsahujících znak `@` (at). Nejobvyklejším chybovým hlášením je: „You can't use "`\spacefactor`" in vertical mode" („Nelze použít „`\spacefactor`" ve vertikálním módu"), ale mohou se objevit i jiná obdobná hlášení.

Podobné problémy jsou obvykle způsobeny vložením souboru třídy (class file) či některého stylového souboru (package file) do dokumentu LaTeXu 2$_\varepsilon$ nevhodným způsobem, tj. bez použití příkazů `\documentclass` či `\usepackage`.

V LaTeXu se jména vnitřních maker definují obvykle tak, že obsahují znak `@`. Tímto způsobem se autoři stylů snaží vyhnout kolizi mezi názvy příkazů použitými uvnitř stylu a příkazů běžně užívanými v dokumentu. Aby však příkazy tohoto typu mohly vůbec fungovat, musí makra `\documentclass` a `\usepackage` měnit význam znaku `@`.

Problém s chybovým hlášením se jednoduše vyřeší tím, že se styly a třídy do dokumentu vkládají prostřednictvím příkazů `\usepackage` a `\documentclass`.

Je však možné namítnout: „V *The LaTeX Companion* se říká, že příkazy obsahující `@` lze použít!"

Ovšem. Například je tam dlouhý oddíl o příkazu `\@startsection` a o tom, jak jej používat k úpravě vzhledu nadpisů oddílů. Je o tom psáno na straně 15 *The LaTeX Companion*, ale je tam rovněž navrženo provádět obdobné změny v preambuli dokumentu (tj. před `\begin{document}`) mezi příkazy `\makeatletter` a `\makeatother`. Definice `\subsection` na straně 26 by mohla vypadat takto:

```
\makeatletter
\renewcommand{\subsection}{\@startsection
  {subsection}%                     % name
  ...
  {\normalfont\normalsize\itshape}}% style
\makeatother
```

104. Kde jsou fonty msx a msy

Fonty msx a msy byly navrženy Americkou matematickou společností (AMS) pro sazbu odborných článků v matematických časopisech již v prvopočátcích existence TEXu. Byly navrženy pro „starý“ METAFONT, který nebyl přenositelný a již se nepoužívá. Velmi dlouhou dobu byly tyto fonty dostupné pouze v rozlišení 300dpi, které se ovšem stěží hodí pro moderní tiskárny. AMS nyní předělala fonty pro soudobou verzi METAFONTu. Nové rodiny fontů byly pojmenovány msa a msb. Lze je získat v fonts/ams/amsfonts/sources/symbols

Přesto nás fonty msx a msy nepřestávají sužovat. Jsou doposud lidé, kteří se nerozhodli k aktualizaci fontů. Avšak, i když nakonec každý bude používat nové fonty, zůstanou problémy se starými dokumenty, které se na msx a msy odkazují.

Máme-li zdrojový .tex soubor, který vyžaduje užití msx a msy, nejlepším způsobem, jak se zbavit problému se starými fonty, je taková úprava souboru, aby se používaly msa a msb namísto msx a msy (stačí změnit pouze jedno písmeno ve jménech fontů).

Máme-li dvi-soubor, který se odkazuje na msx a msy fonty, můžeme k přesměrování starých fontů na nové použít balík virtuálních fontů (viz Otázku ??) fonts/vf-files/msx2msa

105. Kde jsou fonty am

Příležitostně se stále objevuje požadavek na soubor písem am; am je zkratka „Almost [Computer] Modern“ (téměř [Computer] Modern). Tyto fonty jsou předchůdci fontů Computer Modern, které všichni známe a máme (nebo nemáme) rádi[9]. Není toho mnoho, co by se s těmito fonty dalo ještě dělat. Jsou, jak ostatně jejich název napovídá, téměř (ale ne úplně) shodné se souborem fontů cm. Máme-li před sebou dokument, jenž požaduje am fonty, jediná rozumná věc, kterou můžeme udělat, je nahradit v dokumentu am za cm. Dvi soubory, které by požadovaly am fonty, se vyskytují tak zřídka, že se nikdo neujal náročného úkolu vytvoření transformace pomocí virtuálních fontů. Můžeme však využít toho, že většina ovladačů umožňuje použít konfigurační soubor, v němž lze zadat substituci fontů. Zadáme-li, že veškeré fonty am mají být nahrazeny odpovídajícími fonty cm, výstup by měl být téměř správný.

106. „Příliš dlouhý řetězec“ v BIBTEXu

Upozornění BIBTEXu „Warning—you've exceeded 1000, the global-string-size, for entry foo“ (Celková velikost řetězce v položce foo je větší než 1000) není toho druhu, jehož se lze vyvarovat jednoduchým pozměněním stylu BIBTEXu. Ke zvětšení přípustné délky řetězců je třeba znovu překompilovat BIBTEX, což není praktické a příliš žádoucí. Proto je třeba problém řešit změnou databáze bibliografických citací.

Problém je obvykle v tom, že databáze obsahuje velmi rozsáhlý abstrakt či anotaci. Jedinou možností nápravy je vynětí tohoto záznamu z databáze tak, aby nebyl překročen limit BIBTEXu. Avšak většinou je potřeba záznam v databázi nějakým způsobem zachovat, jelikož bude užit v sazbě. V takovém případě se záznam vloží do zvláštního souboru:

```
@article{long.boring,
  author =    "Fred Verbose",
  ...
  abstract =  "{\input{abstracts/long.tex}}"
}
```

Tímto způsobem se zařídí, že BIBTEX pracuje pouze se jmény souborů, ale v případě potřeby TEX zařídí vložení celého vlastního textu do dokumentu.

[9]Fonty získaly přívlastek „Almost“ (téměř) po tom, co jejich první implementace v METAFONTu79 nedopadla tak úplně dobře. Knuthův původní záměr byl, že tyto fonty budou konečným řešením.

P. Proč to dělá tohle?

107. Proč jsou ignorovány parametry odstavce

TEX při uspořádávání textu nepracuje tak, že by bral slovo za slovem, či řádku za řádkou. Nejmenší jednotkou, kterou TEX formátuje, jsou celé odstavce. Odstavec je načten celý do paměti a není dále zpracováván, dokud není načtena značka konce odstavce. Právě v ten okamžik se uplatní parametry odstavce. Chyby použití parametrů formátování odstavce často vznikají právě proto, že se zapomíná na to, v jaké posloupnosti se odstavec zpracovává.

Předpokládejme následující větu v LATEXu:

```
{\raggedright % zarovnávání  textu  vlevo
Tento text by měl být na výstupu zarovnán
pouze vlevo.  Chceme,  aby  se  tak stalo
pouze  v~tomto  odstavci,  a~ proto  zde
ukončíme skupinu.}

Další text je zpracován normálně...
```

TEX otevře skupinu a nastaví parametry formátování odstavce tak, aby byl text uvnitř této skupiny zarovnáván pouze vlevo, poté uloží dvě věty textu, uzavře skupinu a nastaví původní parametry odstavce. Poté načte prázdný řádek, který je zpracován stejně jako příkaz `\par`; vysází dvě věty textu. Jelikož však skupina byla ukončena před načtením konce odstavce, nastavení parametrů uvnitř skupiny ztratilo význam a odstavec bude vysázen s běžnými parametry.

Aby nastavené parametry zůstaly v platnosti po celou dobu zpracovávání odstavce, je třeba ukončit odstavec uvnitř skupiny. Nahradíme-li poslední tři řádky v předchozím příkladě za:

```
ukončíme skupinu.\par}
Další text je zpracován normálně...
```

ukončí se odstavec ve chvíli, kdy jsou parametry odstavce nastavené v uzavřené skupině stále v platnosti.

108. Proč se v LATEXu užívá ochran (protection)

LATEX si ukládá některá data, jež bude zpracovávat až později. Těmito daty jsou zejména argumenty některých příkazů, takzvané pohyblivé argumenty. Pohyblivé proto, že s daty se nějakým způsobem manipuluje. Jedná se argumenty těch příkazů, které zapisují do obsahu, seznamu tabulek atd., tj. data, která jsou zapisována do pomocného souboru, z něhož jsou později opět čtena. Jinými daty jsou ta, která se mohou objevit v záhlavích. Nejvýznamnějšími příkazy tohoto typu jsou popisy obrázků a tabulek (captions) a veškeré nadpisy. Úplný seznam lze najít v Lamportově manuálu (viz Otázku **??**).

Co se za tím vším skutečně skrývá? Příkazy, jež se použijí v pohyblivých argumentech, jsou v průběhu ukládání plně expandovány. Někdy je výsledkem takového rozvoje špatný TEXovský kód, což se projeví až při jeho následném čtení. Příkazem `\protect\cmd` je LATEXu řečeno, aby uložil `\cmd` bez expanze, jako `\cmd`.

Co je to „křehký příkaz“ (fragile command)? To je příkaz, který je během ukládání rozvinut do chybného TEXovského kódu.

Co je to „robustní příkaz“ (robust command)? To je příkaz, který je během ukládání rozvinut do správného TEXovského kódu.

Nikdo se (samozřejmě) z takto nepřehledné situace neraduje. Skupina projektu LATEX3 při práci na LATEXu 2_ε odstranila potřebu některých ochran, avšak techniky, které jsou jim dostupné v současném LATEXu, činí věc poměrně složitou. Dlouhodobým cílem této skupiny zůstává odstranění všech ochran.

109. Proč `\verb` nefunguje uvnitř ...

Funkce příkazu LATEXu pro sazbu textu v tom tvaru, v němž je uveden ve zdrojovém souboru (`verbatim`), je založena na využití změny kategorie (category codes) jednotlivých znaků. Knuth v této souvislosti říká: „Je potřeba věnovat jistou péči tomu, aby vše proběhlo ve správném sledu…“. Kategorie znaku se od okamžiku, kdy je mu přiřazena, nemění. Proto `\verb` předpokládá, že je prvním příkazem, který se dívá na svůj textový parametr. Není-li tomu tak, TEX již přiřadil kategorie jednotlivým znakům a `\verb` již nemá šanci kategorii měnit. Například:

```
\verb+\error+
```

bude fungovat (vysází se „`\error`“), ale

```
\newcommand{\unbrace}[1]{#1}
\unbrace{\verb+\error+}
```

fungovat nebude (pokusí se spustit příkaz `\error`).

Proto se také v manuálu LATEXu tolik naléhá na to, aby se příkaz verbatim neobjevil v argumentu žádného jiného příkazu. Tyto příkazy jsou nejen křehké (fragile), ale dokonce zcela nepoužitelné jako parametr jiných příkazů, bez ohledu na ochranu pomocí `\protect` (viz Otázku ??).

110. Chyby související se změnou velikosti písmen

TEX poskytuje dva primitivní příkazy umožňující konverzi textu do malých písmen, `\lowercase`, a velkých písmen, `\uppercase`. Tyto příkazy se nepoužívají příliš často, ale dokáží nadělat v dokumentu zmatek.

Oba příkazy nerozvíjejí text, který je jejich parametrem — výsledkem příkazu `\uppercase{abc}` je „`ABC`“, avšak výsledek příkazu `\uppercase{\abc}` je `\abc` ať už `\abc` znamená cokoli. Příkazy jednoduše interpretují tabulku ekvivalentů mezi velkými a malými formami znaků. Nemají například smysl pro matematiku, a tak

```
\uppercase{About $y=f(x)$}
```

dává

```
ABOUT $Y=F(X)$
```

což zřejmě není to, co jsme původně zamýšleli.

Stejný problém může vzniknout s názvy prostředí v LATEXu, když LATEX užívá implicitně `\uppercase`. Například standardní třídy užívají `\uppercase` k nastavení záhlaví podle argumentu příkazů `\chapter` a `\section`, takže kapitola, jejíž nadpis obsahuje malou tabulku vyjádřenou pomocí `\begin{tabular}` … `\end{tabular}`, nastaví obsah záhlaví na `\begin{TABULAR}`, atd., což je však prostředí, které vůbec neexistuje.

111. Proč je znak # v makrech uveden dvakrát

Při psaní makra je třeba mít na paměti, že `##` zastupuje `#` obdobně jako `#1` zastupuje cokoli, co je prvním argumentem makra.

Definujeme-li a poté použijeme makro:

```
\def\a#1{...#1...#1...#1...}  \a{b}
```

makro po expanzi dává „…b…b…b…“, což jsme očekávali. Avšak dosadíme-li nyní za „…“:

```
\def\a#1{---#1---\def\x #1{xxx#1}}
```

`\a{b}` se rozvine na „---b---`\def\x b{xxxb}`“. Definuje se zde makro `\x`, které je *ohraničené* znakem `b` a které nemá žádný argument. Může se to zdát divné, ačkoli jde jen o zvláštní případ předchozího příkladu. Chceme-li, aby v `\a` bylo definováno makro `\x` s jedním argumentem, musíme upravit předchozí makro takto:

```
\def\a#1{---#1---\def\x ##1{xxx##1}}
```

`\a{b}` se nyní rozvine do tvaru „---b---`\def\x #1{xxx#1}`“, neboť `#1` bude nahrazeno znakem „b“ a `##` bude nahrazeno `#`.

Pro definici vnořenou uvnitř definice uvnitř definice se použije `####1`, jelikož v každé úrovni je `##` nahrazeno `#`. V další úrovni je nutné použít osm `#`, atd.

Q. Současný vývoj

112. Makro NFSS (The New Font Selection Scheme)

NFSS bylo rozšířením LaTeXu napsané Frankem Mittelbachem a Rainerem Schöpfem. Je popsáno v TUGboatu, 1989, **10**(2). V češtině je filosofie makra popsána na str. 66–70 knihy Petra Olšáka *Typografický systém TEX*. V tradiční sazbě jsou písma popsána čtyřmi parametry: *rodina písma* (family) (např. computer modern), *duktus* (series) (tj. světlost či temnost písma), *varianta* (shape) (např. kurzíva) a *stupeň* (size) (např. 10pt). NFSS zavádí mechanismus, který umožňuje uživateli nezávisle měnit libovolný z těchto parametrů. NFSS umožňuje poměrně snadno užívat nestandardní fonty (např. PostScriptové fonty) v LaTeXu a jednoduše měnit matematické fonty. Makro umožňuje rovněž dynamické načítání fontů (tj. nikoliv v době tvorby formátu, ale až při formátování dokumentu).

Jelikož verze LaTeXu 2.09 již není udržována a vylepšována, název NFSS se stal zavádějícím, protože nyní vlastně neexistuje žádný „starý“ způsob výběru fontu, se kterým by tento nový mohl kontrastovat, neboť LaTeX (tj. LaTeX 2_ε) v sobě NFSS zahrnuje.

113. LaTeX 2_ε (nový standard LaTeXu)

LaTeX 2_ε je nová verze balíku maker LaTeX, která je připravena a rozvíjena tvůrčí skupinou projektu LaTeX3. Testovací fáze LaTeXu 2_ε začala v červnu roku 1994 a je nyní standardem LaTeXu. LaTeX 2.09 není již více podporován.

LaTeX 2_ε je zpětně kompatibilní s LaTeXem 2.09, ale má navíc některé nové vlastnosti:

- NFSS (viz Otázku **??**) se stává standardem.
- SLITEX je nyní do něj zahrnut, takže není potřeba používat zvláštního formátu.
- Lepší ovládání plovoucích objektů (prostředí `figure` a `table`).
- Obsahuje dokumentované rozhraní pro autory nových stylů a tříd (avšak zatím chybí obdobné rozhraní pro návrháře vzhledu dokumentu)
- Rozšířené příkazy pro práci s boxy, např. možnost volby výšky minipage.
- Křížový odkaz `\ref` může být použit v `\caption` bez ochrany příkazem `\protect`.
- `\newcommand` umožňuje definovat příkazy s jedním volitelným argumentem.
- standardní balík pro vkládání grafiky a užití barev.

LaTeX 2_ε je průběžně udržován a vyvíjen, proto lze sepsat chyby a těžkosti při jeho užívání vyvoláním příkazu „`latex latexbug`“ a vygenerovanou zprávu zaslat na elektronickou adresu `latex-bugs@uni-mainz.de`.

114. Projekt LaTeX3

Tvůrčí skupina projektu LaTeX3 je malá skupina dobrovolníků, jejichž cílem je vytvořit nový systém pro sazbu dokumentů založený na zásadách, jež jsou obsaženy v současném LaTeXu Leslieho Lamporta. Tento nový systém zůstane volně šiřitelný a bude na všech úrovních podrobně zdokumentován.

Skupina projektu LaTeX3 již uvedla svůj první produkt, LaTeX 2_ε (viz Otázku **??**). Balík maker vychází z původního Lamportova kódu a je upraven tak, aby ho bylo možno lépe udržovat a rozšiřovat.

115. Užití TEXu k sazbě z SGML souborů

Jonathan Fine (`J.Fine@pmms.cam.ac.uk`) vytváří software, který bude umožňovat TEXu sázet přímo ze souborů SGML. Výsledkem práce by měl být balík maker, jenž, oproti dříve zmíněným konverzním programům (viz Otázku **??**), interpretuje přímo zdrojový SGML soubor.

116. Projekt Omega

Omega (Ω) je program, jehož stavba vychází z TEXu, a který pracuje s šestnáctibitovou vnitřní reprezentací znaků (Unicode). To umožňuje pracovat s většinou písem světa bez komplikací s jejich kódováním. Omega má také velmi mocný koncept vstupních a výstupních filtrů, které uživateli umožňují pracovat s existujícími překódovávacími tabulkami, atd. Omega je trvající projekt Johna Plaice (`plaice@cse.unsw.edu.au`) a Yannise Haralambouse (`yannis@fluxus-virus.com`). Chcete-li se dostat do diskusní skupiny projektu, můžete zaslat zprávu `subscribe omega <vaše jméno>` na adresu `listserv@ens.fr`. WWW stránka projektu je `http://www.ens.fr/omega/`

117. Projekt $\mathcal{NTS}$

Projekt $\mathcal{NTS}$ (New Typesetting System) poprvé spatřil světlo světa na Hamburgském setkání DANTE v roce 1992 jako odpověď na úsilí vytvořit něco lepšího než je TeX. Cílem projektu není jen pouhé rozšíření TeXu, neboť, za prvé, vývoj TeXu byl Knuthem zastaven (viz Otázku ??), a za druhé, i kdyby bylo dovoleno dále vyvíjet program TeX, někteří členové týmu $\mathcal{NTS}$ cítí, že se ve své současné formě k dalšímu vývoji prostě nehodí. Ačkoli všichni účastníci projektu jsou úzce spojeni a svázáni s TeXem, docházejí k závěru, že konečný produkt bude mít s TeXem, kromě filosofie, jen velmi málo společného.

Nejprve, navzdory výhradám vyjádřeným při zahajovacím setkání, se skupina zaměřuje na rozšíření TeXu. Úpravy a rozšíření se implementují do TeXu za pomoci standardního změnového souboru (change-file). Tato rozšíření spolu s opravdovým TeXem, vytvoří systém nazvaný ε-TeX, který by měl být 100% kompatibilní s TeXem. Nadto bude možné vytvořit formát, jenž bude TeXem *bez* veškerých úprav a rozšíření.

Konečným cílem projektu je vytvořit zcela nový typografický systém, vystavěný na zkušenostech získaných v předchozích fázích projektu. Tento systém by se měl potom stát stabilním základem pro sazbu budoucnosti, obdobným tomu, jakým je TeX od doby, kdy byl poprvé nabídnut světu.

R. Možná zde *není* odpověď

118. Co dělat, když najdete chybu

Nejprve bychom si měli být jisti, že jsme *opravdu* našli chybu. Dvakrát se přesvědčme v knize o TeXu, LaTeXu (či o tom formátu, který používáme), zda tam není napsáno něco o chování, které pozorujeme. Srovnejme to, co vidíme, s odpověďmi, jež jsou uvedeny výše, zeptejme se všech lidí, o nichž víme, že mají zkušenosti s TeXem. Důvodů k takové opatrnosti je mnoho.

Nalezne-li někdo chybu v samotném TeXu, potom se zřejmě jedná o zcela výjimečnou osobu. Don Knuth si je tak jistý kvalitou svého programu, že nabízí nálezci chyby peněžitou odměnu; šeky však vyplňuje velice zřídka. Nalezne-li se opravdová chyba v samotném TeXu (METAFONTu, v CM fontech, v TeXbooku), není dobré psát ihned Knuthovi, jelikož on se na chyby dívá tak jednou, dvakrát do roka, a i v těchto chvílích se zabývá chybami, které mu doporučí tým jeho spolupracovníků. Nejprve je dobré kontaktovat Barbaru Beeton z AMS (`bnb@math.ams.org`), anebo skupinu uživatelů TeXu (TUG).

Jestliže je nalezena chyba v LaTeXu 2_ε měla by být postoupena týmu LaTeX3. K sepsání zprávy o chybě slouží soubor `latexbug.tex`, který je součástí distribuce LaTeX 2_ε. Tento soubor necháme zpracovat LaTeXem. Během zpracování jsme instruováni o tom, co je třeba dělat.

Prosíme, šetřte s časem této skupiny, která se snaží pracovat pro dobro celé LaTeXovské pospolitosti. Veškerý čas, který tito lidé stráví analýzou domnělých chyb, pak chybí při psaní a ladění nových maker.

Nalezne-li se chyba v LaTeX2.09, nebo v nějakém jiném neudržovaném softwaru, nelze toho udělat mnoho. Pomoc (podporu tohoto softwaru) lze nalézt v newsgroup jako například `comp.text.tex` nebo v diskusních listech `uktex@tex.ac.uk`, ale prezentací domnělých chyb na těchto fórech se vystavujete možnému posměchu! Jinak je potřeba si najít ochotného konsultanta[10] v otázkách spojených s TeXem.

[10] TUG udržuje registr konsultantů a UKTUG vytváří další.

S. Lokalizace pro češtinu a slovenštinu, CSTUG

119. Co je to CSTUG

CSTUG je občanské sdružení uživatelů TEXu. Sdružuje individuální a kolektivní členy a jeho oficiální název je „Československé sdružení uživatelů TEXu". Členství je dobrovolné. Stanovy sdružení jsou dostupné na `ftp://ftp.cstug.cz/pub/tex/local/cstug/info/`

120. Jaká je adresa a kontakt na sdružení CSTUG

Adresa sdružení (sídlo) je CSTUG, c/o FI MU, Botanická 68a, 602 00 Brno. Sem můžete posílat korespondenci týkající se CSTUGu. Kontaktní email je `cstug@cstug.cz` nebo `secretary@cstug.cz`.

121. Jaké mám výhody z členství v CSTUGu

Každý individuální člen se svým členstvím hlásí k poslání sdružení, které je specifikováno ve stanovách sdružení. Má právo dostávat občasník zvaný Zpravodaj, který vychází zpravidla alespoň čtyřikrát do roka.

Členové mají slevy na publikace vydávané nebo dotované sdružením, slevy na akcích a školeních pořádaných sdružením.

122. Jak se stanu členem CSTUGu

Přihlásit se můžete e-mailem nebo pozemní poštou na adrese sdružení písemně. Je třeba sdělit jméno a příjmení, rodné číslo (pro identifikaci plateb), adresu pro korespondenci, telefon, adresu a telefon do zaměstnání a email.

Pokud máte přístup k Internetu, můžete se přihlásit také elektronicky pomocí formuláře na WWW, na který se dostanete z domovské stránky CSTUGu (`http://www.cstug.cz/`).

Formulář přihlášky k vytisknutí na papír je také v `ftp://ftp.cstug.cz/pub/local/cstug/info/`.

123. Jaké jsou členské příspěvky a jak je zaplatím

Výše členských příspěvků v roce 1999 je 260,– Kč/Sk (individuální členové) resp. 1500,– Kč/Sk (kolektivní členství). Studenti po prokázání svého statutu platí členský příspěvek 160,– Kč/Sk. Je-li kolektivním členem česká nebo slovenská střední škola, platí členský příspěvek 500,– Kč/Sk ročně.

Aktuální výše příspěvků se dá najít na `http://www.cstug.cz/clenstvi#prispevky` nebo dotazem na `treasurer@cstug.cz`. Platby v ČR se poukazují na účet sdružení u Komerční banky, na účet číslo 34735-021/0100. Jako variabilní symbol se vždy uvádí rodné číslo individuálního člena resp. kontaktní osoby kolektivního člena.

Individuální členové na Slovensku mohou platit členské příspěvky buď osobně na MFF UK (proti stvrzence), anebo poštovní poukázkou. Adresa pro přímou osobní platbu je Ústav informatiky MFF UK, Mlynská dolina, Bratislava, a před placením je lépe domluvit se na adrese `chlebikj@dcs.fmph.uniba.sk` nebo zavolat na telefonní číslo 07/65424000 kl. 396, Janka Chlebíková, katedra vyučovania informatiky. Poštovní poukázkou se příspěvky platí na adresu Janka Chlebíková, Podzáhradná 19, 821 06 Bratislava.

Kolektivní členové ze Slovenska poukazují platby příspěvků na účet sdružení v Brně.

124. Jak si objednám CD-ROM TEXLive či 4AllTEX

Závazné objednávky posílejte na adresu sdružení nebo `orders@cstug.cz`.

125. Jak si objednám publikace vydávané sdružením nebo další TEXovou literaturu

Objednávejte na adrese knihkupectví Mareček, Botanická 68a, 602 00 Brno, `bookorders@cstug.cz`. Pokud uplatňujete členskou slevu, uvádějte své rodné číslo nebo přiložte kopii dokladu o zaplacení členského příspěvku. Je zde možno objednávat i zahraniční TEXovou literaturu a stará čísla Zpravodaje, pokud nejsou rozebraná.

126. Co to je Zpravodaj a jak ho získám

Zpravodaj je tištěný občasník sdružení. Zpravidla vycházejí 4 čísla ročně. Zpravodaj je rozesílán řádným členům sdružení, kolektivní členové dostávají po třech výtiscích od každého čísla.

127. Jak mám postupovat, když chci napsat článek do Zpravodaje?

Korespondence týkající se Zpravodaje se posílá na adresu `zpravodaj@cstug.cz`. Příspěvky ukládejte do adresáře `ftp://ftp.icpf.cas.cz/wagner/incoming/`. Uvedený adresář je uvolněn pro zápis. Pokud nemáte přístup na Internet, můžete zaslat příspěvek na disketě na adresu Zdeněk Wagner, Vinohradská 114, 130 00 Praha 3. Disketa musí být naformátována pro DOS. Nezapomeňte přiložit všechny soubory, které dokument načítá.

Pro sazbu Zpravodaje byl vytvořen a je udržován a zdokumentován styl `ftp://ftp.cstug.cz/pub/tex/local/cstug/zpravodaj/csbul.sty`. Pro usnadnění finální sazby editoři uvítají, když jej autoři budou pro psaní svých příspěvků používat. Dokumentace je k nalezení v souborech `csbuldoc.*` v adresáři `ftp://ftp.cstug.cz/pub/tex/local/cstug/zpravodaj/`

128. Kde je nejbližší zrcadlo archívu CTAN

Pro uživatele v České republice je nejbližší kopie CTANu v Brně na adrese `ftp://ftp.cstug.cz/pub/tex/CTAN/`. Na tomto serveru je také lokální archív CSTUGu — `ftp://ftp.cstug.cz/pub/tex/local/cstug/`.

Uživatelé ze západního Slovenska mají nejblíže na bratislavský server SAV `http://www.savba.sk/cstug/`, kam je zrcadlen brněnský archív.

Nejbližší zrcadlo CTANu na východě Slovenské republiky je na universitě v Košicích, `ftp://oko.fei.tuke.sk/pub/cstex/`.

Pro střední Slovensko je nejbližší zrcadlo CTANu v Banské Bystrici na `ftp://sunsite.uakom.sk/pub/mirrors/CTAN/`.